

Étude et supervision des graphes d'événements temporisés et temporels : vivacité, estimation et commande

THÈSE DE DOCTORAT

Spécialité : Automatique et Informatique Appliquée

ÉCOLE DOCTORALE d'ANGERS

présentée et soutenue publiquement

le 07 Décembre 2005

à l'ISTIA - Université d'Angers

par : Mohamed Khalid DIDI ALAOUI

devant le jury ci-dessous :

Mohammed EL BAGDOURI	Président du jury	<i>Professeur, Université de Technologie de Belfort Montbéliard</i>
Jean Jacques LOISEAU	Rapporteur	<i>Directeur de Recherche CNRS, IRCCyN, Nantes</i>
Geert Jan OLSDER	Rapporteur	<i>Professeur, Delft University of Technology</i>
Pascal YIM	Examineur	<i>Professeur, École Centrale de Lille</i>
Marc BOURCERIE	Examineur	<i>Professeur, Université d'Angers</i>
Philippe DECLERCK	Examineur	<i>Maître de conférences, Université d'Angers</i>

Directeurs de thèse :

M. BOURCERIE et P. DECLERCK

Étude et supervision des graphes d'événements temporisés et temporels : vivacité, estimation et commande

Résumé : Ce mémoire porte sur la modélisation et l'analyse de réseaux de Petri du type Graphes d'Événements (GE) temporisés et temporels, au moyen d'outils algébriques comme l'algèbre des diïdes ou l'algèbre des applications topicales. La modélisation mathématique de ces systèmes dynamiques à événements discrets (SDED) conduit à des systèmes d'équations et d'inéquations utilisant les opérations de minimum, de maximum et d'addition. Dans la première partie, la forme générale obtenue est un modèle dit d'intervalle qui permet de faire une classification des différents GE considérés, selon une analyse de ces fonctions. Le modèle algébrique des GE à flux temporels pour deux sémantiques, englobe les modèles de différents GE temporels qui modélisent le temps sur les places, les transitions ou les arcs (temporisés, p-temporels, t-temporels, arcs temporels,...). Dans une deuxième partie, nous analysons la vivacité des GE temporels. L'étude des phénomènes de synchronisation permet de vérifier l'existence du vecteur d'état par deux approches : la première utilise la théorie spectrale des fonctions (min, max, +) et peut être appliquée sur des GE à flux temporels ; la deuxième exploite une série de matrices et concerne les GE P-temporels. Les SDED sont soumis à des perturbations comme les défaillances qui peuvent générer une mauvaise détermination du vecteur d'état. Nous traitons alors dans cette troisième partie le problème de l'estimation d'état dans les GE temporisés et à flux temporels. L'estimateur proposé permet en même temps le développement d'une nouvelle approche pour la détection de défaillances. Enfin, le problème de la commande optimale est étudié pour les GE à flux temporels. La détermination de la commande est faite en appliquant une méthode du type point fixe comme pour l'estimation. La particularisation de celle-ci au cas des GE temporisés, permet de retrouver les classiques équations "Backward". La vérification de l'existence d'un contrôle optimal est faite en exploitant la théorie spectrale après homogénéisation des fonctions (min, max, +).

Mots-clés : Systèmes dynamiques à événements discrets, graphes d'événements temporisés et temporels, algèbre des applications topicales, diïdes, point fixe, vivacité, estimation, commande.

Study and supervision of time and timed event graphs : liveness, estimate and control

Abstract : This report concerns the modeling and analysis of timed and time Event Graphs (EG), by means of algebraic tools like the dioids or algebra of topical functions. The mathematical modeling of these Discrete Event Dynamic Systems (DEDS) leads to equations and inequations systems using minimum, maximum and addition operations. In the first part, the general form obtained is called an "interval model" which allows making a classification of different EG considered, according to an analysis of these functions. The algebraic model of time stream EG for two semantics, includes the different time EG models which model time on the places, the transitions or the arcs (timed, p-time, t-time, time arcs ...). In the second part, we analyze the liveness of time EG. The study of the synchronization phenomena enables the verification of the existence of the state vector by two approaches : the first uses the spectral theory of the (min, max, +) functions and can be applied to time stream EG ; the second one exploits a matrices series and concerns p-time EG. The DEDS are subjected to disturbances like failures which can generate a bad determination of the state vector. Then We deal the problem of the state estimate in timed and time stream EG. The estimator proposed also allows the development of a new approach for the fault detection. Finally, the optimal control for time stream EG is studied. The determination of the control is made by applying a fixed point approach like the one used for estimation. The particularization of this approach to the timed EG case, allows finding the traditional "Backward" equations. The verification of the existence of an optimal control is made by exploiting the spectral theory of the (min, max, +) homogeneous functions.

Key-words : Discrete event dynamic systems, time and timed event graphs, algebra of topical functions, dioids, fixed point, liveness, estimate, control.

Table des matières

1 Outils algébriques	7
1.1 Introduction	7
1.2 Éléments de la théorie des treillis	8
1.2.1 Ensembles ordonnés	8
1.2.2 Notions de base sur les treillis	10
1.3 Dioïdes et structures ordonnées	12
1.3.1 Notions de base sur les dioïdes	12
1.3.2 Dioïdes comme structures ordonnées	13
1.3.3 Dioïdes des matrices	16
1.4 Éléments de la théorie de la résiduation	17
1.4.1 Homomorphismes de dioïdes, isotonie et continuité	17
1.4.2 Applications résiduables et dualement résiduables	19
1.5 Théorie du point fixe	24
1.5.1 Propriétés des équations de type point fixe	24
1.5.2 Un algorithme de calcul du point fixe	28
1.6 Fonctions topicales et théorie spectrale	30
1.6.1 Fonctions topicales	30
1.6.2 Fonctions (min, max, +)	33
1.7 Conclusion	38
2 Modèles et classes de systèmes temporels	39
2.1 Introduction	39
2.2 Les réseaux de Petri	41
2.2.1 Rappels et notions de base	41
2.2.2 Quelques propriétés des RdPs	43
2.2.3 Sous classes des RdPs	44
2.3 Graphes d'événements temporisés (GET)	46
2.3.1 Temporisation et fonctionnement d'un graphe d'événements	46

2.3.2	Equations aux dateurs et aux compteurs	48
2.3.3	Représentation d'état explicite	53
2.4	Graphes d'événements du type intervalle	56
2.4.1	Graphes d'événements p-temporels	56
2.4.2	Graphes d'événements t-temporels	64
2.4.3	Graphes d'événements à arcs temporels	65
2.4.4	Graphes d'événements à flux temporels	66
2.5	Classification des graphes d'événements temporisés et temporels	72
2.6	Autres modèles : exemples de systèmes (min, max,+)	75
2.6.1	Système de production : "Descriptor systems"	75
2.6.2	Système logique	77
2.7	Conclusion	78
3	De la validation des modèles à la vivacité des graphes d'événements p-temporels	79
3.1	Position du problème et contexte de l'étude	79
3.2	Validation des modèles	80
3.2.1	Modèle d'intervalles	80
3.2.2	Compatibilité dans les systèmes de type ((max, +), (min, max, +))	81
3.2.3	Compatibilité dans les systèmes de type ((min, max, +), (min, +))	85
3.2.4	Compatibilité dans les systèmes de type ((max, +), (min, +))	87
3.3	Vivacité dans les graphes d'événements p-temporels	88
3.3.1	État de l'art	88
3.3.2	Re-formulation du modèle	91
3.3.3	Approche basée sur le graphe de calcul et le vecteur spectral	95
3.3.4	Approche utilisant des séries de matrices	100
3.4	Illustration	102
3.5	Conclusion	104
4	Estimation d'état optimale des graphes d'événements temporisés et temporels. Application à la détection de défaillances	107
4.1	Introduction	107
4.2	Estimation et détection de défaillances pour les systèmes continus	108
4.2.1	Rappels sur les observateurs	108
4.2.2	Rappels sur la détection de défaillances	110
4.3	Estimation en fonctionnement nominal dans les graphes d'événements	113
4.3.1	Synthèse d'un estimateur	114
4.3.2	Existence d'une estimation optimale	116

4.4	Algorithmes de calcul du plus grand état	121
4.4.1	Cas d'un graphe d'événements à flux temporels	121
4.4.2	Cas d'un graphe d'événements temporisé	122
4.5	Illustration	124
4.5.1	Cas d'un graphe d'événements temporisé	125
4.5.2	Cas d'un graphe d'événements à flux temporel	127
4.6	Conclusion	130
5	Commande optimale des graphes d'événements temporisés et temporels	133
5.1	Introduction	133
5.2	Commande des graphes d'événements temporisés	134
5.2.1	Re-formulation de la méthode "Backward"	134
5.2.2	Synthèse de la commande par la méthode de propagation de contraintes MD	136
5.2.3	Commentaires sur le co-état	141
5.2.4	Illustration	141
5.3	Commande en juste à temps des graphes d'événements à flux temporels	144
5.3.1	Formulation et existence d'une loi de commande	144
5.3.2	Calcul de la commande	147
5.3.3	Illustration	148
5.4	Conclusion	149
6	Conclusion et perspectives	151
	Bibliographie	155

Introduction

L'automatique classique considère des systèmes dynamiques où le changement d'état dépend du temps, en continu. Le comportement des systèmes dynamiques en automatique classique est souvent modélisé par des équations, différentielles ou aux différences. En revanche, les Systèmes Dynamiques à Événements Discrets (SDED) correspondent à des processus dont l'état change à l'occurrence d'un événement. Cela recouvre un grand nombre de processus comme les ateliers flexibles, les réseaux de transport, les systèmes informatiques, les systèmes multimédias, ... Une troisième catégorie correspond aux modèles hybrides qui fusionnent ces deux types de phénomènes comme l'indique son intitulé. Ce mémoire considère ainsi les Systèmes Dynamiques à Événements Discrets qui sont l'objet d'attention et d'études de nombreux chercheurs depuis plusieurs décennies, ceci à l'échelle internationale.

Parmi les différents formalismes, les réseaux de Petri jouent un rôle important, présentant la caractéristique d'être un outil à la fois graphique et mathématique de modélisation. L'emploi de ces graphes pour spécifier et analyser ces systèmes, s'est généralisé ces dernières années, car ils permettent de décrire des phénomènes d'assemblage, de synchronisation, de cumul, de partage de ressources, ... Parmi la multitude des modèles développés, deux types de réseaux de Petri se dégagent indiscutablement : les graphes d'états et les graphes d'événements. Pour cette deuxième catégorie, la classe des graphes d'événements temporisés joue un rôle important en raison de son comportement déterministe, moyennant certaines hypothèses car elle intègre explicitement le temps. L'évolution du système est alors décrite par des équations définies dans un dioïde, c'est à dire un semi-anneau idempotent, avec un nouveau type de linéarité par rapport à cette algèbre. A côté de la classe maintenant bien connue des graphes d'événements temporisés, les réseaux de Petri temporels sont une classe importante des réseaux de Petri. Ils sont un langage graphique, permettant de décrire les systèmes dynamiques à événements discrets, à contrainte de temps, sur l'ensemble des places, des transitions ou des arcs. La dynamique de tels systèmes modélisés par ces graphes dépend à la fois du marquage et des intervalles de temps associés aux places.

Les travaux présentés dans cette thèse portent sur l'étude et la supervision des graphes d'événements temporisés et temporels. Les graphes d'événements temporels sont une sous-

classe des réseaux de Petri où l'on a des contraintes de temps présentant la forme d'intervalle. De ce fait, nous désignons dans ce mémoire, ces différents graphes sous le nom de graphes d'événements du type intervalle. On verra que ce modèle fait intervenir des opérations de maximisation, de minimisation et aussi d'addition. Répondant à ces critères, on peut citer les graphes d'événements p-temporels, t-temporels, arcs temporels, à flux temporels...

L'exploitation des ces nouveaux modèles nécessite le développement et l'exploitation d'outils mathématiques plus généraux que la structure algébrique du dioïde : il sera nécessaire d'aborder ce que l'on appelle l'algèbre des fonctions topicales et en particulier, les fonctions (min, max, +), domaine mathématique récemment développé. La notion de vecteur spectral jouera dans le cadre de ce mémoire, un rôle important mais inhabituel : il ne sera pas utilisé pour calculer un taux de production mais pour vérifier l'existence d'au moins une solution dans des problèmes écrits sous forme de point fixe.

Nous détaillons maintenant, les problèmes abordés dans ce mémoire, qui traitent de la vivacité, de l'estimation et de la commande.

Vivacité

De manière générale, les propriétés comme la bornitude, l'atteignabilité et la vivacité, ont fait l'objet de nombreux travaux d'études et de recherches dans le domaine des réseaux de Petri, car un système bien conçu doit pouvoir éviter ces problèmes. Il s'est avéré cependant, que la vérification de telles propriétés pose souvent des difficultés. En particulier, l'étude de la propriété de vivacité des réseaux de Petri, est toujours actuelle motivée par la nécessité de s'assurer du non blocage dans le système de production modélisé.

Il est fréquent, que des tâches manufacturières se déroulent en parallèle dans des systèmes de production. Cela met en évidence des phénomènes de synchronisation et de partage de ressources. Pour un strict respect du bon fonctionnement du système, on doit obliger les activités manufacturières à se synchroniser. Malheureusement ce n'est pas toujours le cas, et dans plusieurs situations on se retrouve devant une mauvaise synchronisation. Par conséquent, on assiste non seulement à une utilisation inefficace du système, mais également à un blocage partiel ou total des activités. Dans l'objectif d'une bonne conception des systèmes manufacturiers, il faut alors étudier et analyser préalablement le problème de la vivacité des systèmes.

Dans le cadre de l'étude de la vivacité, nous nous intéressons à la classe des graphes d'événements p-temporels qui présente des spécificités. Si les propriétés comportementales dépendent naturellement de la structure du graphe d'événements p-temporel, il faut également considérer les intervalles de temps associés aux places ainsi que le marquage initial. Une des caractéristiques de ce modèle est la possibilité de mort de jetons, s'ils séjournent dans une place plus qu'une

durée maximale déterminée. Cette mort se traduit par la disparition du jeton, qui peut être l'origine des phénomènes de blocage ou de désynchronisation, au niveau des transitions qui ont plus d'une place en amont.

Supervision : estimation, commande

Dans un contexte général de commande, le problème de la supervision est :

- de calculer les dates au plus tard des tirs des transitions d'entrée de manière à ce que les tirs des sorties arrivent au plus tard avant les dates désirées.
- d'estimer les valeurs passées de l'état à partir des valeurs connues de l'entrée et de la sortie, ce qui sera l'objet de cette étude.
- de prédire la future évolution de la sortie et de vérifier l'optimum de la commande.

Cependant, les systèmes dynamiques à événements discrets sont soumis à des perturbations comme les défaillances, qui interrompent le fonctionnement habituel du système et réduisent les capacités de prédiction et de commande de la future évolution du système. Les perturbations peuvent être les suivantes :

- la nature physique mais aussi humaine du processus entraîne une variation des temporisations.
- le processus est soumis à des défaillances qui doivent être traitées et qui sont donc également à l'origine de variations des temporisations.
- un des rôles de la maintenance est de prévenir les pannes (maintenance prédictive). Les opérations de maintenance prédictive se traduisent également par une perturbation de la production.

Ces perturbations peuvent être qualifiées d'internes par opposition aux perturbations externes comme les variations de consignes ou de l'approvisionnement du système en produits et pièces. Les perturbations internes produisent des variations du modèle ou même des ruptures de la description du modèle ce qui peut être à l'origine d'une mauvaise détermination du vecteur d'état. Dans ce chapitre, nous considérons donc le problème de l'estimation d'état.

En dehors de la reconstruction d'état, cet estimateur peut avoir comme autre rôle la détection de défaillance. A notre connaissance, ce sujet n'a jamais été abordé dans un cadre "système de production / algèbre des fonctions topicales".

Le contrôle est aussi un sujet important d'études dans le domaine des systèmes dynamiques à événements discrets. En effet, dans de nombreux processus (ateliers flexibles, réseaux de transport, ...) modélisés par des SDED, on cherche à contrôler les entrées afin d'atteindre des performances prédéfinies. Problème classique au laboratoire LISA, différents travaux ont abordé le sujet de la commande. C'est dans le but de continuer ce thème de recherche que vient s'inscrire cette contribution dans le chapitre 5 et de généraliser l'approche dite "Backward". L'objectif

sera de développer une méthode de commande optimale sur de nouveaux modèles du type “intervalle”.

Les lignes directrices des cinq chapitres de ce mémoire, ainsi que son organisation sont présentés de la manière suivante :

- Dans le premier chapitre, nous exposons un ensemble de concepts et d’outils mathématiques importants pour le traitement des problèmes posés dans ce mémoire. Dans une première partie, sont passés en revue les concepts et les fondement de base des treillis, des dioïdes et de la théorie de la résiduation. Ensuite, des résultats sur la théorie du point fixe sont présentés. Ils représentent en fait un outil important pour la suite de ce mémoire. Dans la dernière partie, nous présentons la théorie spectrale des fonctions (\min , \max , $+$) dont le rôle est principal dans les chapitres 3, 4 et 5.
- Le deuxième chapitre est dédié à la modélisation dans ces nouvelles structures algébriques, de différents graphes d’événements temporisés et temporels. La première partie de chapitre est une synthèse bibliographique qui regroupe des résultats bien connus sur les propriétés des réseaux de Petri. Quelques résultats classiques sur la modélisation des graphes d’événements temporisés dans l’algèbre des dioïdes, sont également exposés. Dans la deuxième partie, nous abordons la modélisation des graphes d’événements temporels (du type intervalle) dans l’algèbre des dioïdes. L’analyse de la forme générale des différents modèles nous permettra de faire une classification des différents graphes considérés, l’objectif étant d’avoir un modèle général recouvrant les différents cas. Cette modélisation constitue le premier pas pour les problèmes que nous allons traiter dans les chapitres suivants.
- Dans le troisième chapitre, le centre d’intérêt est la validation de modèles généraux afin de pouvoir les utiliser dans le cadre des chapitres de l’estimation et de la commande. On s’intéresse en particulier à l’analyse de la vivacité des graphes d’événements p-temporels. L’étude des phénomènes de synchronisation permet de vérifier l’existence d’au moins une trajectoire du vecteur d’état. Deux approches sont présentées. La première utilise le vecteur spectral en se basant sur le graphe de calcul de modèles généraux (\min , \max , $+$). Après un rapide état de l’art, la deuxième approche introduit une série particulière de matrices qui permet de résoudre le problème de la vivacité pour les graphes d’événements P-temporels.
- Dans le quatrième chapitre, on traite le problème de l’estimation du plus grand état possible. Nous rappelons d’abord les méthodes des observateurs ainsi que les approches de détection de défaillances utilisées dans le cas des systèmes continus. Nous introduisons ensuite des résultats concernant la vérification de l’existence d’un état en exploitant la théorie spectrale des fonctions (\min , \max , $+$) pour les graphes d’événements temporels. Se basant sur une approche du type propagation de contraintes, deux algorithmes spécifiques sont

proposés afin d'estimer le plus grand état des graphes d'événements temporisés (GET) et à flux temporels. L'estimateur synthétisé, est réutilisé dans ce chapitre afin de proposer une approche permettant la détection de défaillances.

- Le cinquième chapitre a pour but d'étudier le problème de la commande optimale des graphes d'événements temporisés et à flux temporels. Une première approche de commande des GET est proposée. Elle intègre un algorithme spécifique aux GET calculant la plus grande commande et le co-état. Cette approche est comparée avec la méthode "Backward". Le problème de la commande optimale des graphes d'événements à flux temporels est résolu grâce à une approche duale de celle de l'estimation vue au chapitre 4. Cette deuxième approche procède dans un premier temps à la vérification de l'existence d'un contrôle optimal de la classe modèle d'intervalle, exploitant la propriété d'homogénéité des fonctions (min, max, +). Dans un deuxième temps la détermination de la commande est faite en appliquant une méthode du type "point fixe". Cette dernière permet l'élaboration d'un algorithme de calcul de la commande optimale spécifique aux graphes d'événements à flux temporels. L'approche généralise de manière naturelle les équations "Backward" appliquées classiquement pour synthétiser la commande des GET.

Dans le dernier chapitre, nous allons dresser un bilan des travaux réalisés dans cette thèse en proposant des perspectives et d'autres pistes de recherches.

Chapitre 1

Outils algébriques

1.1 Introduction

Dans ce chapitre, nous allons passer en revue la théorie de base des dioïdes et des fonctions topicales. Les notations, les outils algébriques et les principaux concepts qui seront utilisés dans ce manuscrit seront précisés.

Dans un premier temps, nous introduirons les notions de base relatives à la théorie des treillis. Ensuite, nous présenterons les dioïdes qui sont liés à la théorie des treillis. Cela, vient du fait, comme nous allons le voir, qu'un dioïde peut être considéré comme un sup-demi-treillis.

Les lois de compositions internes (la somme et le produit) définies sur des dioïdes ne sont pas inversibles en général. Pour remédier à ce problème, on fait appel à la théorie dite de la résiduation. Nous exposerons un bref rappel sur les éléments de base qui constituent cette théorie, ainsi que les principaux résultats relatifs à la résiduation. Ces derniers seront appliqués à la résolution d'un certain type d'inéquations. Un autre type d'équations dites équations au point fixe définies sur des dioïdes complets sera étudié. Deux cas intéressants d'équations ($x = ax \oplus b$ et $x = a \setminus x \wedge b$) seront considérés dans le cadre de l'étude de la théorie du point fixe.

La dernière partie de ce chapitre est moins classique car nous parlerons des fonctions monotones (isotones) et homogènes. Ces fonctions sont apparues dans plusieurs travaux : [MS69], [Ols91b], [Gun94b], [GK95] et [CG79]. Les fonctions vérifiant les propriétés de monotonie et d'homogénéité [CT80] sont appelées topicales [BM03] et [KG94]. Ces fonctions peuvent prendre des formes plus ou moins complexes selon la nature du système modélisé. Dans le contexte de cette thèse, nous nous intéresserons en particulier à une classe des fonctions topicales qui sont les fonctions $(\min, \max, +)$. Ces dernières apparaissent en modélisant certains systèmes dynamiques à événements discrets ainsi que d'autres modèles particuliers qui seront présentés dans le chapitre 2. Un autre sujet important abordé dans cette partie est l'étude de l'existence

de point fixe et le calcul du vecteur spectral (vecteur temps de cycle) relatif aux fonctions (min, max, +). Le calcul du vecteur spectral jouera un rôle très important dans la suite de cette thèse surtout dans le cadre de la résolution de systèmes (min, max, +) de type $x \leq f(x)$ et $x \geq f(x)$.

Des présentations approfondies des outils algébriques sont données, pour l'essentiel dans [BCOQ92], [Bir40], [BJ72], [CG79], [CGQ98], [Gau92], [GG98b], [Gun03], [GK95] et [GM01] et correspondent donc à une littérature importante.

1.2 Éléments de la théorie des treillis

L'association d'une relation d'ordre avec un ensemble (ou sous ensemble) définit un ensemble ordonné qui à son tour introduit un treillis. Nous rappelons brièvement dans cette partie un ensemble de notions, de définitions et de propriétés sur les ensembles ordonnés et les treillis. Le lecteur intéressé par les structures algébriques pourra consulter en particulier [BCOQ92], [KG95], [BJ72], [GM01] et [DP02].

1.2.1 Ensembles ordonnés

On rappelle qu'une relation d'ordre sur un ensemble E , notée \leq , est une relation binaire :

réflexive $\forall x \in E : x \leq x$
 transitive $x \leq y$ et $y \leq z \implies x \leq z$
 antisymétrique $x \leq y$ et $y \leq x \implies x = y$

Un ensemble E est ordonné, s'il est muni d'une relation d'ordre \leq . On le notera (E, \leq)

On dit que deux éléments x et y dans E sont non-comparables, si aucune des relations suivantes $x \leq y$ et $y \leq x$ n'est vérifiée.

Définition 1.1. (*Ordre partiel*)

Soit un ensemble ordonné (E, \leq) . S'il existe deux éléments $x, y \in E$ tels que x et y sont non-comparables, alors la relation d'ordre \leq est partielle. On dit aussi que (E, \leq) est partiellement ordonné.

Exemple 1.1 :

Soit l'ensemble des entiers naturels \mathbb{N}^* . La relation d'ordre \leq est définie comme suit :

$$x \leq y \text{ si et seulement si, } x \text{ divise } y$$

Il est clair alors que (E, \leq) est partiel ($3 \not\leq 5$ car 3 n'est pas divisible par 5 dans \mathbb{N}^* et inversement)

Définition 1.2. (*Ordre total*)

Soit (E, \leq) , si $\forall x, y \in E$ tels que $x \leq y$ ou $y \leq x$. On dit que l'on a un ordre total et E est appelé un ensemble totalement ordonné.

Exemple 1.2 :

Les deux ensembles (\mathbb{R}, \leq) (\mathbb{Z}, \leq) , où \leq est une relation d'ordre naturel sont totalement ordonnés.

Un ensemble (E, \leq) totalement ordonné est appelé aussi une chaîne.

Les ensembles totalement ou partiellement ordonnés sont souvent décrits graphiquement en utilisant un diagramme dit diagramme de Hasse. Ce dernier est un graphe acyclique direct tel qu'il existe un arc direct qui va de x vers y si et seulement si, $x \leq y$ (dans ce cas, x et y sont comparables). Notons que les arcs triviaux $x \mapsto x$ et les arcs se déduisant par transitivité sont omis. L'ordre de croissance est pris conventionnellement dans le sens du bas vers le haut.

Exemple 1.3 :

Soit l'ensemble $(\{x, y, z, t\}, \leq)$ partiellement ordonné avec son diagramme de Hasse montré dans la figure 1.1. Chaque élément est représenté par un point.

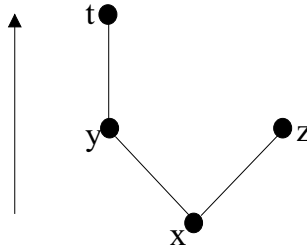


Figure 1.1 : Diagramme de Hasse

On a $x \leq y$, $x \leq z$ et $y \leq t$. On peut constater dans la figure 1.1 que y et z , ainsi que z et t sont non-comparables.

Dans la définition qui suit, nous donnons un vocabulaire relatif aux éléments dans les ensembles ordonnés. Remarquons que ce vocabulaire n'est pas standard et change selon les auteurs.

Définition 1.3. (Vocabulaire)

Soit (E, \leq) un ensemble ordonné et F un sous-ensemble non vide de E ($F \subseteq E$).

- Soit $F \subseteq E$, tout élément $b \in F$ vérifiant : $\nexists x \in F, x \neq b$ tel que $b \leq x$ est appelé **élément maximal** de F . De même, on appelle **élément minimal** de $F \subseteq E$ tout $a \in F$ vérifiant : $\nexists x \in F, x \neq a$ tel que $x \leq a$
- Lorsqu'il existe un **élément maximal** (respectivement **minimal**), il n'est pas toujours unique.
- Un élément $b \in E$ vérifiant $\forall x \in F : x \leq b$ est appelé un **majorant** ou **borne supérieure**. Si $b \in F$, alors b est unique et on l'appelle **maximum** de F .
- De même $a \in E$ est un **minorant** ou **borne inférieure** de F si et seulement si, $\forall x \in F : a \leq x$. Si $a \in F$, alors a est appelé **minimum** de F et il est unique.

- Un sous ensemble $F \subseteq E$ est dit borné s'il admet un majorant et un minorant.
- Lorsque l'ensemble des majorants ou bornes supérieures de $F \subseteq E$ a un plus petit élément, ce dernier est appelé **plus petite borne supérieure** de F , et notée $\sup(F)$. De même lorsque l'ensemble des minorants ou **bornes inférieures** de F a un plus grand élément, on l'appelle **plus grande borne inférieure** de F . On le note $\inf(F)$. On parle aussi respectivement de **supremal** ou **infimal** de F pour $\sup(F)$ et $\inf(F)$. Ces éléments sont uniques s'ils existent.
- L'ensemble (E, \leq) est dit complet si toute partie $F \subseteq E$ admet une borne supérieure notée $\bigvee_{x \in F} x$. De même il est dit complet pour l'ordre dual si $\text{Inf}(F)$ existe, elle est notée $\bigwedge_{x \in F} x$.

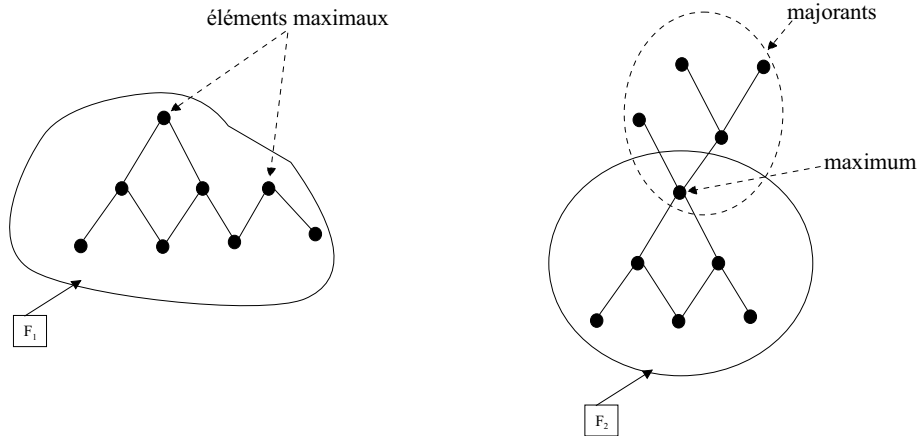


Figure 1.2 : Représentation des éléments maximaux, maximums et majorants des ensembles F_1 et F_2

La figure 1.2 montre les différentes définitions.

1.2.2 Notions de base sur les treillis

Définition 1.4. (Demi-treillis)

Un ensemble ordonné (E, \leq) est un **sup-demi-treillis** si toute partie de deux éléments x_1 et x_2 de E admet une plus petite borne supérieure (plus petit majorant) b . On notera $b = x_1 \vee x_2$. Duale, E est un **inf-demi-treillis** si toute partie $\{x_1, x_2\}$ admet une plus grande borne inférieure (plus grand minorant) $a = x_1 \wedge x_2$.

Définition 1.5. (Treillis)

On dit que l'ensemble (E, \leq) est un treillis, si à la fois (E, \leq) est sup-demi-treillis et inf-demi-treillis.

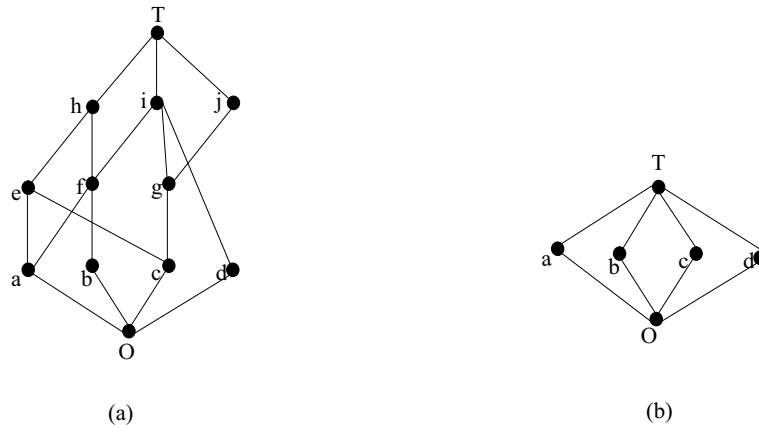


Figure 1.3 : Diagrammes de Hasse pour deux ensembles ordonnés différents[DP02]

Exemple 1.4 : .

Soit l'ensemble $E = \{O, a, b, c, d, e, f, g, h, i, j, T\}$ muni de la relation d'ordre \leq dont le diagramme de Hasse est montré sur la figure 1.3a. On est tenté à la première vue du diagramme, de dire que $b \vee c = i$. Or, l'ensemble des majorants de b et c est égal à $\{T, h, i\}$. Comme cet ensemble des majorants de b et c a deux éléments minimaux distincts h et i , par conséquent $b \vee c$ n'existe pas. Il en découle que l'ensemble E n'est pas un sup-demi-treillis, et par suite n'est pas un treillis.

Exemple 1.5 : .

L'ensemble des intervalles de \mathbb{R} ordonné par l'inclusion a une structure de treillis, où la borne supérieure (respectivement inférieure) est donnée par $A \vee B = A \cup B$ (respectivement $A \wedge B = A \cap B$) avec l'ensemble vide comme plus petit sous-ensemble.

Définition 1.6. (Demi-treillis complet)

Un *sup-demi-treillis complet* (respectivement *inf-demi-treillis complet*) est un ensemble E ordonné tel qu'il existe une plus petite borne supérieure (respectivement une plus grande borne inférieure) pour tout sous-ensemble fini ou infini de E .

Définition 1.7. (Treillis complet)

Un ensemble (E, \leq) est dit complet s'il est, à la fois sup-demi-treillis complet et inf-demi-treillis complet.

Exemple 1.6 : .

Soit l'ensemble ordonné dont le diagramme de Hasse est donné dans la figure 1.3b. Cet ensemble ordonné est un treillis complet.

Remarque : Si E muni des deux opérations \vee et \wedge est un treillis complet, alors il existe un plus petit élément (en anglais appelé bottom), noté ε , et un plus grand élément (top), noté T . L'élément ε est neutre pour \vee , il est aussi absorbant pour l'opération \wedge , $\varepsilon \wedge x = \varepsilon$. Dualement, T est neutre pour \wedge , il est absorbant pour \vee , $T \vee x = T$.

◇

Définition 1.8. (*Treillis distributif*)

Un treillis E doté des deux opérations \vee et \wedge est distributif si \vee distribue par rapport à \wedge , $x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z)$ et réciproquement, $x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z)$.

Nous concluons cette partie par le résultat fondamental suivant :

Théorème 1.9. *Un sup-demi-treillis complet E est un treillis complet si et seulement si, il admet un élément minimum.*

Preuve : La démonstration est donnée par exemple dans [Men97].

□

1.3 Diïdes et structures ordonnées

Dans cette partie, nous rappelons les fondements concernant les demi-anneaux idempotents ou diïdes, et nous montrons aussi le lien qui existe entre ces derniers et les structures ordonnées. Les diïdes apparaissent ainsi comme des demi-treillis. De plus, un diïde complet a la structure d'un treillis complet.

Ce travail bibliographique concernant cette partie a été fait à partir des documents suivants auxquels le lecteur pourra se référer : [BCOQ92], [Gau92], [CG79], [Men97] et [Lho03]

1.3.1 Notions de base sur les diïdes

Définition 1.10. (*Monoïde*)

Un monoïde (E, \oplus) est un ensemble doté d'une certaine loi de composition interne \oplus associative, et possédant un élément neutre. Si cette loi est commutative, ce monoïde est dit commutatif.

Définition 1.11. (*Monoïde idempotent*)

Un monoïde (E, \oplus) est dit idempotent si la loi \oplus est commutative, associative et idempotente, c'est-à-dire vérifie :

$$\forall a \in E \quad a \oplus a = a$$

Définition 1.12. (*Demi-anneau*)

Un demi-anneau est un ensemble \mathcal{D} muni de deux lois de compositions internes \oplus et \otimes , appelées respectivement somme et produit tels que :

1. (\mathcal{D}, \oplus) est un monoïde commutatif avec un élément neutre (zéro) noté ε .
2. (\mathcal{D}, \otimes) est un monoïde dont l'élément neutre e est appelé identité ¹.
3. La loi \otimes est distributive à gauche et à droite par rapport à la loi \oplus .

$$\forall a, b, c \in \mathcal{D} \quad a \otimes (b \oplus c) = (a \otimes b) \oplus (a \otimes c)$$

$$(a \oplus b) \otimes c = (a \otimes c) \oplus (b \otimes c)$$
4. Le zéro est absorbant pour le produit : $\forall a \in \mathcal{D}, a \otimes \varepsilon = \varepsilon \otimes a = \varepsilon$

Si le produit \otimes est commutatif, $(\mathcal{D}, \oplus, \otimes)$ est un demi-anneau commutatif.

Définition 1.13. (Dioïde)

Un dioïde $(\mathcal{D}, \oplus, \otimes)$ est un demi-anneau tel que le monoïde (\mathcal{D}, \oplus) est idempotent.

Si la loi \otimes est commutative, alors le dioïde $(\mathcal{D}, \oplus, \otimes)$ est commutatif.

Exemple 1.7 :

- L'ensemble $\mathbb{R} \cup \{-\infty\}$, muni de \oplus et \otimes qui correspondent respectivement à l'opération max et à l'addition usuelle, est un dioïde noté \mathbb{R}_{max} . Classiquement, ce dioïde est appelé algèbre $(\max, +)$ avec $e = 0$ et $\varepsilon = -\infty$.

- De même, on définit l'algèbre $(\min, +)$ comme un dioïde $(\mathcal{D} \cup \{-\infty\}, \oplus, \otimes)$ dans lequel \oplus coïncide avec l'opérateur min et $\varepsilon = +\infty$. L'algèbre $(\min, +)$ est aussi appelée algèbre tropicale.

Notons que les deux structures algébriques $(\max, +)$ et $(\min, +)$ sont commutatives.

1.3.2 Dioïdes comme structures ordonnées

Du fait que l'opération \oplus a la propriété d'idempotence, il est possible de définir une relation d'ordre sur un dioïde comme suit :

Définition 1.14. Relation d'ordre dans un dioïde

Dans un dioïde $(\mathcal{D}, \oplus, \otimes)$, on définit la relation d'ordre \leq par :

$$a \leq b \iff a \oplus b = b \iff a = a \wedge b$$

On peut vérifier facilement les propriétés de réflexivité, d'antisymétrie et de transitivité. En effet, puisque l'addition \oplus est idempotente, on a alors $a = a \oplus a \leq a$, ce qui montre la réflexivité. Considérons maintenant $a \leq b$ et $b \leq a$, d'après la définition précédente, on a $a = b = a \oplus b$, d'où l'antisymétrie. Pour la transitivité, il suffit de prendre $a \leq b$ et $b \leq c$ et de montrer que $a \leq c$. Pour cela, on sait que $b = a \oplus b$ et $c = b \oplus c$, d'où $c = (a \oplus b) \oplus c = a \oplus (b \oplus c) = a \oplus c$. On en déduit que $a \leq c$.

Théorème 1.15. La relation d'ordre définie dans le dioïde $(\mathcal{D}, \oplus, \otimes)$ vérifie les propriétés suivantes :

¹certains auteurs appellent e unité

1. $a \leq b \iff \exists c \in \mathcal{D}, b = a \oplus c.$
2. la relation d'ordre \leq est compatible avec les lois \oplus et \otimes de \mathcal{D}

$$a \leq b \implies \forall c \in \mathcal{D}, a \oplus c \leq b \oplus c$$

$$a \leq b \implies \forall c \in \mathcal{D}, a \otimes c \leq b \otimes c$$

Preuve : Soit $a \leq b$ et prenons $c = b$, d'où il est évident d'écrire $b = a \oplus c$. Réciproquement, soit $b = a \oplus c$

$\implies a \oplus b = a \oplus (a \oplus c) = (a \oplus a) \oplus c = a \oplus c = b$, d'où la propriété 1.

Nous démontrons maintenant la deuxième propriété. Soit a, b et c dans \mathcal{D} . $a \leq b$ est équivalent à $b = a \oplus b$. Alors, $b \oplus c = (a \oplus b) \oplus c = a \oplus b \oplus c = a \oplus c \oplus b \oplus c = (a \oplus c) \oplus (b \oplus c)$, d'où $a \oplus c \leq b \oplus c$.

De la même façon, $b \otimes c = (a \oplus b) \otimes c = (a \otimes c) \oplus (b \otimes c)$, d'où $a \otimes c \leq b \otimes c$.

La propriété de multiplication à gauche peut être démontrée de la même manière. On peut écrire donc $a \leq b \implies \forall c \in \mathcal{D}, c \otimes a \leq c \otimes b$

□

Remarque :

1. La relation d'ordre d'un dioïde est dite totale si et seulement si,

$$\forall a, b \in \mathcal{D}, a \leq b \text{ ou } b \leq a \iff a \oplus b = a \text{ ou } b.$$

2. Dans \mathbb{R}_{max} , la relation \leq coïncide avec l'ordre usuel ($1 \leq 3$). En revanche, dans \mathbb{R}_{min} , la relation \leq est l'inverse de l'ordre usuel ($3 \leq 1$).

◇

Définition 1.16. (Dioïde complet)

Un dioïde \mathcal{D} est dit complet si, dans ce dioïde, toute somme finie ou infinie d'éléments est définie et telle que la distributivité (à droite et à gauche) de \otimes par rapport à \oplus s'étende aux sommes infinies. $\forall A \subset \mathcal{D}, \forall B \subset \mathcal{D}$ et $\forall b \in \mathcal{D}$, on a :

$$\left(\bigoplus_{a \in A} a \right) \otimes b = \bigoplus_{a \in A} a \otimes b$$

$$b \otimes \left(\bigoplus_{a \in A} a \right) = \bigoplus_{a \in A} b \otimes a$$

On en déduit que

$$\left(\bigoplus_{a \in A} a \right) \otimes \left(\bigoplus_{b \in B} b \right) = \bigoplus_{(a,b) \in A \times B} (a \otimes b)$$

Dans un dioïde complet \mathcal{D} , le plus grand élément (top) existe et est noté T . Correspondant à la somme de tous les éléments du dioïde, on écrit alors $T = \bigoplus_{x \in \mathcal{D}} x$. L'élément T est absorbant pour la loi \oplus , $\forall a \in \mathcal{D}, T \oplus a = T$.

Exemple 1.8 :

Les dioïdes suivants sont complets :

1. $(\mathbb{R} \cup \{\pm\infty\}, \max, +)$, $T = +\infty$ avec la convention $(-\infty) + (+\infty) = (-\infty)$. Ce dioïde sera noté $\overline{\mathbb{R}}_{\max}$.
2. Dualement, le dioïde que l'on notera par la suite $\overline{\mathbb{R}}_{\min}$ est le dioïde $(\mathbb{R} \cup \{\pm\infty\}, \min, +)$, tel que $T = -\infty$ et $(+\infty) + (-\infty) = (+\infty)$

On peut munir un dioïde complet d'une borne inférieure dont la définition est donnée comme suit :

Définition 1.17. (Plus grande borne inférieure)

Soit \mathcal{D} un dioïde complet, $a, b \in \mathcal{D}$. La plus grande borne inférieure de a et b existe et est donnée par :

$$a \wedge b = \bigvee \{x \mid x \leq a \text{ et } x \leq b\} \quad (1.1)$$

D'une manière analogue, la plus grande borne inférieure d'une partie X de \mathcal{D} est donnée par :

$$\bigwedge X = \bigvee \{y \in \mathcal{D} \mid \forall x \in X, y \leq x\} \quad (1.2)$$

Remarquons que les ensembles à droite de 1.1 et 1.2 sont non vides puisque \mathcal{D} admet ε comme minimum.

L'opération \wedge est associative, commutative et idempotente et dont l'élément neutre est T ($\forall a, T \wedge a = a$). Cette loi (\wedge) fait du dioïde $(\mathcal{D}, \oplus, \wedge)$ un treillis complet. Par conséquent, les équivalences ci-dessous sont vérifiées :

$$a \geq b \iff a = a \oplus b \iff b = a \wedge b \quad (1.3)$$

En effet, le théorème 4.28 dans [BCOQ92] montre que tout couple d'éléments a une plus petite borne supérieure et ainsi que la relation d'idempotence de \mathcal{D} induit une structure de sup-demi-treillis sur \mathcal{D} . Comme la loi \oplus est associative, commutative et idempotente, le théorème 1.21 dans [Men97] donne le même résultat. Le sup-demi-treillis relatif à un dioïde complet est également complet car tout sous-ensemble non vide admet une plus petite borne supérieure. Comme un dioïde a un plus petit élément ε , le théorème 1.9 peut être appliqué et ainsi, un dioïde complet est un treillis complet. D'où, tous les résultats sur les treillis complets sont transposables sur les dioïdes complets.

Définition 1.18. (Dioïde distributif)

Un dioïde $(\mathcal{D}, \oplus, \otimes)$ est distributif s'il est complet et si pour tout sous-ensemble $\mathcal{C} \in \mathcal{D}$ et $\forall a \in \mathcal{D}$ on a :

$$\left(\bigwedge_{x \in \mathcal{C}} x \right) \oplus a = \bigwedge_{x \in \mathcal{C}} (x \oplus a)$$

et

$$\left(\bigwedge_{x \in \mathcal{C}} x \right) \otimes a = \bigwedge_{x \in \mathcal{C}} (x \otimes a)$$

L'introduction de la borne inférieure soulève la question du comportement mutuel entre \oplus et \wedge . En fait, si le dioïde n'est pas distributif, et si les applications $x \mapsto x \oplus a$ et $x \mapsto x \wedge a$ sont isotones, les inéquations suivantes sont toujours vérifiées :

$$\begin{aligned} (a \wedge b) \oplus c &\leq (a \oplus c) \wedge (b \oplus c) && \text{sous-distributivité} \\ (a \oplus b) \wedge c &\geq (a \wedge c) \oplus (b \wedge c) && \text{sur-distributivité} \end{aligned}$$

Rappelons qu'un treillis est distributif si les deux relations précédentes sont des égalités (1.8).

1.3.3 Dioïdes des matrices

Soit un dioïde \mathcal{D} des "scalaires". Nous considérons l'ensemble des matrices carrées de dimensions $n \times n$ dont les éléments appartiennent à \mathcal{D} . On définit alors $(\mathcal{D}^{n \times n}, \oplus, \otimes)$ comme un dioïde matriciel, avec l'élément neutre (zéro) pour \oplus sous la forme d'une matrice dont tous les coefficients $A_{ij} = \epsilon$. Le produit \otimes a également un élément neutre (identité) noté $I_{\mathcal{D}}$ composé de la manière suivante :

$$A_{ij} = \begin{cases} e & \text{si } i = j, \\ \epsilon & \text{si } i \neq j. \end{cases}$$

Si \mathcal{D} est complet alors $\mathcal{D}^{n \times n}$ l'est aussi.

Opérations sur les matrices

La somme et le produit des matrices de dimensions appropriées, se présentent généralement de la manière suivante :

$$\begin{aligned} \forall A \in \mathcal{D}^{n \times q}, \forall B \in \mathcal{D}^{n \times q} & \quad (A \oplus B)_{ij} = A_{ij} \oplus B_{ij} \\ \forall A \in \mathcal{D}^{n \times q}, \forall B \in \mathcal{D}^{q \times p} & \quad (A \otimes B)_{ij} = \bigoplus_{k=1}^n A_{ik} \otimes B_{kj} \end{aligned}$$

Dans l'algèbre $(\min, +)$, les deux opérations sont définies comme suit :

$$\begin{aligned} (A \wedge B)_{ij} &= A_{ij} \wedge B_{ij} \\ (A \odot B)_{ij} &= \bigwedge_{k=1}^n A_{ik} \odot B_{kj} \end{aligned}$$

L'opérateur \odot est utilisé pour désigner un produit dans l'algèbre $(\min, +)$.

Exemple 1.9 :

Nous donnons ici un exemple d'un produit et une somme de deux matrices carrées dans l'algèbre $(\max, +)$

$$A = \begin{pmatrix} 3 & 7 \\ 4 & 5 \end{pmatrix} \quad B = \begin{pmatrix} 1 & 6 \\ 2 & 8 \end{pmatrix}$$

$$A \oplus B = \begin{pmatrix} \max[3, 1] & \max[7, 6] \\ \max[4, 2] & \max[5, 8] \end{pmatrix} = \begin{pmatrix} 3 & 7 \\ 4 & 8 \end{pmatrix}$$

$$A \otimes B = \begin{pmatrix} \max[3+1, 7+2] & \max[3+6, 7+8] \\ \max[4+1, 5+2] & \max[4+6, 5+8] \end{pmatrix} = \begin{pmatrix} 9 & 15 \\ 7 & 13 \end{pmatrix}$$

On peut définir la multiplication d'une matrice A par une constante α :

$$(\alpha \otimes A)_{ij} = \alpha \otimes A_{ij} = \alpha + A_{ij}$$

En général, la multiplication des matrices dans $\mathcal{D}^{n \times n}$ n'est pas commutative, même si $(\mathcal{D}, \oplus, \otimes)$ est commutative. $\mathcal{D}^{n \times n}$ est distributif si \mathcal{D} l'est. $A \geq B$ dans $\mathcal{D}^{n \times n} \Leftrightarrow \{A_{ij} \geq B_{ij} \text{ dans } \mathcal{D}, i = 1, \dots, n, j = 1, \dots, n\}$

1.4 Éléments de la théorie de la résiduation

La résolution de problèmes qui se résument à l'inversion d'applications dans l'algèbre usuelle, est souvent réalisée grâce à la propriété de bijection de ces applications. En effet, si f une application bijective définie sur $\mathbb{R}, \mathbb{Z}, \dots$, on peut définir une application inverse notée f^{-1} . En revanche, si on considère une application Π définie sur un dioïde, et comme les opérateurs \oplus et \otimes ne sont pas inversibles alors, Π^{-1} ne peut pas être toujours définie directement. C'est dans ce contexte que la théorie de la résiduation intervient comme une alternative à la notion d'inversibilité. Cette théorie concerne uniquement les applications monotones définies sur des ensembles ordonnés.

Après introduction des définitions de base sur la monotonie, les homomorphismes, et la semi-continuité, nous exposerons un ensemble de résultats importants issus de cette théorie. Une exploitation directe de ces résultats va nous permettre d'étudier (existence et calcul) la plus petite et la plus grande solution respectivement des inéquations suivantes :

$$x \leq \Pi(x) \tag{1.4}$$

$$\Pi(x) \leq x \tag{1.5}$$

Le lecteur pourra trouver des études détaillées sur les notions de la résiduation dans : [BJ72], [BCOQ92], [CGQ98], [Cot99] et [Men97].

1.4.1 Homomorphismes de dioïdes, isotonie et continuité

Définition 1.19. (Isotonie, antitonie)

Soit Π une application définie sur des ensembles ordonnés $\Pi : E \longrightarrow F$.

Π est dite isotone si :

$$\forall x, y \in E, \quad x \leq y \implies \Pi(x) \leq \Pi(y)$$

Π est dite antitone si :

$$\forall x, y \in E, \quad x \leq y \implies \Pi(x) \geq \Pi(y)$$

Remarque : Une application est dite monotone si elle est isotone ou antitone.

◇

Définition 1.20. (Homomorphisme)

Une application Π définie d'un dioïde \mathcal{D} vers un dioïde \mathcal{C} est un homomorphisme si $\forall x, y \in \mathcal{D}$

$$\Pi(x \oplus y) = \Pi(x) \oplus \Pi(y) \quad \text{et} \quad \Pi(\varepsilon) = \varepsilon \tag{1.6}$$

$$\Pi(x \otimes y) = \Pi(x) \otimes \Pi(y) \quad \text{et} \quad \Pi(e) = e \tag{1.7}$$

Notons que chaque application qui vérifie la propriété 1.6 sera appelée \oplus -morphisme. Aussi, une application qui ne vérifie que la propriété 1.7 sera appelée \otimes -morphisme. Un homomorphisme est donc \oplus -morphisme et \otimes -morphisme.

Remarque : Si l'application Π est un \oplus -morphisme, alors Π est forcément isotone. En effet, soit $x \leq y$, d'où $x \oplus y = y$ et $\Pi(x \oplus y) = \Pi(y) = \Pi(x) \oplus \Pi(y)$. D'où le résultat $\Pi(x) \leq \Pi(y)$.

Dualement, soit Π est un \wedge -morphisme. Si $x \leq y$, alors $\Pi(x \wedge y) = \Pi(x) = \Pi(x) \wedge \Pi(y)$. On en déduit que $\Pi(x) \leq \Pi(y)$, d'où Π est isotone.

◇

La réciproque est cependant fautive comme le montre le lemme suivant. La démonstration permet de mieux saisir ces équivalences.

Lemme 1.21. [BCOQ92]

Soit Π une application d'un dioïde \mathcal{D} dans un autre dioïde \mathcal{C} . Alors les propositions suivantes sont équivalentes :

1. L'application Π est isotone.
2. L'application Π est \oplus -supermorphisme, c'est à dire :

$$\forall x, y \in \mathcal{D}, \quad \Pi(x \oplus y) \geq \Pi(x) \oplus \Pi(y)$$

3. Si la borne inférieure existe dans \mathcal{D} et \mathcal{C} , Π est un \wedge -sousmorphisme, c'est à dire :

$$\forall x, y \in \mathcal{D}, \quad \Pi(x \wedge y) \leq \Pi(x) \wedge \Pi(y)$$

Preuve :

(1 \Leftrightarrow 2) Supposons que Π est isotone. Puisque x et y sont inférieurs à $x \oplus y$, alors, $\Pi(x)$ et $\Pi(y)$ ainsi que leur borne supérieure $\Pi(x) \oplus \Pi(y)$ sont inférieurs à $\Pi(x \oplus y)$.

Inversement, soit $y \leq x$, ou de façon équivalente, $x = x \oplus y$. Alors, $\Pi(x) = \Pi(x \oplus y) \geq \Pi(x) \oplus \Pi(y)$, ce qui montre que $\Pi(y) \leq \Pi(x)$, d'où Π est isotone. Alors, les deux points 1 et 2 sont équivalents.

(1 \Leftrightarrow 3) Soit Π isotone. Puisque x et y sont supérieurs à $x \wedge y$, alors, $\Pi(x)$ et $\Pi(y)$ ainsi que leur borne inférieure (supposée existante) $\Pi(x) \wedge \Pi(y)$ sont supérieurs à $\Pi(x \wedge y)$.

Inversement, soit $y \geq x$, ou de façon équivalente, $x = x \wedge y$. Alors, $\Pi(x) = \Pi(x \wedge y) \leq \Pi(x) \wedge \Pi(y)$, ce qui montre que $\Pi(y) \geq \Pi(x)$, d'où Π est isotone. D'où l'équivalence entre les deux points 1 et 3.

□

Remarque : Si \mathcal{D} est une chaîne, $x \oplus y$ est égal à x ou y . Par conséquent, $\Pi(x \oplus y)$ est égal à $\Pi(x)$ ou $\Pi(y)$. Si Π est isotone, on peut appliquer le lemme 1.21 et donc $\Pi(x \oplus y) = \Pi(x) \oplus \Pi(y)$, et donc Π est \oplus -morphisme dans ce cas.

Dualement, Π est \wedge -morphisme.

◇

Définition 1.22. (Semi-continuité inférieurement)

Une application Π d'un dioïde complet \mathcal{D} dans un dioïde complet \mathcal{C} est semi-continue inférieurement, en abrégé S.C.I. si, pour tout sous ensemble (fini ou infini) X de \mathcal{D} ,

$$\Pi\left(\bigoplus_{x \in X} x\right) = \bigoplus_{x \in X} \Pi(x) \quad (1.8)$$

Définition 1.23. (Semi-continuité supérieurement)

Une application Π d'un dioïde complet \mathcal{D} dans un dioïde complet \mathcal{C} est semi-continue supérieurement, en abrégé S.C.S. si, pour tout sous ensemble (fini ou infini) X de \mathcal{D} ,

$$\Pi\left(\bigwedge_{x \in X} x\right) = \bigwedge_{x \in X} \Pi(x) \quad (1.9)$$

Nous remarquons que toute application S.C.I. (respectivement S.C.S.) est un \oplus -morphisme (respectivement est un \wedge -morphisme) et donc un \oplus -supermorphisme (respectivement un \wedge -sousmorphisme). D'après le lemme 1.21, Π est isotone.

1.4.2 Applications résiduelles et dualement résiduelles

Nous considérons d'abord des ensembles ordonnés.

Définition 1.24. Application résiduable

Soit Π une application isotone définie d'un ensemble ordonné (\mathcal{D}, \leq) dans un ensemble ordonné (\mathcal{C}, \leq) . L'application Π est dite résiduable, si pour tout $y \in \mathcal{C}$ l'équation $\Pi(x) \leq y$ admet une plus grande solution dans \mathcal{D} . L'application résiduée Π^\sharp est définie par $\Pi^\sharp(y) = \bigoplus\{x \in \mathcal{D} \mid \Pi(x) \leq y\}$.

La définition duale est donnée comme suit :

Définition 1.25. Application dualement résiduable

Soit Π une application isotone définie d'un ensemble ordonné (\mathcal{D}, \leq) dans un ensemble ordonné (\mathcal{C}, \leq) . L'application Π est dite dualement résiduable, si pour tout $y \in \mathcal{C}$ l'équation $\Pi(x) \geq y$ admet une plus petite solution dans \mathcal{D} . L'application résiduée duale Π^\flat est définie par $\Pi^\flat(y) = \bigwedge\{x \in \mathcal{D} \mid \Pi(x) \geq y\}$.

Les démonstrations de l'ensemble des théorèmes 1.26, 1.27, 1.28 et 1.29 sont exposées en détail dans : [BJ72], [BCOQ92], [CGQ98] et [Men97].

Théorème 1.26. Soit Π une application isotone de (\mathcal{D}, \leq) dans (\mathcal{C}, \leq) .

Alors sont équivalents :

1. l'application Π est résiduable.
2. Il existe une application Π^\sharp de \mathcal{C} dans \mathcal{D} isotone telle que

$$\Pi \circ \Pi^\sharp \leq I_{\mathcal{C}} \quad \text{avec } I_{\mathcal{C}} \text{ identité de } \mathcal{C} \quad (1.10)$$

$$\Pi^\sharp \circ \Pi \geq I_{\mathcal{D}} \quad \text{avec } I_{\mathcal{D}} \text{ identité de } \mathcal{D} \quad (1.11)$$

Le théorème 1.27 dual relatif à l'application dualement résiduable est le suivant.

Théorème 1.27. Soit Π une application isotone de (\mathcal{D}, \leq) dans (\mathcal{C}, \leq) . Sont équivalents :

1. l'application Π est dualement résiduable.
2. Il existe une application Π^\flat de \mathcal{C} dans \mathcal{D} isotone telle que

$$\Pi \circ \Pi^\flat \geq I_{\mathcal{C}} \quad \text{avec } I_{\mathcal{C}} \text{ identité de } \mathcal{C} \quad (1.12)$$

$$\Pi^\flat \circ \Pi \leq I_{\mathcal{D}} \quad \text{avec } I_{\mathcal{D}} \text{ identité de } \mathcal{D} \quad (1.13)$$

Unicité de Π^\sharp et Π^\flat

Du fait que l'application Π est résiduable, cela implique que l'application Π^\sharp est unique. Ceci peut être montré par l'absurde en considérant qu'il existe une autre application Ψ qui satisfait le point 2 du théorème 1.26. Nous constatons alors que :

$$\Pi^\sharp = I_{\mathcal{D}} \circ \Pi^\sharp \leq (\Psi \circ \Pi) \circ \Pi^\sharp = \Psi \circ (\Pi \circ \Pi^\sharp) \leq \Psi \circ I_{\mathcal{C}} = \Psi$$

$$\Psi = I_{\mathcal{D}} \circ \Psi \leq (\Pi^{\sharp} \circ \Pi) \circ \Psi = \Pi^{\sharp} \circ (\Pi \circ \Psi) \leq \Pi^{\sharp} \circ I_{\mathcal{C}} = \Pi^{\sharp}$$

$$\implies \Psi = \Pi$$

De façon similaire, on peut montrer que lorsque Π est dualement résiduable, l'application Π^{\flat} est unique.

Les théorèmes 1.28 et 1.29 suivants considèrent des dioïdes.

Théorème 1.28. *Soit Π une application isotone d'un dioïde complet \mathcal{D} dans un dioïde complet \mathcal{C} . Sont équivalents :*

1. *l'application Π est résiduable.*
2. *Π est S.C.I. et $\Pi(\varepsilon) = \varepsilon$*

Le théorème dual de 1.28 est le suivant.

Théorème 1.29. *Soit Π une application isotone d'un dioïde complet \mathcal{D} dans un dioïde complet \mathcal{C} . Sont équivalents :*

1. *Π est une application dualement résiduable.*
2. *Π est S.C.S. et $\Pi(T) = T$*

Le théorème suivant donne quelques propriétés existant entre des applications résiduables et leurs résiduées ainsi que pour les applications dualement résiduables et leurs résiduées duales.

Théorème 1.30. [BCOQ92]

1. *Si les deux applications $\Pi : \mathcal{D} \rightarrow \mathcal{D}$ et $\Phi : \mathcal{D} \rightarrow \mathcal{D}$ sont résiduables avec \wedge existe dans un dioïde complet \mathcal{D} , alors $\Pi \oplus \Phi$ est résiduable*

$$(\Pi \oplus \Phi)^{\sharp} = \Pi^{\sharp} \wedge \Phi^{\sharp} \tag{1.14}$$

2. *Si les deux applications $\Pi : \mathcal{D} \rightarrow \mathcal{D}$ et $\Phi : \mathcal{D} \rightarrow \mathcal{D}$ sont dualement résiduables avec \wedge existe dans un dioïde complet \mathcal{D} , alors $\Pi \wedge \Phi$ est aussi dualement résiduable*

$$(\Pi \wedge \Phi)^{\flat} = \Pi^{\flat} \oplus \Phi^{\flat} \tag{1.15}$$

1.4.2.1 Résiduation de l'addition et de la multiplication

Les applications T_a (translation), L_a (multiplication à gauche) et R_a (multiplication à droite) sur un dioïde \mathcal{D} complet sont définies comme suit :

$$T_a : x \mapsto a \oplus x \tag{1.16}$$

$$L_a : x \mapsto a \otimes x \tag{1.17}$$

$$R_a : x \mapsto x \otimes a \quad (1.18)$$

Les références [BJ72], [BCOQ92] et [Gau92] proposent plus de détails sur les propriétés relatives aux applications : L_a^\sharp , R_a^\sharp et T_a^\flat .

Le fait que le dioïde \mathcal{D} soit complet implique que les applications L_a et R_a sont semi-continues inférieurement (distributivité de \otimes à gauche et à droite sur \oplus pour des ensembles finis ou infinis). En outre, comme ε est absorbant pour \otimes , alors $L_a(\varepsilon) = a \otimes \varepsilon = \varepsilon$ et $R_a(\varepsilon) = \varepsilon \otimes a = \varepsilon$. D'après le théorème 1.28, les applications L_a et R_a sont résiduables. On peut donc considérer L_a^\sharp et R_a^\sharp les applications résiduées respectivement de L_a et R_a qui sont définies par :

$$L_a^\sharp(b) = \bigoplus \{x \mid L_a(x) = a \otimes x \leq b\} = \frac{b}{a} = a \setminus b$$

$$R_a^\sharp(b) = \bigoplus \{x \mid R_a(x) = x \otimes a \leq b\} = \frac{b}{a} = a / b$$

A l'opposé, T_a n'est pas en général résiduable car la condition $T_a(\varepsilon)$ n'est pas vérifiée (sauf pour $a = \varepsilon$). Cependant, si le dioïde \mathcal{D} est distributif, alors l'application T_a est semi-continue supérieurement. De plus, $T_a(T) = a \oplus T = T$, d'où T_a est dualement résiduée. La résiduée duale de T_a est donnée par :

$$T_a^\flat(b) = \bigwedge \{x \mid T_a(x) = a \oplus x \geq b\}$$

le théorème qui suit regroupe un ensemble de propriétés concernant les applications L_a et R_a qui seront utiles dans la suite de ce mémoire.

Théorème 1.31.

$$\begin{array}{lll} a \setminus (x \wedge y) = a \setminus x \wedge a \setminus y & f.1 & (x \wedge y) / a = x / a \wedge y / a \\ a \setminus (x \oplus y) \geq a \setminus x \oplus a \setminus y & f.2 & (x \oplus y) / a \geq x / a \oplus y / a \\ (a \oplus b) \setminus x = a \setminus x \wedge b \setminus x & f.3 & x / (a \oplus b) = x / a \wedge x / b \\ (a \wedge b) \setminus x \geq a \setminus x \oplus b \setminus x & f.4 & x / (a \wedge b) \geq x / a \oplus x / b \\ a(a \setminus x) \leq x & f.5 & (x / a)a \leq x \\ a \setminus (ax) \geq x & f.6 & (xa) / a \geq x \\ a(a \setminus (ax)) = ax & f.7 & ((xa) / a)a = xa \\ a \setminus (a(a \setminus x)) = a \setminus x & f.8 & ((x / a)a) / a = x / a \\ (ab) \setminus x = b \setminus (a \setminus x) & f.9 & x / (ba) = (x / a) / b \\ (a \setminus x) / b = a \setminus (x / b) & f.10 & b \setminus (x / a) = (b \setminus x) / a \\ b(a \setminus x) \leq (a / b) \setminus x & f.11 & (x / a)b \leq x / (b \setminus a) \end{array}$$

$$(a \setminus x)b \leq a \setminus (xb) \quad f.12 \quad b(x/a) \leq (bx)/a$$

$$(a \setminus x) \oplus b \leq a \setminus (x \oplus ab) \quad f.13 \quad (x/a) \oplus b \leq (x \oplus ba)/a$$

Dans le dioïde $\overline{\mathbb{R}}_{max}$, on peut noter que dans le cas où deux éléments d'un dioïde a et b sont scalaires et finis, alors une résiduation à gauche $a \setminus b$ ou à droite b/a se résume à une soustraction dans l'algèbre usuelle $b - a$. Néanmoins, ceci ne couvre pas les cas avec a ou b infini. Les règles de manipulation sont données comme suit :

$$\begin{aligned} a \setminus b &= b - a && \text{si } a \text{ et } b \text{ sont finis} \\ a \setminus T &= T && \text{pour tout } a \\ a \setminus \varepsilon &= \varepsilon && \text{pour tout } a \text{ fini} \\ \varepsilon \setminus a &= T && \text{pour tout } a \\ T \setminus a &= \varepsilon && \text{pour } a \neq T \end{aligned}$$

Remarquons aussi que $T \otimes \varepsilon = \varepsilon = -\infty$ alors que $\varepsilon \setminus \varepsilon = T$.

On note que ces règles sont valables aussi dans le cas où l'on a une résiduation à droite $/$. Il en résulte les mêmes règles d'opérations dans le dioïde $\overline{\mathbb{R}}_{min}$ avec $\varepsilon = +\infty$ et $T = -\varepsilon$.

On peut étendre la relation $a \setminus b = b - a$ aux valeurs infinies si l'on prend la convention $+\infty - \infty = +\infty$

1.4.2.2 Résiduation dans le cas matriciel

Tous les résultats et propriétés présentés précédemment restent valables dans le cas matriciel à condition que les dimensions des matrices soient compatibles. On considère l'application L_A donnée par

$$\begin{aligned} L_A : \mathcal{D}^{p \times q} &\rightarrow \mathcal{D}^{n \times q} \\ X &\mapsto A \otimes X \quad (A \in \mathcal{D}^{n \times p}) \end{aligned}$$

Théorème 1.32. Soit $B \in \mathcal{D}^{n \times q}$. La plus grande solution de $Ax \leq B$ est la matrice $L_A^\sharp(B) \in \mathcal{D}^{p \times q}$, que l'on peut noter $\frac{B}{A} = A \setminus B$.

$$(A \setminus B)_{ij} = \bigwedge_{k=1}^n A_{ki} \setminus B_{kj}$$

Preuve : Les propositions suivantes sont équivalentes :

$$A \otimes X \leq B$$

$$(A \otimes X)_{kj} = \bigoplus_{i=1}^p (A_{ki} X_{ij}) \leq B_{kj} \quad \forall k \in [1, n], \forall j \in [1, q]$$

$$A_{ki} X_{ij} \leq B_{kj} \quad \forall k \in [1, n], \forall j \in [1, q], \forall i \in [1, p]$$

$$X_{ij} \leq A_{ki} \setminus B_{kj}$$

$$X_{ij} \leq \bigwedge_{k=1}^n A_{ki} \setminus B_{kj} \quad \forall j \in [1, q], \forall i \in [1, p]$$

□

De même pour l'application

$$\begin{aligned} R_A : \mathcal{D}^{q \times n} &\rightarrow \mathcal{D}^{q \times p} \\ X &\mapsto X \otimes A \quad (A \in \mathcal{D}^{n \times p}) \end{aligned}$$

Soit $B \in \mathcal{D}^{q \times p}$, alors :

$$(B/A)_{ij} = \bigwedge_{k=1}^p B_{ik}/A_{jk}$$

Remarque : Nous remarquons d'une part que $(A \setminus B)_{ij} = \bigwedge_{k=1}^n A_{ki} \setminus B_{kj} = \bigwedge_{k=1}^n (B_{kj} - A_{ki}) = \bigwedge_{k=1}^n (B_{kj} + (-A_{ik}^t))$ avec A^t la transposée de A et la convention ci-dessus. D'autre part, $(A \odot B)_{ij} = \bigwedge_{k=1}^n A_{ik} \odot B_{kj} = \bigwedge_{k=1}^n A_{ik} + B_{kj}$, d'où le résultat suivant :

$$A \setminus B = (-A^t) \odot B$$

De la même manière, on peut vérifier que :

$$A \setminus' B = (-A^t) \otimes B$$

◇

1.5 Théorie du point fixe

1.5.1 Propriétés des équations de type point fixe

On s'intéresse dans cette partie à l'étude de la résolution des équations au point fixe suivantes :

$$\Pi(x) = x \tag{1.19}$$

$$\Pi(x) \leq x \tag{1.20}$$

$$\Pi(x) \geq x \tag{1.21}$$

avec Π une application isotone de \mathcal{D} vers \mathcal{D} .

Les principaux résultats proposés ici, proviennent des références suivantes : [KG94], [KG95], et [BCOQ92].

Soient les sous ensembles suivants :

$$E_{\Pi} = \{x \mid \Pi(x) = x\}, \quad U_{\Pi} = \{x \mid \Pi(x) \leq x\}, \quad L_{\Pi} = \{x \mid \Pi(x) \geq x\} \tag{1.22}$$

Le théorème suivant de Knaster et Tarski montre que chaque application définie sur un treillis complet a un plus petit point fixe et un plus grand point fixe sous condition d'isotonie. Comme un dioïde complet a une structure de treillis complet, les résultats énoncés dans ce théorème 1.33 s'appliquent aussi dans le cas où \mathcal{D} est un dioïde complet.

Théorème 1.33. [KG94]

Soit une application $\Pi : \mathcal{D} \rightarrow \mathcal{D}$ isotone tel que (\mathcal{D}, \leq) est un treillis complet. Alors on a

1. $\inf E_\Pi \in E_\Pi$ et $\inf E_\Pi = \inf U_\Pi$
2. $\sup E_\Pi \in E_\Pi$ et $\sup E_\Pi = \sup L_\Pi$

Preuve : Comme \mathcal{D} est un treillis complet, $\sup \mathcal{D} \in \mathcal{D}$. La définition du suprémal entraîne que $\Pi(\sup \mathcal{D}) \leq \sup \mathcal{D}$, c'est-à-dire $\sup \mathcal{D} \in U_\Pi$, ce qui implique que U_Π est non vide. Aussi, puisque \mathcal{D} est un treillis complet, alors $\inf U_\Pi \in \mathcal{D}$. On a par définition $E_\Pi \subseteq U_\Pi$. Par conséquent $\inf U_\Pi \leq \inf E_\Pi$. Il reste à montrer que $\inf E_\Pi \leq \inf U_\Pi$. Pour cela, il suffit de montrer que $\inf U_\Pi$ est un point fixe de Π ($\Pi(\inf U_\Pi) = \inf U_\Pi$) puisque $\inf E_\Pi$ est le plus petit point fixe de Π .

D'une part, par définition on a $\inf U_\Pi \leq z$ pour tout $z \in U_\Pi$. L'isotonie de Π donne $\Pi(\inf U_\Pi) \leq \Pi(z)$ pour tout $z \in U_\Pi$. On a aussi $\Pi(z) \leq z$ qui montre que $\Pi(\inf U_\Pi) \leq z$ pour tout $z \in U_\Pi$, d'où

$$\Pi(\inf U_\Pi) \leq \inf U_\Pi$$

On déduit par l'isotonie de Π que $\Pi(\Pi(\inf U_\Pi)) \leq \Pi(\inf U_\Pi)$. D'après la définition de U_Π , on a donc $\Pi(\inf U_\Pi) \in U_\Pi$. La définition du infimum implique :

$$\inf U_\Pi \leq \Pi(\inf U_\Pi)$$

D'où le résultat donné dans le premier point du théorème. La démonstration du point 2 du théorème est le dual du point 1. □

Nous définissons Π^k comme suit $\Pi^k = \underbrace{\Pi \circ \Pi \circ \dots \circ \Pi}_{k \text{ fois}}$ avec $\Pi^0 = I_{\mathcal{D}}$ et nous introduisons les notations suivantes :

$$\Pi^* = \bigoplus_{k \in \mathbb{N}} \Pi^k \quad \Pi_* = \bigwedge_{k \in \mathbb{N}} \Pi^k \quad (1.23)$$

avec l'opérateur “ $*$ ” est appelé étoile de Kleene. Un autre opérateur noté “ $+$ ” dérivé de “ $*$ ” est défini par :

$$\Pi^+ = \bigoplus_{k=1}^{+\infty} \Pi^k \quad \Pi_+ = \bigwedge_{k=1}^{+\infty} \Pi^k \quad (1.24)$$

Théorème 1.34. [BCOQ92] Soit \mathcal{D} un dioïde complet.

1. Soient deux applications Π et Ψ . Si $\Pi \geq \Psi$, alors $U_\Pi \subseteq U_\Psi$.
2. Si $\mathcal{C} \in U_\Pi$, alors $\bigwedge_{x \in \mathcal{C}} x \in U_\Pi$; autrement dit, l'ensemble U_Π muni de la relation d'ordre induit par \mathcal{D} est un inf-demi-treillis complet ayant la même opération \wedge que \mathcal{D} . De plus,

$T \in U_{\Pi}$. Par conséquent (par le théorème 1.9), U_{Π} est aussi un treillis complet, mais l'opération qui donne le maximum n'a pas besoin d'être identique à celle de \mathcal{D} et est notée \oplus .

3. Si Π est S.C.I., alors $\mathcal{C} \subseteq U_{\Pi}$ implique $\bigoplus_{x \in \mathcal{C}} x \in U_{\Pi}$; autrement dit, l'ensemble U_{Π} muni de la relation d'ordre induit par \mathcal{D} est un sup-demi-treillis complet ayant la même opération \oplus que \mathcal{D} .

4. La proposition 3 est vraie aussi pour E_{Π} .

5. En général, $U_{\Pi} = U_{\Pi^*} = E_{\Pi^*}$. Autrement dit,

$$\Pi(x) \leq x \Leftrightarrow \Pi^*(x) \leq x \Leftrightarrow \Pi^*(x) = x \quad (1.25)$$

6. Si Π est S.C.I., alors $U_{\Pi} = \Pi^*(\mathcal{D})$. Le plus petit élément est $\Pi^*(\varepsilon)$ qui appartient aussi à E_{Π} . Ainsi, il est aussi le plus petit élément pour ce sous ensemble.

Le lecteur intéressé par la démonstration complète de ce théorème pourra consulter [BCOQ92].

Le théorème suivant est le dual au théorème 1.34.

Théorème 1.35.

1. Si $\Pi \geq \Psi$, alors $L_{\Psi} \subseteq L_{\Pi}$

2. Si $\mathcal{C} \in L_{\Pi}$, alors $\bigoplus_{x \in \mathcal{C}} x \in L_{\Pi}$; autrement dit, l'ensemble L_{Π} muni de la relation d'ordre induit par \mathcal{D} est un sup-demi-treillis complet ayant la même opération \oplus que \mathcal{D} . De plus, $\varepsilon \in L_{\Pi}$. Par conséquent (par le théorème 1.9), L_{Π} est aussi un treillis complet, mais l'opération qui donne le minimum n'a pas besoin d'être identique à celle de \mathcal{D} et est notée \wedge .

3. Si Π est S.C.S., alors $\mathcal{C} \subseteq L_{\Pi}$ implique $\bigwedge_{x \in \mathcal{C}} x \in L_{\Pi}$; autrement dit, l'ensemble L_{Π} muni de la relation d'ordre induit par \mathcal{D} est un inf-demi-treillis complet ayant la même opération \wedge que \mathcal{D} .

4. La proposition 3 est vraie aussi pour E_{Π}

5. En général, $L_{\Pi} = L_{\Pi^*} = E_{\Pi^*}$

6. Si Π est S.C.S., alors $L_{\Pi} = \Pi_*(\mathcal{D})$. Le plus grand élément est $\Pi_*(T)$ qui appartient aussi à E_{Π} . Ainsi, il est aussi le plus grand élément pour ce sous ensemble.

Application à la résolution des équations linéaires dans les dioïdes

Les deux corollaires suivants se déduisent des théorèmes 1.34 et 1.35. Il suffit de remarquer que l'application $x \rightarrow A \otimes x \oplus B$ est S.C.I. et d'appliquer alors le point 6 du théorème 1.34. De même, l'application $x \rightarrow A \odot x \wedge B$ est S.C.S., d'où on peut appliquer le point 6 du théorème 1.35.

Corollaire 1.36. *Soit A et B deux matrices dans un dioïde complet. Nous considérons l'équation $x = A \otimes x \oplus B$ et l'inéquation*

$$x \geq A \otimes x \oplus B$$

Les deux équations admettent $A^ \otimes B$ comme plus petite solution et chaque solution x vérifie $x = A^* \otimes x$. L'élément T est la plus grande solution de l'inéquation.*

Corollaire 1.37. *Soit A et B deux matrices dans un dioïde complet et soit l'équation $x = A \odot x \wedge B$ et l'inéquation*

$$x \leq A \odot x \wedge B$$

Les deux équations admettent $A_ \otimes B$ comme plus grande solution et chaque solution x vérifie $x = A_* \odot x$. L'élément ε est la plus petite solution de l'inéquation.*

Le théorème suivant fournit une méthode de calcul du plus petit et plus grand point fixe sous certaines conditions. Notons aussi que ce résultat généralise les points 6 des théorèmes 1.34 et 1.35 en raison de la définition moins stricte de la semi-continuité que nous présentons.

Définition 1.38. [KG95]

Soit un ensemble ordonné (E, \leq) . Une application $\Pi : E \rightarrow E$ est dite semi-continue inférieurement sur chaîne si pour toute chaîne croissante $\{y_i, i \geq 0\} \subseteq E$:

$$\Pi\left(\bigvee_{i \geq 0} y_i\right) = \bigvee_{i \geq 0} \Pi(y_i)$$

Elle est dite semi-continue supérieurement sur chaîne si pour toute chaîne décroissante $\{y_i, i \geq 0\} \subseteq E$:

$$\Pi\left(\bigwedge_{i \geq 0} y_i\right) = \bigwedge_{i \geq 0} \Pi(y_i)$$

Théorème 1.39. [KG94]

Soit (E, \leq) un treillis complet et $\Pi : E \rightarrow E$ une application. Alors on a :

1. *si Π est semi-continue inférieurement sur chaîne, alors $\inf E_\Pi = \Pi^*(\inf E_\Pi)$,*
2. *si Π est semi-continue supérieurement sur chaîne, alors $\sup E_\Pi = \Pi_*(\sup E_\Pi)$.*

Preuve : Pour cette démonstration, on pose $\inf E_\Pi = x_0$. Dans un premier temps, on montre que $\{\Pi^i(x_0), i \geq 0\}$ est une chaîne croissante. Comme l'application Π est semi-continue inférieurement sur chaîne, alors elle est isotone. Par conséquent, $\inf E_\Pi \in E_\Pi$ d'après le théorème 1.33. On sait que :

$$\Pi^0(x_0) = x_0 \leq \Pi(x_0) = \Pi^1(x_0)$$

Par conséquent, l'isotonie de Π implique que $\Pi^i(x_0) \leq \Pi^{i+1}(x_0)$ pour chaque $i \geq 0$.

Dans un deuxième temps, on prouve que $\Pi^*(x_0)$ est un point fixe (c'est-à-dire $\Pi(\Pi^*(x_0)) = \Pi^*(x_0)$).

$$\begin{aligned} \Pi(\Pi^*(x_0)) &= \Pi\left(\bigoplus_{i \geq 0} \Pi^i(x_0)\right) \\ &= \bigoplus_{i \geq 0} (\Pi^{i+1}(x_0)) \quad (\text{car } \Pi \text{ est semi-continue inférieurement sur chaîne}) \\ &= \bigoplus_{i \geq 1} (\Pi^i(x_0)) \\ &= \bigoplus_{i \geq 0} (\Pi^i(x_0)) \quad (\text{puisque } \Pi^0(x_0) = x_0 \leq \Pi^i(x_0)) \\ &= \Pi^*(x_0) \end{aligned}$$

On montre maintenant que $\Pi^*(x_0)$ est la plus petite solution, autrement dit $\Pi^*(x_0) \leq y$ pour tout $y \in E_\Pi$. Soit $y \in E_\Pi$, par définition on a $x_0 \leq y$. Comme Π est isotone, $x_0 \leq y$ implique :

$$\forall i \geq 0, \Pi^i(x_0) \leq \Pi^i(y) = y \text{ du fait que } y \text{ est un point fixe}$$

On conclut que $\Pi^*(x_0) = \bigoplus_{i \geq 0} \Pi^i(x_0) \leq y$.

La démonstration du deuxième point est analogue (duale) à celle faite pour le point 1. □

1.5.2 Un algorithme de calcul du point fixe

Dans cette partie, nous allons présenter un algorithme itératif classique [MD92], [Wal95] et [WB98] qui permet de calculer efficacement la plus grande solution de l'équation de type point fixe suivante : $x \leq f(x)$

Définition 1.40. Une contrainte bornant par dessus (CBD) est une inéquation linéaire qui se met sous la forme : $x_k \leq a_1x_1 \oplus a_2x_2 \oplus \dots \oplus a_nx_n$. Dans un système qui contient n variables $\{x_1, x_2, \dots, x_n\}$ et m CBD, la i ème CBD a la forme:

$$x_i \leq \bigoplus_{j=1}^n a_{ij} \otimes x_j \oplus b_i \tag{1.26}$$

la variable x_i est appelée variable cible.

Remarque : Dans le système 1.26 donné dans la définition 1.40, l'indice i prend des valeurs allant de 1 à m . Plusieurs CBD peuvent avoir alors la même variable cible x_i .

◇

L'algorithme général de Mc Millan et Dill (MD) opère sur un système d'inéquations CBD dont la forme est donnée par 1.26. Nous donnons maintenant cet algorithme.

Algorithme général de Mc Millan et Dill (MD) :

Début

Initialisation : $\forall i$ allant de 1 à n , $x_i \leftarrow T$

Répéter tant que les valeurs de x_i ne sont pas constantes

Pour chaque contrainte $x_i \leq \bigoplus_{j=0}^n a_{ij} \otimes x_j \oplus b_i$, faire :

Si $x_i > \bigoplus_{j=0}^n a_{ij} \otimes x_j \oplus b_i$, alors $x_i \leftarrow \bigoplus_{j=0}^n a_{ij} \otimes x_j \oplus b_i$

Écrire les valeurs de x_i

Fin

Pour bien expliquer le fonctionnement de l'algorithme général MD, nous donnons l'exemple suivant d'un système d'inéquations 1.27 dont nous cherchons la solution maximale.

$$\left\{ \begin{array}{l} x_1 \leq 9 \\ x_1 \leq x_2 \\ x_2 \leq x_3 \oplus 7 \\ x_3 \leq (x_1 - 2) \oplus (x_2 - 1) \\ x_3 \leq 10 \end{array} \right. \quad (1.27)$$

Le tableau suivant regroupe toutes les valeurs données par l'algorithme MD dans chaque itération.

	k=0	k=1	k=2	k=3	k=4	k=5	k=6
x_1	T	9	9	9	9	8	7
x_2	T	T	10	9	8	7	7
x_3	T	10	9	8	7	6	6

Dans l'exemple du système 1.27, nous pouvons remarquer que la convergence dépend des valeurs des constantes a_{ij} liées à chaque contrainte et du nombre des variables constituant le système $n = 3$. En général, la complexité de l'algorithme MD est pseudo-polynomial [MD92]. Cet algorithme présente des limitations [Wal95]. En effet, l'algorithme MD pose des problèmes de démarrage car dans certains cas, la solution finale fournie reste bloquée à sa valeur d'initialisation T . Ensuite, cet algorithme pose un problème de convergence dans sa version de base. En effet, si une des solution est égale à $\varepsilon = -\infty$ le nombre d'itérations tend vers l'infini.

L'algorithme de Mc Millan et Dill peut être globalisé sous la forme plus concise suivante, parfois appelé algorithme de Kleene. Le terme $f(x)$ désigne ici une fonction (min, max, +).

Début

Initialisation : $\forall i$ allant de 1 à n , $x_i \leftarrow T$

Tant que $x_{k+1} \neq x_k$

$$x_{k+1} \leftarrow x_k \wedge f(x_k)$$

Écrire les valeurs de x_i

Fin

1.6 Fonctions topicales et théorie spectrale

Dans cette dernière partie, nous donnons des notions de base concernant les fonctions topicales, les fonctions (min, max, +) et la théorie spectrale relatives à ces fonctions. Le vecteur spectral sera en particulier utilisé ultérieurement. Pour plus d'informations, le lecteur pourra se référer aux documents suivants : [CTCG⁺98], [GG98b], [GG99], [BM03], [Gav02], [Wou01] et [WS03]. Rappelons que dans le cadre de l'étude des fonctions topicales, on se place dans \mathbb{R} qui n'est pas un treillis complet.

1.6.1 Fonctions topicales

Dorénavant, un vecteur dans \mathbb{R}^n sera noté x , et x_i représentera la i -ème composante du vecteur x .

Notons ici qu'on peut utiliser un vecteur et un scalaire dans la même formule : il suffit d'appliquer une opération ou relation scalaire sur chacune des composantes du vecteur. Par exemple, $x + h$ avec x vecteur et h scalaire veut dire qu'on ajoute h sur toutes les composantes x_i . Par similarité, $x \leq h$ est équivalent à $x_i \leq h$ pour tout $1 \leq i \leq n$.

Définition 1.41. Soit $x \in \mathbb{R}^n$, nous définissons le sup t et le min b d'un vecteur x comme suit :

$$t(x) = x_1 \oplus x_2 \cdots \oplus x_n$$

$$b(x) = x_1 \wedge x_2 \cdots \wedge x_n$$

Le formulaire suivant donne des relations élémentaires que vérifient les fonctions t et b . Son exploitation permet d'établir la proposition 1.43.

$$\begin{aligned}
b(x) &= -t(-x) \\
x &\leq t(x) \\
t(\lambda x) &= \lambda t(x) \text{ si } \lambda \geq 0 \\
t(x) &\leq t(y) \text{ si } x \leq y \\
t(x+h) &= t(x) + h \\
t(x+y) &\leq t(x) + t(y) \\
t(x \oplus y) &= t(x) \oplus t(y) \\
t(x \wedge y) &\leq t(x) \wedge t(y) \\
\|t(x)\| &= t(x) \oplus -b(x)
\end{aligned} \tag{1.28}$$

La notation $\|x\|$ fait référence à la norme l_∞ définie dans \mathbb{R}^n par :

$$\|x\| = \|x_1\| \oplus \|x_2\| \oplus \dots \oplus \|x_n\|$$

Les définitions suivantes permettent de caractériser les fonctions topicales.

Définition 1.42. Soit F et $x, y \in \mathbb{R}^n$ et $h \in \mathbb{R}$.

- 1• **Monotonie:** si $x \leq y$ alors $F(x) \leq F(y)$ (M).
- 2• **Homogénéité:** $F(x+h) = F(x) + h$ (H).
- 3• **Non expansivité de F :** $\|F(x) - F(y)\| \leq \|x - y\|$ (N).
- 4• **Non expansivité de t :** $t(F(x) - F(y)) \leq t(x - y)$ (T).
- 5• **Non contractivité de b :** $b(F(x) - F(y)) \geq b(x - y)$ (B).

Proposition 1.43. [GK95]

Soit $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$, les propriétés suivantes sont équivalentes :

- (i) F vérifie (H) et (M)
- (ii) F vérifie (H) et (N)
- (iii) F vérifie (B)
- (iv) F vérifie (T)

Définition 1.44. Fonctions topicales

Une fonction $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$ est dite topicale si elle vérifie les propriétés d'homogénéité (H) et de monotonie (M).

L'ensemble des fonctions topicales de $\mathbb{R}^n \rightarrow \mathbb{R}^n$ sera noté $\mathbf{T}_p(\mathbb{R}^n, \mathbb{R}^n)$.

A partir de la définition 1.44 et de la proposition 1.43, on peut donc dire qu'une fonction $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$ est topicale si elle vérifie l'une des propriétés de la proposition 1.43.

Le calcul du temps de cycle d'une fonction topicale est donné par la définition suivante. Si le système peut être modélisé par une unique fonction $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$, le vecteur spectral ou temps de cycle donnera une image de sa performance.

Définition 1.45. Vecteur spectral ou temps de cycle d'une fonction topicale

Soit une fonction topicale $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$ vérifiant **(N)**. Le vecteur spectral ou temps de cycle de F , noté $\chi(F) \in \mathbb{R}^n$, s'il existe, est défini par :

$$\chi(F) = \lim_{k \rightarrow +\infty} \frac{F^k(x)}{k} \quad (1.29)$$

Notons que le temps de cycle est un vecteur. L'appellation vecteur spectral ou vecteur temps de cycle est ainsi plus explicite. Cette remarque sera exploitée dans le chapitre 3 sur la vivacité.

Remarque : La limite n'existe pas toujours et un problème ouvert est de caractériser les fonctions topicales pour qui elle existe.

◇

Définition 1.46. Soit une fonction topicale $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$ et $x \in \mathbb{R}^n$. On dit que x est un vecteur propre de F , si $F(x) = x + h$ pour $h \in \mathbb{R}$. On dit aussi que x est un point périodique de F de période p , si x est un vecteur propre de F^p , mais pas de F^k pour tout $0 < k < p$.

La définition suivante généralise la d-cyclicité des matrices $(\max, +)$.

Définition 1.47. Le couple $(\eta, v) \in (\mathbb{R}^n)^2$ est un régime affine ultime pour F , s'il existe un entier K tel que :

$$\forall k \geq K, F(v + k\eta) = v + (k + 1)\eta$$

Lemme 1.48. [CT01]

Soit une fonction topicale $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$ satisfaisant **(N)**. Si $\lim_{k \rightarrow +\infty} \frac{F^k(x)}{k}$ existe quelque part alors elle existe partout et a une valeur unique.

Proposition 1.49. [CT01]

Si une fonction topicale possède un vecteur propre, alors le vecteur spectral existe et il est indépendant du vecteur propre.

Preuve : Supposons que f est une fonction topicale possédant un vecteur propre tel que : $F(x) = x + h$. Par itération, nous arrivons à $F^k(x) = x + k.h$. De là ($\lim_{k \rightarrow +\infty} \frac{x + kh}{k} = h$), $\chi(F)$ existe et $\chi(F) = h$. Cela veut dire que chaque composante de $\chi(F)$ a la même valeur h . Or le lemme 1.48 stipule que cette valeur est unique. Elle est donc indépendante du choix du vecteur propre.

□

Nous définissons maintenant les fonctions de type $(\min, \max, +)$ qui font partie des fonctions topicales.

1.6.2 Fonctions (min, max, +)

1.6.2.1 Définitions

Définition 1.50. *Fonction (min, max, +) de type (n, 1)*

Une fonction (min, max, +) de type (n, 1) est une fonction $f : \mathbb{R}^n \rightarrow \mathbb{R}^1$, qui peut être écrite comme un mot dans la grammaire suivante :

$$f = x_1, x_2, \dots, x_n \mid f + a \mid f \wedge f \mid f \oplus f \quad (1.30)$$

où a est un réel arbitraire ($a \in \mathbb{R}$)

Les barres verticales séparent la construction récursive des termes. Cela signifie qu'on construit une fonction (min, max, +) $f : \mathbb{R}^n \rightarrow \mathbb{R}^1$ en appliquant un nombre fini d'opérations max et min sur chaque composante de la forme $x_i + a$ où $1 \leq i \leq n$ et $a \in \mathbb{R}$.

Définition 1.51. *Fonction (min, max, +) de type (n, m)*

Une fonction (min, max, +) de type (n, m) est toute fonction $F : \mathbb{R}^n \rightarrow \mathbb{R}^m$, telle que chaque composante F_i est une fonction (min, max, +) de type (n, 1).

L'ensemble des fonctions (min, max, +) de type (n, m) sera noté $D^*(n, m)$ ou D^* .

Les fonctions (min, max, +) de type (n, m) comprennent des applications max-plus linéaires sous la forme :

$$g_i(x) = \bigoplus_{1 \leq j \leq n} (A_{ij} + x_j)$$

Par dualité, nous pouvons définir des applications min-plus linéaires qui sont incluses dans les fonctions (min, max, +) de type (n, m) :

$$h_i(x) = \bigwedge_{1 \leq j \leq n} (B_{ij} + x_j)$$

Les matrices A et B sont deux matrices de dimensions $(n \times n)$. Les éléments de A appartiennent à $\mathbb{R} \cup \{-\infty\}$. En revanche, les coefficients de B sont dans $\mathbb{R} \cup \{+\infty\}$.

Exemple 1.10 : .

Nous considérons trois fonctions F^1 , F^2 et F^3 dans $D^*(2, 2)$ définies par:

$$\begin{array}{lll} F_1^1(x_1, x_2) = x_1 + 2 & F_1^2(x_1, x_2) = x_1 + 2 & F_1^3(x_1, x_2) = x_2 - 2 \\ F_2^1(x_1, x_2) = x_2 + 3 & F_2^2(x_1, x_2) = x_1 + 4 & F_2^3(x_1, x_2) = (x_2 + 3) \wedge (x_1 + 4) \end{array}$$

Si une fonction est à la fois max-plus et min-plus, alors elle est dite **simple**. Pour que $F \in D^*(n, m)$ soit simple, il faut que chaque composante F_i soit simple. Par exemple, les fonctions F^1 et F^2 sont simples. De plus, F^3 est min-plus (min-seulement) et $F^1 \oplus F^2$ est max-plus (max-seulement).

Si maintenant $F \in D^*(n, m)$ telle que chaque F_i est max-plus ou min-plus, alors F est dite **séparée**. Ainsi, $(F^1 \wedge F^2) \oplus F^3$ est séparée.

1.6.2.2 Vecteur propre et point fixe des fonctions (min, max, +)

Nous présentons maintenant les propriétés relatives au vecteur spectral [CTCG⁺98], [CT01], [GG99], [Gun94a] et [GK95].

Tous les résultats suivants se basent sur le vecteur spectral et nécessitent ainsi qu'il existe toujours. Le corollaire suivant lève toute ambiguïté sur ce point.

Corollaire 1.52. [CTGG99] *Chaque fonction dans $\mathcal{D}^*(\mathbb{R}^n, \mathbb{R}^n)$ a un vecteur spectral. De plus, $\chi(F) = \eta$, pour chaque régime affine ultime $(\eta, v) \in (\mathbb{R}^n)^2$ de F .*

Les résultats qui seront donnés par la suite nécessitent l'introduction de la représentation rectangulaire des fonctions (min, max, +).

Définition 1.53. Max-représentation Soit $F \in \mathcal{D}^*(n, m)$. Un sous-ensemble $S \subseteq \mathcal{D}^*(n, m)$ est appelé max-représentation de F si S est un ensemble fini de fonctions (max, +) tel que :

$$F(x) = \bigwedge_{H \in S} H(x) \quad (1.31)$$

Dualement, on appelle un sous-ensemble T min-représentation de F si T est un ensemble fini de fonctions (min, +) tel que :

$$F(x) = \bigoplus_{G \in T} G(x) \quad (1.32)$$

Supposons que $\chi(F)$ existe et soit $H \in S$, $F \leq H$. En appliquant la propriété de monotonie (**M**), $\chi(F) \leq \chi(H)$. La max-représentation permet d'évaluer une borne supérieure du vecteur temps de cycle. Il est donné par :

$$\chi(F) \leq \bigwedge_{H \in S} \chi(H) \quad (1.33)$$

Or, une fonction donnée de type (min, max, +) admet plusieurs max-représentations. On aboutit donc par l'application de 1.33 à différentes bornes supérieures du vecteur temps de cycle correspondant à cette fonction. Il est donc nécessaire de considérer l'ensemble des représentations (max, +) possibles. C'est ce que fait la rectangularisation suivante.

Tout d'abord, on constate que l'ensemble des fonctions (min, max, +) $\mathcal{D}^*(n, m)$, admet aussi comme représentation :

$$D^*(n, m) = \underbrace{D^*(n, 1) \times \cdots \times D^*(n, 1)}_{m \text{ fois}}$$

L'opérateur \times désigne le produit cartésien sur les "m lignes".

Soit $S \subseteq \mathcal{D}^*(n, m)$ un sous-ensemble de ce produit cartésien. $\Pi_1(S)$ est la projection de S sur le i-ème facteur ou la "i-ème ligne".

Définition 1.54. Rectangularisation

La rectangularisation de S , notée $Rec(S)$, est définie par:

$$Rec(S) = \Pi_1(S) \times \cdots \times \Pi_m(S) \quad (1.34)$$

S est dite rectangulaire si $S = Rec(S)$

On peut donner une autre définition à la rectangularisation comme suit : l'ensemble S des fonctions $(\min, \max, +)$ est rectangulaire si pour tout $G, G' \in S$, et pour tout $i = 1, \dots, n$, la fonction obtenue par le remplacement de la i -ème composante de G par la i -ème composante de G' appartient à S .

Il a été montré dans [GG98a] que toute fonction de type $(\min, \max, +)$ décrite par 1.31 et 1.32 peut être réécrite sous la forme :

$$F(x) = \bigoplus_{G \in T} G(x) = \bigwedge_{H \in S} H(x) \quad (1.35)$$

avec, S et T représentant ici des représentations rectangulaires $(\max, +)$ et $(\min, +)$ respectivement de F .

Le théorème cité ci-dessous 1.55 permet de donner des conditions nécessaires et suffisantes de l'existence du vecteur propre d'une fonction $(\min, \max, +)$. Sa preuve permet de déduire un algorithme calculant le vecteur spectral associé à cette fonction [GG99] lorsque chaque fonction $(\max, +)$ présente un point fixe. Un algorithme opérant dans le cas général est présenté dans [GG98a].

Théorème 1.55. [CT01] Soit $F \in D^*(n, n)$ et supposons que $S, T \in D^*(n, n)$ soient rectangulaires et respectivement, une max-représentation et une min-représentation de F . Les conditions suivantes sont équivalentes.

1. F a un vecteur propre avec $F(x) = x + h$
2. $h = \bigwedge_{H \in S} \chi(H)$
3. $h = \bigoplus_{G \in T} \chi(G)$

Les théorèmes cités ci-dessous 1.56, 1.57 et 1.58, présentent des résultats primordiaux pour la suite du mémoire. En effet, ils donnent les conditions nécessaires et suffisantes sur l'existence d'une solution finie pour des inéquations de types $x \leq F(x)$, $x \geq F(x)$ et $x = F(x)$, avec $F(x)$ une fonction $(\min, \max, +)$.

Nous présentons les preuves des théorèmes 1.56 et 1.57 ci-dessous à partir d'indications données par S. Gaubert.

Théorème 1.56. [GG98a][CT01]

Soit $F \in D^*$. Les deux points suivants sont équivalents :

(i) il existe un vecteur x fini tel que $x \leq F(x)$

(ii) $\chi(F) \geq 0$

Preuve :

(i) \Rightarrow (ii)

Nous avons $x \leq F(x)$ et $F(x)$ est monotone, donc $x \leq F^2(x) \Rightarrow x \leq F^k(x)$ avec $k \geq 1$. Par application de la définition 1.45 $\lim_{k \rightarrow +\infty} \frac{F^k(x)}{k} = \lim_{k \rightarrow +\infty} \frac{x(k)}{k} \geq \lim_{k \rightarrow +\infty} \frac{x}{k}$ puisque x est fini.

(i) \Leftarrow (ii)

Le corollaire 1.52 cité ci-dessus exprime que chaque fonction dans D^* a un régime ultime affine. En plus, $\chi(F)$ égale η , pour tout régime ultime $(\eta, v) \in (\mathbb{R}^n)^2$ de F . Soit un régime ultime (η, v) . Nous avons $\forall k \geq K$, $F(v + k\eta) = v + (k + 1)\eta$. Nous posons $y = v + k\eta$ avec $k \geq K$. La variable y vérifie $F(y) = y + \eta$. Puisque $\eta = \chi(F) \geq 0$, nous déduisons qu'il existe un vecteur y tel que $F(y) \geq y$

Notons aussi que le vecteur infini $-\infty$ ou $(\varepsilon, \dots, \varepsilon)^t$ peut être considéré comme une solution sur \mathbb{R}_{max}^n mais pas sur \mathbb{R}^n .

□

Théorème 1.57. [GG98a][CT01]

Soit $F \in D^*$. Les deux points suivants sont équivalents :

(i) il existe un vecteur x fini tel que $x \geq F(x)$

(ii) $\chi(F) \leq 0$

Preuve :

La démonstration de ce théorème se présente comme le dual de la démonstration du théorème précédent 1.56.

(i) \Rightarrow (ii)

Nous avons comme point de départ $x \geq F(x)$. Or, du fait de la non-décroissance de F , on peut écrire $x \geq F(x) \geq F^2(x) \geq F^k(x)$ avec $k \geq 1$. Par application de la définition 1.45, on a $\chi(F, x) = \lim_{k \rightarrow \infty} x(k)/k = \lim_{k \rightarrow \infty} F^k(x)/k \leq \lim_{k \rightarrow \infty} x/k = 0$ car x est un vecteur fini.

(i) \Leftarrow (ii)

Nous savons que chaque fonction F dans D^* a régime ultime, et $\chi(F) = \eta$, pour tout régime ultime $(\eta, v) \in (\mathbb{R}^n)^2$ de F . Donnons un régime ultime (η, v) . Nous avons $\forall k \geq K$, $F(v + k\eta) = v + (k + 1)\eta$. Nous posons $y = v + k\eta$ avec $k \geq K$. Le vecteur y vérifie $F(y) = y + \eta$. Comme $\eta = \chi(F) \leq 0$, nous déduisons qu'il existe un vecteur y tel que $F(y) \leq y$

□

Théorème 1.58. *Soit $F \in \mathcal{D}^*$. Les deux points suivants sont équivalents:*

(i) *il existe un vecteur x fini tel que $x = F(x)$*

(ii) $\chi(F) = 0$

Preuve :

La démonstration de ce théorème découle immédiatement des résultats énoncés dans les deux théorèmes cités ci-dessus 1.56 et 1.57.

□

Remarque :

Le dernier théorème 1.58 peut être considéré aussi comme un corollaire du théorème 1.55 quand $h = 0$

◇

L'étude des démonstrations précédentes montre que celles-ci nécessitent uniquement l'existence du vecteur spectral ou temps de cycle et du régime affine ultime. On peut donc généraliser les théorèmes précédents aux fonctions topicales sous condition d'existence de celles-ci. Le résultat peut encore être amélioré par le fait que l'existence du vecteur spectral entraîne l'existence du régime affine et inversement [GG98b], [CT01]. Dans les trois théorèmes précédents 1.56, 1.57 et 1.58, le terme “Soit $F \in \mathcal{D}^*$ ” peut ainsi être remplacé par “Soit $F \in \mathbf{T}_p$ ” tel que le vecteur spectral existe.

1.7 Conclusion

Ce premier chapitre nous a permis de faire un tour d'horizon sur un ensemble d'outils et d'éléments de base mathématiques dont les principaux sont la théorie spectrale des fonctions (\min , \max , $+$) et la résolution des inéquations présentant une forme du type point fixe. Le problème posé est de savoir si l'équation $x \leq f(x)$ (respectivement $x \geq f(x)$) admet une solution et de la calculer dans le cas affirmatif. Le vecteur spectral se présente ainsi comme un moyen permettant de répondre au problème de l'existence d'une solution. Nous avons pu constater que la solution de ce problème présentée dans [KG94] (voir théorème 1.39) généralise les résultats donnés dans [BCOQ92] (voir les théorèmes 1.34 et 1.35) en donnant une définition moins restrictive de la semi-continuité. Nous avons rappelé aussi les principes de base de la théorie de la résiduation et ses applications dans les dioïdes. Il s'avérera que la résiduation participe aussi à la résolution des problèmes du type point fixe.

Après cette étude relative aux outils algébriques, nous donnerons dans le chapitre suivant une présentation générale des modèles graphiques utilisés dans ce mémoire.

Chapitre 2

Modèles et classes de systèmes temporels

2.1 Introduction

Dans ce chapitre, l'articulation entre les aspects graphique et algébriques des différents systèmes, se fera grâce à une mise en équations du graphe dans l'algèbre des dioïdes ou exotique. Ce choix est en effet justifié par la cohérence et la puissance des outils mathématiques développés, comme l'a montré le chapitre précédent. Les systèmes d'équations et/ou d'inéquations obtenus constitueront des supports permettant un traitement plus efficace que la manipulation directe du réseau de Petri. Dans ce chapitre, on s'est intéressé particulièrement à une sous classe des réseaux de Petri appelée graphe d'événements, pour laquelle il sera vu que cette écriture mathématique est réalisable de manière satisfaisante.

La modélisation des graphes d'événements temporisés dans l'algèbre des dioïdes est un exemple maintenant bien connu dans la communauté scientifique [CGQ98], [CHBF01] et [Men97]. Il est cependant nécessaire d'élargir sa puissance de modélisation. C'est pour cela qu'un travail de modélisation des graphes d'événements temporels dans l'algèbre des dioïdes, est introduit dans cette thèse.

De manière classique, il existe deux façons d'introduire des temporisations constantes dans un réseau de Petri. En l'associant aux transitions, on parle alors du temps de franchissement et le modèle est du type réseaux de Petri t -temporisés [Ram74]. Si l'on associe le temps aux places, on parle de durée de séjour minimale et le modèle est du type réseaux de Petri p -temporisés [Sif77]. Les deux modèles sont équivalents [Sif79] et le passage entre ces deux sous classes a été montré possible [Mur89].

Cependant, une deuxième famille considère la possibilité d'associer des intervalles de temps

$[a, b]$ au modèle. Cette association a d'abord concerné les transitions [Mer74] et [MF76]. Puis, plusieurs autres modèles temporels ont été développés : réseaux de Petri à arcs temporels [Wal83], réseaux de Petri à flux temporels [Dia01] et ces dernières années, réseaux de Petri à places temporelles [Kha97]. Plus récemment, en 1999, les réseaux de Petri statiquement temporisés ¹ ont été introduits par A. Cerone et A. Maggiolo-Schettini [CMS99]. Ce dernier modèle généralise les réseaux de Petri temporels en associant des intervalles de temps à la fois aux places, aux transitions et aux arcs incidents aux transitions.

Pour la modélisation dans l'algèbre des dioïdes des graphes d'événements temporisés dont le comportement est déterministe, les résultats sont bien connus [CGQ98], [Cot99], [Men97], [Lho03]. En revanche, pour les autres graphes d'événements temporels, aucun travail de modélisation dans l'algèbre des dioïdes n'a été fait jusqu'à ce jour à notre connaissance. L'objectif est donc de faire un travail de modélisation des graphes d'événements temporels dans l'algèbre des dioïdes ainsi qu'une classification des différentes sous-classes présentées.

Ce chapitre est structuré comme suit :

Dans la première partie, nous rappellerons, tout d'abord, les notions de base concernant les réseaux de Petri, en évoquant quelques propriétés de ces derniers. L'introduction de sous-classes importantes que constituent les graphes d'événements et les graphes d'état, se fera aussi dans cette partie. Cela nous permettra, ensuite, de modéliser les graphes d'événements temporisés dans l'algèbre des dioïdes.

Dans une deuxième partie, nous introduirons la modélisation des graphes d'événements temporels. Les modèles qui seront générés, sont de type "intervalle", dont le fonctionnement est régi par des inéquations (inégalités). Nous présenterons une classification de différents modèles vus dans ce chapitre. Le critère pris afin de réaliser cette classification sera le type du système mathématique représentant chacun de ces modèles. Enfin, nous exposons quelques autres modèles physiques dont l'évolution dynamique sera représentée par des équations de type (min, max, +).

Les premiers résultats relatifs à la modélisation des graphes d'événements temporels sont proposés dans [DAD04], [DDA05c] et [DDA05a].

Remarque : Un RdP est dit ordinaire, lorsque tous les poids des arcs sont égaux à 1.

◇

¹on notera qu'on aurait dû utiliser le terme temporels au lieu de temporisés. On a gardé le dernier terme, car dans l'article original, les auteurs ont utilisé *Statically timed Petri nets*

2.2 Les réseaux de Petri

2.2.1 Rappels et notions de base

Nous présentons dans cette première section des notions élémentaires sur les réseaux de Petri. Des développements plus approfondis peuvent être trouvés dans [Bra83], [DA97], [Pet62], [Mur89], [DAJ92] et [DA04].

Définition 2.1. (Réseau de Petri) Un réseau de Petri (RdP) est un tuple $N = \langle P, T, A, W \rangle$ tel que

- P est un ensemble fini de places,
- T est un ensemble fini de transitions, disjoint de P ($P \cap T = \emptyset$),
- $\mathcal{A} \subset (P \times T) \cup (T \times P)$ est un ensemble fini d'arcs,
- $W : \mathcal{A} \longrightarrow \mathbb{N}$ est la fonction poids attachée aux arcs.

La grandeur w_{ij}^- (respectivement w_{ij}^+) représente le poids de l'arc qui relie une transition t_j à une place en amont p_i (respectivement une transition t_j à une place en aval p_i). Si $w_{ij}^- \neq 0$ (resp. $w_{ij}^+ \neq 0$), alors, il n'existe pas d'arc entre p_i en amont de t_j (resp. p_i en aval de t_j). On définit la matrice d'incidence d'un RdP comme suit :

$$W = W^+ - W^-$$

La matrice W^- est la matrice d'incidence avant tel que $W^- = [w^-]$, et W^+ est la matrice d'incidence arrière tel que $W^+ = [w^+]$

Remarque : Toutes les sous-classes (graphes d'événements) des réseaux de Petri (RdPs) traitées dans cette thèse, seront considérées comme ordinaires.

◇

Définition 2.2. (Marquage) Soit N un réseau de Petri. Un marquage est une fonction M de l'ensemble des places P dans \mathbb{N} .

On note $\mathcal{M}(N)$ l'ensemble des marquages d'un réseau de Petri.

Définition 2.3. (Réseau de Petri Marqué) Un réseau de Petri marqué est un couple $\langle N, M_0 \rangle$ tel que :

- N est un réseau de Petri
- $M_0 : P \longrightarrow \mathbb{N}$ est un marquage dit marquage initial.

Nous adoptons désormais les notations suivantes dans la suite de ce mémoire :

- t l'ensemble des places d'entrée de la transition t
- t^\bullet l'ensemble des places de sortie de la transition t
- p l'ensemble des transitions d'entrée de la place p

p^\bullet l'ensemble des transitions de sortie de la place p

Un RdP est un modèle dynamique dont l'évolution est liée à l'évolution du marquage. Son état à un instant donné est représenté par son marquage M à cet instant. La caractérisation d'un RdP à un instant donné est souvent donnée par le couple (N, M) .

On dit qu'une transition est tirable (franchissable) si, quelque soit $p_i \in \bullet t_j$, $M(p_i) \geq w_{ij}^-$. Autrement dit, si toute place d'entrée p_i de t_j contient un nombre de jetons au moins égal au poids attaché à l'arc qui relie p_i à t_j . Dans le cas d'un RdP ordinaire, il suffit que toutes les places d'entrée d'une transition contiennent au moins un jeton pour qu'elle soit franchissable.

Une transition franchissable peut être franchie ou non. Franchir une transition t_j consiste à :

- retirer w_{ij}^- jetons de toute place $p_i \in \bullet t_j$,
- ajouter w_{ij}^+ jetons dans toute place $p_i \in t_j^\bullet$.

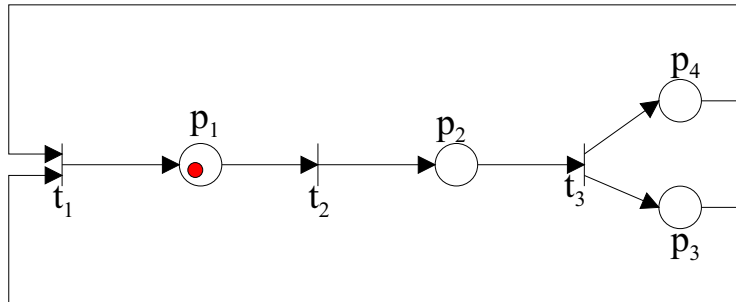


Figure 2.1 : Exemple d'un RdP ordinaire

Une séquence de franchissements à partir d'un marquage M est représentée par une suite de transitions. Si nous considérons le RdP de la figure 2.1 et son marquage initial $M_0 = (1, 0, 0, 0)^t$, alors, nous dirons que nous sommes passés de M_0 à $M_2 = (0, 0, 1, 1)$ en effectuant le tirage de la séquence $\sigma < t_2, t_3 >$ et nous écrivons $M_0 \xrightarrow{\sigma} M_2$.

L'équation fondamentale qui caractérise l'évolution et la dynamique d'un RdP est :

$$M_k = M_0 + W.S \quad (2.1)$$

M_k est un marquage que l'on atteint à partir de M_0 après une séquence de franchissements σ réalisable.

S est un vecteur qui représente cette séquence σ . Sa dimension est égale au nombre de transitions que contient le RdP. Chaque composante S_j vaut le nombre de fois où la transition t_j a été franchie pendant la séquence σ .

Reprenons l'exemple de la figure 2.1. Le vecteur S est égal $S = (0, 1, 1)^t$ que l'on peut interpréter de la manière suivante : t_2 et t_3 sont franchies chacune une seule fois pour passer de M_0 à M_2 . La matrice d'incidence W est donnée par :

$$W = W^+ - W^- = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix} - \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{pmatrix}$$

Le marquage M_2 peut être aussi calculé à partir des matrices M_0 , S et W données ci-dessus et en appliquant l'équation 2.1.

2.2.2 Quelques propriétés des RdPs

Les principales propriétés des RdPs qui peuvent être étudiées, sont classées en deux groupes : celles qui dépendent du marquage initial sont appelées “propriétés comportementales” tandis que celles qui ne dépendent que de la structure du RdP sont appelées “propriétés structurelles”.

Pour étudier et vérifier les propriétés comportementales, on établit l'arbre de couverture appelé aussi graphe de couverture. Il est constitué de noeuds², et d'arcs orientés, correspondant au franchissement d'une transition qui fait évoluer le RdP d'un état ou marquage à un autre.

- **Accessibilité** : Le problème d'accessibilité consiste à trouver si l'on peut atteindre un marquage M' à partir de M_0 .

Soit un RdP (N, M_0) , on dit qu'un marquage M' est accessible (ou atteignable) à partir de M_0 s'il existe une séquence de franchissement σ telle que $M_0 \xrightarrow{\sigma} M'$. $\mathcal{R}(M_0)$ est l'ensemble des marquages que l'on peut atteindre en partant de M_0 .

- **Bornitude** : Un RdP (N, M_0) est borné si, quelque soit la place p_i et quelque soit le marquage accessible M' à partir M_0 , le nombre de jetons dans cette place p_i est borné :

$$\forall M \in \mathcal{R}(M_0), \forall p \in P, \exists k \in \mathbb{N} \text{ tel que } M(p) \leq k$$

- **Vivacité** : Une transition t_j est dite vivante si elle peut être franchie quelque soit le marquage atteint :

$$\forall M \in \mathcal{R}(M_0), \exists M' \in \mathcal{R}(M_0) \text{ tel que } t_j \text{ soit franchissable pour } M'$$

- Un RdP (N, M_0) est vivant si chacune de ses transitions est vivante.
- Dans un RdP (N, M_0) , un état de blocage est un marquage tel qu'aucune transition n'est validée.

²chaque noeud correspond à un marquage atteignable

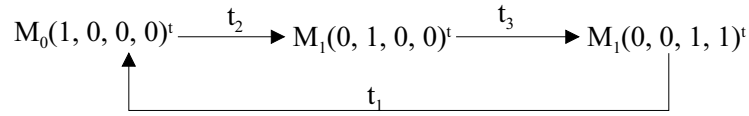


Figure 2.2 : Graphe des marquages accessibles du RdP de la figure 2.1

Nous pouvons dire que le graphe de la figure 2.1 est vivant et borné, tout simplement, en analysant son arbre de couverture donné par la figure 2.2.

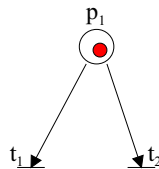
Comme la vérification de ces propriétés (vivacité et accessibilité) utilise le graphe de couverture basé sur le marquage, cette méthode devient inefficace dans le cas où l'établissement de ce graphe est difficile, comme, par exemple, dans le cas d'un RdP non borné.

2.2.3 Sous classes des RdPs

Dans l'étude des systèmes dynamiques à événements discrets (SDED), on rencontre fréquemment des phénomènes de types concurrence, synchronisation et parallélisme, qui sont modélisés par différentes sous-classes des RdPs que nous exposons maintenant.

- **Graphes d'états** : Un RdP ordinaire marqué est un graphe d'état si et seulement si, toute transition a exactement une place d'entrée et une place de sortie :

$$|\bullet t| = |t\bullet| = 1 \forall t \in T$$



(b)

Figure 2.3 : Conflit structurel représenté par un graphe d'états

Le RdP de la figure 2.3 représente un graphe d'états. Les transitions t_1 et t_2 de 2.3 sont en conflit structurel puisqu'elles ont une place d'entrée commune p_1 . Comme le montre le RdP de cette figure, le jeton de la place p_1 peut franchir soit t_1 ou t_2 , mais pas les deux transitions simultanément. Le problème d'exclusion mutuelle modélise souvent des systèmes comportant un partage de ressources.

- **Graphes d'événements** : La définition d'un graphe d'événements est duale de celle d'un graphe d'état. Un RdP marqué est un graphe d'événements si et seulement si, toute place a

exactement une transition d'entrée et une transition de sortie :

$$|\bullet p| = |p\bullet| = 1 \quad \forall p \in P$$

Le graphe de la figure 2.1 est un graphe d'événements.

Nous rappelons ici brièvement quelques propriétés structurelles des graphes d'événements et qui sont indépendantes du marquage initial. L'analyse de ces propriétés utilise l'algèbre linéaire [Sif78]. Nous allons aussi définir les p-invariants et les t-invariants.

Définition 2.4. *On dit qu'un vecteur X est un p-invariant, si X vérifie la relation suivante :*

$$X.W = 0$$

avec X un vecteur ligne entier non-négatif appelé aussi, P -semi-flot et W la matrice d'incidence.

L'ensemble des places qui correspondent aux composantes non nulles d'un p-invariant, est appelé support du p-invariant.

La recherche des p-invariants pour les graphes d'événements est présentée dans [PX95] sous la forme suivante :

Soit $X = (x_1, x_2, \dots, x_n)$ un vecteur à composantes entières non négatives, où n est le nombre de places du graphe d'événements considéré. Soit c un circuit élémentaire de ce graphe et $p = \{p_1, p_2, \dots, p_n\}$ l'ensemble des places du graphe. Si, pour i allant de 1 à n

$$x_i = \begin{cases} 1 & \text{si } p_i \in c \\ 0 & \text{sinon} \end{cases}$$

Alors, $\forall M_k \in \mathcal{R}(M_0)$, on a :

$$X.M_0 = X.M_k$$

Ce résultat découle immédiatement de l'équation fondamentale de l'évolution dynamique d'un RdP 2.1.

Si nous considérons un graphe d'événements pour lequel toutes les places appartiennent à un support de p-invariant, alors ce graphe d'événements est borné.

Le théorème suivant exprime la conservation du nombre de jetons dans un circuit d'un graphe d'événements.

Théorème 2.5. *Dans un graphe d'événements, la somme des marques des places d'un circuit donné est constante.*

Définition 2.6. *Un vecteur Y est un t-invariant, s'il vérifie l'équation suivante :*

$$W.Y^t = 0$$

avec Y un vecteur ligne appelé aussi T -semi-flot dont la dimension est le nombre de transitions du RdP.

De manière similaire au support du p-invariant, on définit le support du t-invariant comme l'ensemble des transitions qui correspondent aux composantes non nulles d'un t-invariant.

L'existence d'un t-invariant pour un graphe d'événements est une condition nécessaire pour que le graphe ait un fonctionnement répétitif (cyclique). Soit maintenant un graphe d'événements avec m transitions. Le vecteur Y , à m composantes toutes égales à 1, est l'unique t-invariant [PX95]. Une autre manière d'exprimer ce résultat est de dire que l'on retrouve le même marquage initial après avoir franchi une seule fois chaque transition.

Théorème 2.7. [PX95] *Un graphe d'événements autonome³ est sans blocage et vivant, si et seulement si, chaque circuit élémentaire contient au moins un jeton.*

Après avoir présenté des notions de base sur les RdPs, notre attention va se porter maintenant sur les graphes d'événements (temporisés et temporels). Le paramètre temps sur ces derniers sera introduit de différentes manières.

2.3 Graphes d'événements temporisés (GET)

2.3.1 Temporisation et fonctionnement d'un graphe d'événements

L'introduction d'un nouveau paramètre qui est le temps sur les graphes d'événements va nous permettre de définir les graphes d'événements temporisés. Deux façons sont envisageables :

1. soit nous associons à chaque transition t une durée minimale de tir $\theta(t)$, qui représente le temps de réservation d'un jeton dans une place en amont, avant d'être disponible à nouveau, dans une place en aval. On parle donc des graphes d'événements t-temporisés.
2. soit nous associons à chaque place une durée minimale $\theta(p)$, qui correspond à un temps d'indisponibilité d'une marque après son arrivée à cette place, avant d'être utile pour un nouveau franchissement. On parle donc des graphes d'événements p-temporisés.

Du fait de la possibilité de transformer une place temporisée en une transition temporisée et vice versa, l'équivalence entre les deux types de graphes peut être établie [Chr83]. A partir du paragraphe 2.3.2, nous pouvons donc ne considérer, que les graphes d'événements p-temporisés, qui seront notés GET.

Fonctionnement au plus tôt d'un GET

On appelle fonctionnement au plus tôt d'un GET, le mode de fonctionnement suivant : toutes les transitions internes et puits sont franchies dès que possible. Autrement dit, les jetons

³on dit qu'un graphe d'événement est autonome s'il ne contient pas de transitions sources

qui arrivent dans une place, pourront contribuer au franchissement de la transition en aval dès que le temps de séjour minimum se termine. Aussi, les transitions sources sont en permanence franchissables, mais ne sont franchies que sur l'occurrence d'événements externes associés à ces transitions. Il a été montré dans [DA97] et [DA04] que le fonctionnement au plus tôt⁴ d'un GET conduit à un régime périodique stationnaire au bout d'un certain temps fini, pour tout marquage initial tel que le GET soit borné et vivant.

Fonctionnement FIFO

- Si le k -ième jeton qui arrive dans une place d'un graphe d'événements p -temporisé est aussi le k -ième jeton disponible pour franchir une transition en aval, alors cette place fonctionne selon la règle FIFO⁵.

- Si le k -ième tir d'une transition d'un graphe d'événements t -temporisé à commencer est aussi le k -ième tir à se terminer, alors cette transition fonctionne selon la règle FIFO.

Définition 2.8. *On dit qu'un graphe d'événements est FIFO, si toutes ses places et transitions sont FIFO.*



Figure 2.4 : Recyclage d'une transition et d'une place pour garantir le fonctionnement FIFO

On peut imposer une condition structurelle sur le graphe d'événements pour garantir le fonctionnement FIFO des transitions. Cela peut se traduire par l'introduction d'une boucle sur chaque transition comme le montre la figure 2.4a.

Cependant, comme nous associons systématiquement des temporisations seulement aux places pour les exemples que l'on traite dans cette thèse, le franchissement des transitions est instantané. On peut donc déduire un fonctionnement FIFO pour les places et pour les transitions avales par conséquent.

Remarque : - De même, on introduit une boucle de façon identique à celle faite sur une transition, pour garantir un fonctionnement FIFO pour les places 2.4b.

⁴dans [DA97] on parle aussi du fonctionnement à vitesse maximale

⁵First In, First Out : premier entré, premier sortant

- Notons qu'il suffit que le temps de séjour lié à une place, soit constant, pour que celle-ci fonctionne sous l'hypothèse FIFO.

◇

2.3.2 Equations aux dateurs et aux compteurs

Rappels sur la transformée en γ

La transformée en γ joue un rôle analogue à la transformée en z dans la théorie des systèmes échantillonnés classiques. Elle permet d'écrire les convolutions en des produits de séries formelles.

Définition 2.9. Soit $x(\cdot)$ un dateur associé à une transition. La série formelle définie par :

$$X(\gamma) = \bigoplus_{k \in \mathbb{Z}} x(k) \gamma^k \quad (2.2)$$

est la transformée en γ de $\{x(k)\}_{k \in \mathbb{Z}}$

Le terme γ peut être interprété comme un opérateur que l'on appellera "opérateur de décalage" ou de "retard". En effet, si nous considérons deux dateurs $x_1(\cdot)$ et $x_2(\cdot)$ tels que $x_1(k) = x_2(k - q)$, alors :

$X_1(\gamma) = \bigoplus_{k \in \mathbb{Z}} x_1(k) \gamma^k = \bigoplus_{k \in \mathbb{Z}} x_2(k - q) \gamma^k = \gamma^q \bigoplus_{k \in \mathbb{Z}} x_2(k - q) \gamma^{k-q} = \gamma^q X_2(\gamma)$ avec $q \in \mathbb{Z}$. Plus formellement, on écrit :

$$\gamma^q x(k) = x(k - q) \quad (2.3)$$

La structure algébrique $\overline{\mathbb{Z}}_{\max} [[\gamma]]$ est le dioïde complet [BCOQ92] de séries formelles en γ à coefficients dans $\overline{\mathbb{Z}}_{\max}$ et exposants dans \mathbb{Z} qu'il convient d'utiliser pour cette nouvelle représentation. Ce dioïde est muni de deux lois de compositions internes : \oplus (l'opérateur max) est la somme de séries formelles et \otimes est le produit de séries formelles. L'élément neutre de la multiplication est la série $e(\gamma) = e\gamma^e$ et l'élément neutre de l'addition est la série $\varepsilon(\gamma) = \bigoplus_{k \in \mathbb{Z}} \varepsilon \gamma^k$ ($e = 0$ et $\varepsilon = -\infty$ sont les deux éléments neutres respectivement des loi \otimes et \oplus dans $\overline{\mathbb{Z}}_{\max}$).

Les séries formelles présentant des règles de manipulation des éléments de $\overline{\mathbb{Z}}_{\max} [[\gamma]]$ dont voici un exemple.

$$(\gamma^n \oplus \gamma^p)x(k) = \gamma^{\min(n,p)}x(k).$$

En effet,

$$\begin{aligned} (\gamma^n \oplus \gamma^p)x(k) &= x(k - n) \oplus x(k - p) \\ &= \max(x(k - n), x(k - p)) \\ &= x(k - \min(n, p)) \quad (\text{puisque } x \text{ est isotone}) \\ &= \gamma^{\min(n,p)}x(k) \end{aligned}$$

Modélisation

Jusqu'à présent, nous nous sommes intéressés uniquement à l'aspect structurel des RdP. L'évolution dynamique de tels systèmes est décrite par des équations liées aux marquages des places. Si par exemple, on veut analyser les performances d'un système présenté par un graphe d'événements temporisé, alors le moyen de modélisation basé uniquement sur le marquage devient inefficace. Il convient donc de chercher un autre outil de modélisation. Désormais, on va introduire de nouvelles variables d'état liées, non plus au marquage du graphe d'événements, mais aux transitions de ce dernier. Après la mise en équations, le comportement d'un graphe d'événements temporisé peut être décrit par un modèle linéaire dans l'algèbre $(max, +)$ (ou l'algèbre $(min, +)$).

Il existe deux manières pour établir les relations algébriques représentant l'évolution d'un graphe d'événements temporisé. La première voie consiste à décrire le comportement temporel des graphes d'événements, en travaillant dans un domaine qui est l'espace des événements appelé domaine événementiel. On parle alors de dateurs, et chaque date d'événement sera repérée par rapport à un numéro d'un événement. La deuxième voie est duale de la première, le domaine de travail étant l'espace du temps aussi appelé domaine temporel. On parle alors usuellement de compteurs et chaque compte sera référencé par rapport au temps avec une certaine parenté avec le modèle continu de l'automatique classique. Nous définissons maintenant les dateurs et les compteurs et donnons les modèles correspondants.

Nous adoptons les notations suivantes dans la suite de ce mémoire :

$x(.)$ désigne une variable d'état liée à une transition interne

$u(.)$ désigne une variable d'état associée à une transition d'entrée

$y(.)$ désigne une variable d'état associée à une transition de sortie

Nous gardons le même nom de variables pour les deux types de présentations (équations aux dateurs ou bien aux compteurs). Le seul paramètre qui peut différencier les deux modèles est repéré au niveau de l'argument, ainsi, k (respectivement t) désigne un dateur (respectivement un compteur).

2.3.2.1 Equations aux dateurs

Définition 2.10. *Un dateur d est une application monotone (croissante) de \mathbb{Z} dans $\mathbb{R} \cup \{-\infty, +\infty\}$.*

Un dateur doit donc vérifier :

$$d(k+1) \geq d(k)$$

Nous associons à chaque transition x_i (respectivement u_i ou y_i) un dateur $x_i(k)$ (respectivement $u_i(k)$ ou $y_i(k)$), qui représente la date à valeurs réelles (et donc possiblement négatives), à laquelle l'événement k à valeurs entières (possiblement négatives), associé à chaque variable, a lieu.

Conditions initiales canoniques

Nous supposons, lors de l'établissement des équations d'évolution du système, que les jetons qui existent dans chaque place p_j à l'instant $t = 0$ sont disponibles depuis $-\infty$. Cela veut dire que $\forall k < 0, x_i(k) = -\infty$. Les jetons présents dans une place à $t = 0$ doivent effectuer leur durée de séjour minimale, avant qu'ils ne soient disponibles pour la participation au tir d'une transition en aval. Par convention, on considère qu'un événement k qui n'a jamais eu lieu est formulé mathématiquement comme suit : $x_i(k) = +\infty$. Par convention, le k ème jeton de la place p_j ($k \geq 0$) est celui qui va contribuer au franchissement d'une transition aval p^\bullet pour la k ème fois durant l'évolution du graphe d'événements, en incluant les jetons du marquage initial. Les tirs des transitions sont numérotés à partir de 1 (le premier tir d'une transition porte le numéro $k = 1$).

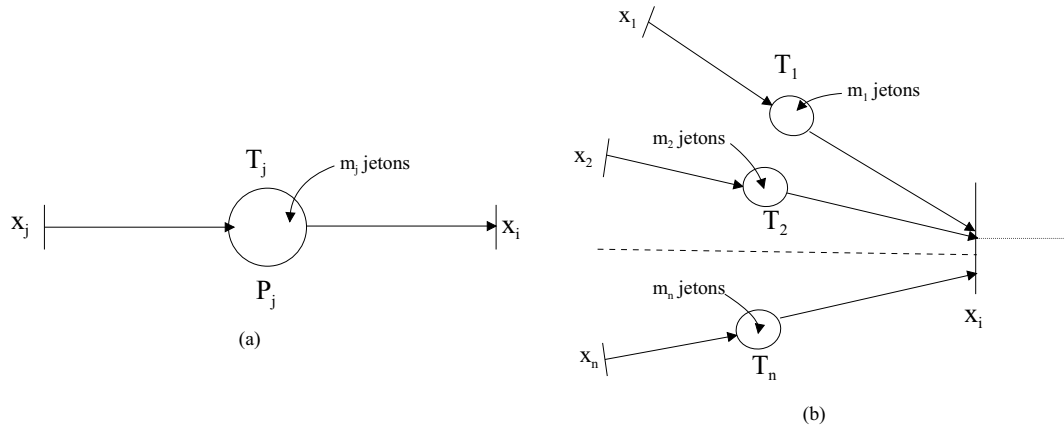


Figure 2.5 : Parties d'un graphe d'événements temporisé

Nous rappelons que dans un graphe d'événements, chaque transition peut avoir plusieurs places en amont ou en aval. Les différentes parties que l'on peut trouver dans les GET, sont représentées dans la figure 2.5. Les m_j jetons sont supposés présents depuis l'instant $t = 0$, et ils vont séjourner au minimum pendant T_j unités de temps dans la place p_j . Si la constante T_j est la temporisation associée à la place p_j et m_j le nombre de jetons présents initialement dans p_j , alors le tir de x_i aura lieu au plus tôt pour la k -ième fois après le tir de x_j pour la $(k-m_j)$ -ième fois, plus un temps de séjour minimal. Les deux dateurs $x_j(k)$ et $x_i(k)$ de la figure 2.5a satisfont l'inéquation suivante :

$$x_i(k) \geq x_j(k - m_j) + T_j \quad (2.4)$$

De même, pour la transition x_i de la figure 2.5b, le franchissement de x_i pour la k ème fois est conditionné par :

- le tir de x_1 pour la $(k-m_1)$ ème fois
- le tir de x_2 pour la $(k-m_2)$ ème fois

- et ainsi de suite jusqu'au tir de x_n pour la $(k-m_n)$ ième fois

En tenant compte de ces conditions, nous obtenons donc le système :

$$\left\{ \begin{array}{l} x_i(k) \geq x_1(k - m_1) + T_1 \\ x_i(k) \geq x_2(k - m_2) + T_2 \\ \vdots \\ x_i(k) \geq x_n(k - m_n) + T_n \end{array} \right. \iff x_i(k) \geq \max(x_1(k - m_1) + T_1, x_2(k - m_2) + T_2, \dots, x_n(k - m_n) + T_n) \quad (2.5)$$

Dans le dioïde $\overline{\mathbb{R}}_{\max}$ (où $a \oplus b = \max(a, b)$ et $a \otimes b = a + b$), l'inéquation 2.5 peut se mettre sous forme d'un modèle linéaire, on obtient donc la forme implicite :

$$x_i(k) \geq \bigoplus_{j=1}^n x_j(k - m_j) \otimes T_j \quad (2.6)$$

Notons, que pour l'instant, il n'est pas possible de déduire l'évolution future de la transition x_i avec l'équation 2.6 si on connaît les valeurs de $x_j(k - m_j)$. Pour résoudre cette difficulté, l'hypothèse d'un fonctionnement au plus tôt, peut être choisie au niveau de la transition x_i , ce qui peut être considéré déjà comme une règle de comportement ou de commande locale. Sous cette hypothèse de fonctionnement, le comportement deviendra déterministe et le système 2.6 pourra se mettre sous la forme :

$$x_i(k) = \bigoplus_{j=1}^n x_j(k - m_j) \otimes T_j \quad (2.7)$$

que l'on peut aussi présenter par des séries formelles en γ

$$x_i(\gamma) = \bigoplus_{j=1}^n \gamma^{m_j} x_j \otimes T_j \quad (2.8)$$

2.3.2.2 Equations aux compteurs

Comme dans le cadre de la théorie des systèmes classiques, les équations qui décrivent le fonctionnement dynamique de ces systèmes, sont en fonction du temps t . D'une manière analogue, nous allons associer à chaque transition, une fonction $x(t)$ qui est appelée compteur.

Contrairement aux dateurs, nous pouvons distinguer plusieurs interprétations de la définition des compteurs.

- Le compteur $x(t)$ indique le numéro de la dernière activation de la transition x survenue avant ou à t .

$$c(t) = \sup\{k \text{ tel que } d(k) \leq t\}$$

- Le compteur $x(t)$ indique le numéro de la première activation de la transition x survenue à t ou après t .

$$c(t) = \inf\{k \text{ tel que } d(k) \geq t\}$$

Définition 2.11. *Un compteur c est une application monotone croissante de \mathbb{R} dans $\mathbb{Z} \cup \{+\infty, -\infty\}$.*

On a donc :

$$c(t + \Delta t) \geq c(t), \quad \text{avec } \Delta t \geq 0$$

Dans ces deux interprétations, il faut interpréter le compteur comme une fonction, affectant à chaque instant t , un numéro ou un rang. Le compteur $c(t)$ est donc un entier mais qui peut être négatif. Il peut alors être défini sur \mathbb{Z} et ses extensions.

- Le compteur $x(t)$ indique la somme cumulée [Lib96]. Il représente alors une fonction effectuant un comptage, en fonction du temps. Le compteur $c(t)$ est donc un entier mais ne pouvant être que positif ou nul. Il peut alors être défini sur \mathbb{N} et son extension $\mathbb{N} \cup \{+\infty\}$.

Nous reprenons l'exemple de la figure 2.5, et nous associons pour chaque transition x_j la variable $x_j(t)$ qui représente le nombre de fois où la transition x_j a été activée, avant ou à la date t . Nous travaillons sous les mêmes conditions initiales et les mêmes hypothèses, comme pour les dateurs. Le compteur $x_i(t)$ ne peut pas excéder le compteur $x_j(t - T_j)$ car tous les jetons doivent séjourner dans p_j , au moins T_j unités de temps. Par conséquent, la valeur $x_i(t)$ ne peut pas dépasser $x_j(t - T_j) + m_j$, du fait qu'il existe m_j jetons en réserve venant du franchissement de x_j antérieurement. Les deux compteurs $x_j(t)$ et $x_i(t)$ de l'exemple 2.5a satisfont l'inéquation suivante :

$$x_i(t) \leq x_j(t - T_j) + m_j \quad (2.9)$$

Avec un raisonnement similaire à celui qui a été fait dans le cas des dateurs, les transitions de

la figure 2.5b obéissent au système suivant :

$$\left\{ \begin{array}{l} x_i(t) \leq x_1(t - T_1) + m_1 \\ x_i(t) \leq x_2(t - T_2) + m_2 \\ \vdots \\ x_i(t) \leq x_n(t - T_n) + m_n \end{array} \right. \iff$$

$$x_i(t) \leq \min(x_1(t - T_1) + m_1, x_2(t - T_2) + m_2, \dots, x_n(t - T_n) + m_n) \quad (2.10)$$

Dans le dioïde $\overline{\mathbb{Z}}_{\min}$ (où $a \wedge b = \min(a, b)$ et $a \odot b = a + b$), l'inéquation 2.10 suit un modèle linéaire de la forme implicite :

$$x_i(t) \leq \bigwedge_{j=1}^n x_j(t - T_j) \odot m_j \quad (2.11)$$

Si nous considérons un fonctionnement au plus grand numéro ou rang (au plus tôt) d'un GET, le système 2.11 peut se mettre sous la forme :

$$x_i(t) = \bigwedge_{j=1}^n x_j(t - T_j) \odot m_j \quad (2.12)$$

2.3.3 Représentation d'état explicite

Si on considère un graphe d'événements temporisé multi-entrées, multi-sorties, sous l'hypothèse d'un fonctionnement au plus tôt, on obtient la forme implicite suivante sur le dioïde $\overline{\mathbb{R}}_{max}$:

$$\begin{cases} x(k) = \bigoplus_{i=0}^M A_i \otimes x(k-i) \oplus \bigoplus_{i=0}^M B_i \otimes u(k-i) \\ y(k) = \bigoplus_{i=0}^M C_i \otimes x(k-i) \end{cases} \quad (2.13)$$

où $M = \bigoplus_{j \in P} m_j^0$ avec m_j^0 est le marquage initial de chaque place, à l'état initial.

La représentation d'état sur le dioïde $\overline{\mathbb{Z}}_{min}$

$$\begin{cases} x(t) = \bigwedge_{i=0}^{\Gamma} A_i \odot x(t-i) \wedge \bigwedge_{i=0}^M B_i \odot u(t-i) \\ y(t) = \bigwedge_{i=0}^{\Gamma} C_i \odot x(t-i) \end{cases} \quad (2.14)$$

où $\Gamma = \bigoplus_{j \in P} T_j$ avec T_j est le temps de séjour minimum associé à chaque place.

Remarque : Les coefficients des matrices A_i , B_i et C_i utilisés pour la représentation des dateurs (système 2.13) et compteurs (système 2.14), sont naturellement différents.

◇

Exemple 2.1 :

Nous considérons maintenant le graphe d'événements donné par la figure 2.6.

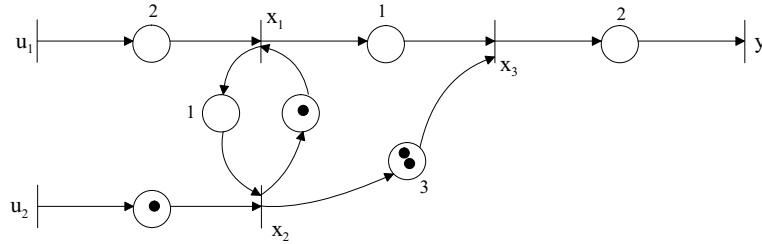


Figure 2.6 : Un graphe d'événements temporisé

Sous l'hypothèse d'un fonctionnement au plus tôt, les dateurs (sur $\overline{\mathbb{R}}_{max}$) et les compteurs (sur $\overline{\mathbb{Z}}_{min}$), décrivant le fonctionnement du graphe de la figure 2.6, vérifient respectivement les deux systèmes suivants :

$$\begin{cases} x_1(k) = 2 \otimes u_1(k) \oplus 0 \otimes x_2(k-1) \\ x_2(k) = 0 \otimes u_2(k-1) \oplus 1 \otimes x_1(k) \\ x_3(k) = 1 \otimes x_1(k) \oplus 3 \otimes x_2(k-2) \\ y(k) = 2 \otimes x_3(k) \end{cases} \quad \text{et} \quad \begin{cases} x_1(t) = 0 \odot u_1(t-2) \wedge 1 \odot x_2(t) \\ x_2(t) = 1 \odot u_2(t) \wedge 0 \odot x_1(t-1) \\ x_3(t) = 0 \odot x_1(t-1) \wedge 2 \odot x_2(t-3) \\ y(t) = 0 \odot x_3(t-2) \end{cases}$$

Pour les dateurs, nous obtenons la représentation d'état sous forme récurrente :

$$\left\{ \begin{array}{l} x(k) = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon \\ 1 & \varepsilon & \varepsilon \\ 1 & \varepsilon & \varepsilon \end{pmatrix} \otimes x(k) \oplus \begin{pmatrix} \varepsilon & e & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon \end{pmatrix} \otimes x(k-1) \oplus \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & 3 & \varepsilon \end{pmatrix} \otimes x(k-2) \\ \oplus \begin{pmatrix} 2 & \varepsilon \\ \varepsilon & \varepsilon \\ \varepsilon & \varepsilon \end{pmatrix} \otimes u(k) \oplus \begin{pmatrix} \varepsilon & \varepsilon \\ \varepsilon & e \\ \varepsilon & \varepsilon \end{pmatrix} \otimes u(k-1) \\ y(k) = \begin{pmatrix} \varepsilon & \varepsilon & 2 \end{pmatrix} \otimes x(k) \end{array} \right. \quad (2.15)$$

avec $x(k) = \begin{pmatrix} x_1(k) & x_2(k) & x_3(k) \end{pmatrix}^t$ et $u(k) = \begin{pmatrix} u_1(k) & u_2(k) \end{pmatrix}^t$

Théorème 2.12. *Si un graphe d'événements temporisé est vivant, le fonctionnement au plus tôt du système 2.13 (respectivement 2.14) peut être décrit par la forme d'équation d'état récurrente sur $\overline{\mathbb{R}}_{max}$:*

$$\begin{cases} x(k) = A \otimes x(k-1) \oplus B \otimes u(k) \\ y(k) = C \otimes x(k) \end{cases} \quad (2.16)$$

Respectivement (s'il n'y a pas de circuit sans temporisation) sur $\overline{\mathbb{Z}}_{min}$, la forme récurrente :

$$\begin{cases} x(t) = A \odot x(k-1) \wedge B \odot u(t) \\ y(t) = C \odot x(t) \end{cases} \quad (2.17)$$

Preuve : On considère l'établissement du système 2.16. Le raisonnement est dual pour le système 2.17.

- Il faut que toutes les places situées entre deux transitions internes, ne contiennent pas plus d'un jeton. Ceci est possible en décomposant chaque place qui contient plus d'un jeton, en un ensemble de places contenant chacune, un seul jeton. Ceci nécessite l'introduction de transitions intermédiaires fictives.

- De même, il faut que les places situées en aval des transitions d'entrée, soient sans jeton. Cela est possible aussi en introduisant des places et transitions intermédiaires fictives.

- Il faut que les places en amont des transitions de sortie ne contiennent pas de jetons. On obtient donc $y(k) = C \otimes x(k)$ en introduisant aussi des transition intermédiaires.

Ces différentes transformations se traduisent par la création de variables d'état. Notons que le passage d'un graphe modélisé par la forme implicite 2.13 à ce nouveau graphe se fait au détriment de la dimension du vecteur d'état $x(k)$ qui est augmentée.

Le fait que le graphe d'événements soit vivant implique que le graphe ne contienne aucun circuit sans jeton. Le sous graphe associé à A_0 est acyclique et donc le poids maximum des circuits de A_0 est égal à $\varepsilon = -\infty$ ce qui assure donc l'existence de A_0^* . Nous pouvons maintenant supprimer la partie implicite dans l'équation $x(k) = \bigoplus_{i=0}^1 A_i \otimes x(k-i) \oplus B \otimes u(k)$. Pour cela, on utilise le corollaire 1.36 qui permet de donner la plus petite solution de l'équation $x = a \otimes x \oplus b$

qui est donnée par $x = a^*b$. Nous obtenons la forme récurrente explicite suivante $x(k) = A_0^*A_1 \otimes x(k-i) \oplus A_0^*B \otimes u(k)$.

□

L'exemple 2.3.3 illustre l'explication précédente.

Exemple 2.2 :

Dans un premier temps, nous introduisons des états intermédiaires sur le graphe de la figure 2.6, nous obtenons le graphe de la figure 2.7 dont le comportement entrée-sortie est équivalent à celui de la figure 2.6.

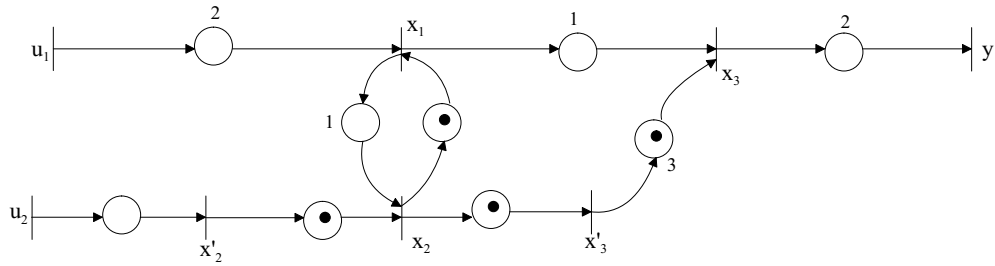


Figure 2.7 : GET équivalent au graphe de la figure 2.6

La nouvelle forme d'état explicite sur $\overline{\mathbb{R}}_{max}$ qui représente le graphe de la figure 2.6 est :

$$\begin{cases} x(k) = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ 1 & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & 1 \end{pmatrix} x(k) \oplus \begin{pmatrix} \varepsilon & e & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & e & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & e & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & 3 & \varepsilon \end{pmatrix} x(k-1) \oplus \begin{pmatrix} 2 & \varepsilon \\ \varepsilon & e \\ \varepsilon & \varepsilon \\ \varepsilon & \varepsilon \\ \varepsilon & \varepsilon \end{pmatrix} u(k) \\ y(k) = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon & 2 \end{pmatrix} x(k) \end{cases} \quad (2.18)$$

avec $x(k) = \left(x_1(k) \quad x'_e(k) \quad x_2(k) \quad x'_3(k) \quad x_3(k) \right)^t$

Nous considérons donc le système 2.18 et en appliquant le théorème 2.12 nous aboutissons à la forme récurrente explicite qui représente le graphe de la figure 2.6 :

$$\begin{cases} x(k) = \begin{pmatrix} \varepsilon & e & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & 1 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & e & \varepsilon & \varepsilon \\ \varepsilon & 1 & \varepsilon & 3 & \varepsilon \end{pmatrix} x(k-1) \oplus \begin{pmatrix} 2 & \varepsilon \\ \varepsilon & e \\ 3 & \varepsilon \\ \varepsilon & \varepsilon \\ 3 & \varepsilon \end{pmatrix} u(k) \\ y(k) = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon & 2 \end{pmatrix} x(k) \end{cases}$$

2.4 Graphes d'événements du type intervalle

Dans les modèles suivants, un intervalle de temporisation peut être associé :

1. soit aux places (graphes d'événements p-temporels)
2. soit aux transitions (graphes d'événements t-temporels)
3. soit aux arcs (graphes d'événements à arcs temporels et à flux temporels)

2.4.1 Graphes d'événements p-temporels

2.4.1.1 Présentation des graphes d'événements p-temporels

Développés dans [KDCD96] et [Kha97], les réseaux de Petri p-temporels ont pour objectif de modéliser des systèmes dans lesquels le temps prend un aspect non déterministe. En effet, le temps de séjour associé à chaque place p_i n'est plus considéré déterminé comme pour les GET, mais appartient à un intervalle de temps IS_i . Nous consacrons la première partie de cette section au rappel des définitions et notations relatives aux réseaux de Petri p-temporels. Comme domaines d'applications, on peut citer la ligne de galvanoplastie, les chaînes alimentaires, les manufactures comportant un four, les traitements thermiques,...

Définition 2.13. *Un réseau de Petri p-temporel est défini par le doublet $\langle R, IS \rangle$ où :*

R est un réseau de Petri marqué (places-transitions)

$$IS : P \longrightarrow (\mathbb{R}^+ \cup \{0\}) \times (\mathbb{R}^+ \cup \{+\infty\})$$

$$p_i \longrightarrow IS_i = [a_i, b_i] \text{ avec } 0 \leq a_i \leq b_i$$

$[a_i, b_i]$ définit l'intervalle statique de temps de séjour d'une marque dans la place p_i .

Le temps de séjour d'un jeton est donc compris dans un intervalle de temps. Le jeton qui arrive dans une place est indisponible (ne peut pas franchir la transition en aval) pendant un certain temps. Il doit passer un temps de séjour minimum correspondant à la borne minimale de l'intervalle. Le jeton atteint un âge de maturité (qui lui permet de franchir une transition) après a_i unités de temps après son arrivée dans la place. Il reste dans cet état de disponibilité durant $b_i - a_i$ unités de temps. Une des spécificités des réseaux de Petri p-temporels est la possibilité de la mort de jetons dont le fonctionnement est le suivant. Après un séjour de b_i (borne max de l'intervalle) unités de temps dans la place, le jeton se trouve dans l'obligation de quitter cette dernière, sinon, il se retrouve dans un état de mort. Autrement dit, le jeton ne peut plus participer aux franchissements, et cela peut générer d'éventuels dysfonctionnements du système dans le futur si par exemple, le jeton représente une ressource importante. Par conséquent, ce modèle va nécessiter un contrôle des tirs des transitions afin d'éviter la mort des jetons.

2.4.1.2 Dateurs et intervalles de tirs admissibles

Nous rappelons que $x_i(k)$ correspond à une date de tir comme dans le cas des graphes d'événements temporisés. Nous adoptons les notations suivantes : \oplus opérateur maximum, \wedge opérateur minimum et \otimes (respectivement \odot) produit dans $\overline{\mathbb{Z}}_{max}$ (respectivement dans $\overline{\mathbb{Z}}_{min}$). La définition suivante va nous permettre d'introduire la notion d'admissibilité qui sera utilisée dans la suite pour la modélisation des graphes d'événements p-temporels.

Définition 2.14. (Fonctionnement admissible)

Un fonctionnement est dit admissible pour un réseau de Petri p-temporel, lorsque l'évolution dynamique préserve la vivacité des jetons. En conséquence, il ne mène pas le système à un état de blocage.

La proposition suivante donne les conditions que doivent satisfaire les dates de tirs des transitions pour garder un fonctionnement admissible. Ces conditions se présentent sous forme d'un modèle d'intervalle.

Proposition 2.15. Pour qu'un graphe d'événements p-temporel respecte un fonctionnement admissible, la modélisation correspondant à chacune de ses transitions, qui n'est pas une transition source, est donnée par l'expression suivante (qui doit être vérifiée) :

$$\bigoplus_{j \in s_i} (x_j(k - m_j) + a_j) \leq x_i(k) \leq \bigwedge_{j \in s_i} (x_j(k - m_j) + b_j) \tag{2.19}$$

avec s_i le nombre des places d'entrées de la transition x_i et m_j le marquage initial de chacune de ces places.

Preuve :

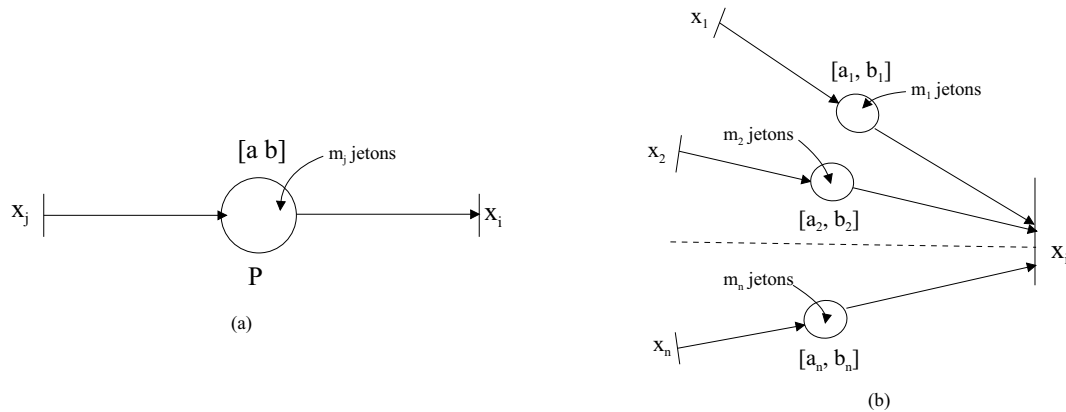


Figure 2.8 : Parties d'un graphe d'événements p-temporel

Dans un graphe d'événements p-temporel, en dehors des transitions sources, une transition (interne ou de sortie) peut avoir une seule place en amont ou, être une transition de synchronisation.

Nous associons un dateur $x_i(k)$ pour chaque transition, qui a une seule place p_j en amont. Cette dernière se trouve en aval d'une transition qui a aussi pour dateur $x_j(k)$ (voir la figure 2.8a). Sous l'hypothèse d'un fonctionnement FIFO, les m_j marques qui franchissent x_j à la date $x_j(k - m_j)$ sont prêtes pour franchir x_i au plus tôt à la date $x_j(k - m_j) + a_j$, et au plus tard, doivent quitter cette place à la date $x_j(k - m_j) + b_j$. D'où $x_i(k) \geq x_j(k - m_j) + a_j$ et $x_i(k) \leq x_j(k - m_j) + b_j$. Si x_i est une transition de synchronisation (voir la figure 2.8b), $x_i(k)$ doit satisfaire les deux systèmes d'inéquations suivants :

$$\begin{cases} x_i(k) \geq x_1(k - m_1) + a_1 \\ x_i(k) \geq x_2(k - m_2) + a_2 \\ \vdots \\ x_i(k) \geq x_n(k - m_n) + a_n \end{cases} \quad \text{et} \quad \begin{cases} x_i(k) \leq x_1(k - m_1) + b_1 \\ x_i(k) \leq x_2(k - m_2) + b_2 \\ \vdots \\ x_i(k) \leq x_n(k - m_n) + b_n \end{cases}$$

Dans le dioïde $\overline{\mathbb{R}}_{\max}$, le premier système se réécrit : $x_i(k) \geq \bigoplus_{j=1}^n a_j \otimes x_j(k - m_j)$

Dans le dioïde $\overline{\mathbb{Z}}_{\min}$, le deuxième système se réécrit : $x_i(k) \leq \bigwedge_{j=1}^n b_j \odot x_j(k - m_j)$

$$\implies \bigoplus_{j=1}^n a_j \otimes x_j(k - m_j) \leq x_i(k) \leq \bigwedge_{j=1}^n b_j \odot x_j(k - m_j)$$

□

Pour un graphe d'événements p-temporel, le modèle d'état correspondant à un fonctionnement admissible est :

$$\begin{cases} \bigoplus_{i=1}^M A_i^- \otimes x(k - i) \oplus \bigoplus_{i=1}^M B_i^- \otimes u(k - i) \leq x(k) \leq \bigwedge_{i=1}^M A_i^+ \odot x(k - i) \wedge \bigwedge_{i=1}^M B_i^+ \odot u(k - i) \\ \bigoplus_{i=1}^M C_i^- \otimes x(k - i) \leq y(k) \leq \bigwedge_{i=1}^M C_i^+ \odot x(k - i) \end{cases} \quad (2.20)$$

La présentation d'état en séries formelles γ est donnée par :

$$\begin{cases} \bigoplus_{i=1}^M A_i^- \otimes x(\gamma)\gamma^i \oplus \bigoplus_{i=1}^M B_i^- \otimes u(\gamma)\gamma^i \leq x(\gamma) \leq \bigwedge_{i=1}^M A_i^+ \odot x(\gamma)\gamma^i \wedge \bigwedge_{i=1}^M B_i^+ \odot u(\gamma)\gamma^i \\ \bigoplus_{i=1}^M C_i^- \otimes x(\gamma)\gamma^i \leq y(\gamma) \leq \bigwedge_{i=1}^M C_i^+ \odot x(\gamma)\gamma^i \end{cases} \quad (2.21)$$

Les coefficients des matrices A_i^- , B_i^- et C_i^- (respectivement A_i^+ , B_i^+ et C_i^+) dans \mathbb{R}_{\max} représentent les bornes inférieures (respectivement supérieures) des intervalles de temps associés au places. Ces coefficients sont positifs.

Afin d'illustrer les différentes démarches de modélisation des graphes d'événements p-temporels, nous présentons maintenant l'exemple d'un atelier de production que nous avons développé.

Exemple 2.3 :

Une cellule de production de pots de peinture est donnée par la figure 2.9. Elle est constituée de deux machines. La première machine M_1 s'occupe du malaxage (mélange des différentes couleurs) et du chargement de la peinture dans les pots. Cette opération dure au minimum 2 unités de temps, et au maximum 7 unités de temps (sinon, il y aura un débordement du produit et le pot devra être retiré de la chaîne de production). Le temps de fermeture des pots, remplis par la machine M_2 , est compris entre 1 au minimum (sinon le pot sera mal fermé) et 6 unités de temps au maximum (sinon, le pot sera fermé avec trop de force). Un premier convoyeur a besoin entre 2 et 5 unités de temps pour transporter les produits sortant de M_1 vers une zone de transition. Ces derniers transitent par x_3 pour atteindre un deuxième convoyeur qui met entre 1 et 4 unités de temps pour transporter les produits de la zone de transition vers la machine M_1 . On impose aux deux machines qu'ils ne restent pas libres au delà de 4 unités de temps. Les deux convoyeurs mettent un temps minimum de 1 unité temps pour retourner à leurs états initiaux une fois qu'ils ont livré un produit. Ils peuvent aussi attendre un produit jusqu'à l'infini noté T .

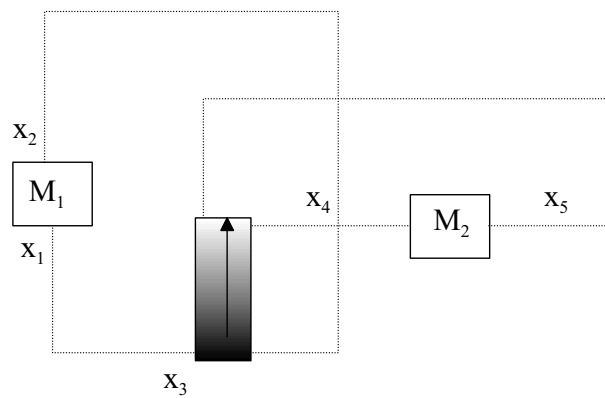


Figure 2.9 : Cellule de production des pots de peinture

Afin de modéliser ce système, on utilise donc le graphe d'événements p-temporel représenté dans la figure 2.10. Ce dernier va permettre aussi de traduire toutes les contraintes temporelles, correspondant à chacune des opérations relatives au systèmes de la figure 2.9, décrites auparavant.

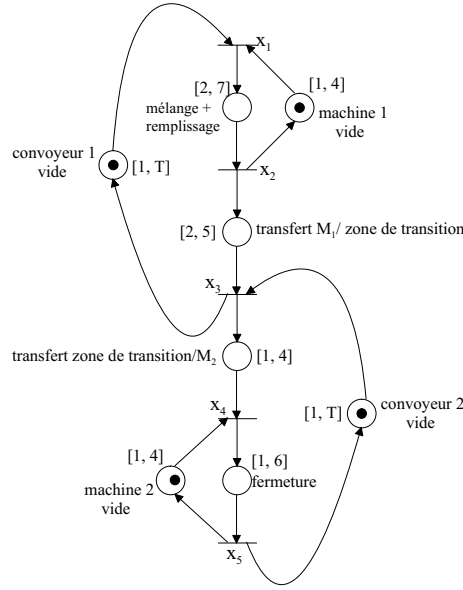


Figure 2.10 : Graphe d'événements p-temporel qui représente le système de la figure 2.9

En appliquant la proposition 2.15 au graphe de la figure 2.10, on peut établir le système suivant :

$$\left\{ \begin{array}{l} (1 \otimes x_2(k-1)) \oplus (1 \otimes x_3(k-1)) \leq x_1(k) \leq (4 \odot x_2(k-1)) \wedge (T \odot x_3(k-1)) \\ 2 \otimes x_1(k) \leq x_2(k) \leq 7 \odot x_1(k) \\ (2 \otimes x_2(k)) \oplus (1 \otimes x_5(k-1)) \leq x_3(k) \leq (5 \odot x_2(k)) \wedge (T \odot x_5(k-1)) \\ (1 \otimes x_3(k)) \oplus (1 \otimes x_5(k-1)) \leq x_4(k) \leq (4 \odot x_3(k)) \wedge (4 \odot x_5(k-1)) \\ 1 \otimes x_1(k) \leq x_5(k) \leq 6 \odot x_1(k) \end{array} \right. \quad (2.22)$$

Si on pose $x(k) = (x_1, x_2, x_3, x_4, x_5)^t$ un vecteur colonne, le système 2.22 peut se mettre sous la forme matricielle :

$$\left\{ \begin{array}{l} \left(\begin{array}{ccccc} \varepsilon & 1 & 1 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & 1 \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & 1 \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{array} \right) \otimes x(k-1) \oplus \left(\begin{array}{ccccc} \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ 2 & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & 2 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 1 & \varepsilon & \varepsilon \\ 1 & \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{array} \right) \otimes x(k) \leq x(k) \\ \\ x(k) \leq \left(\begin{array}{ccccc} \varepsilon & 4 & T & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & T \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & 4 \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{array} \right) \odot x(k-1) \wedge \left(\begin{array}{ccccc} \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ 7 & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & 5 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 4 & \varepsilon & \varepsilon \\ 6 & \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{array} \right) \odot x(k) \end{array} \right. \quad (2.23)$$

La réécriture sous forme de séries formelles en γ pour cet exemple est donnée ci-dessous et présente une forme plus réduite que celle donnée par 2.23 :

$$\left\{ \begin{array}{l} \left(\begin{array}{ccccc} \varepsilon & 1\gamma^1 & 1\gamma^1 & \varepsilon & \varepsilon \\ 2\gamma^0 & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & 2\gamma^0 & \varepsilon & \varepsilon & 1\gamma^1 \\ \varepsilon & \varepsilon & 1\gamma^0 & \varepsilon & 1\gamma^1 \\ 1\gamma^0 & \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{array} \right) \otimes x(\gamma) \leq x(\gamma) \\ \\ x(\gamma) \leq \left(\begin{array}{ccccc} T & 4\gamma^1 & T\gamma^1 & T & T \\ 7\gamma^0 & T & T & T & T \\ T & 5\gamma^0 & T & T & T\gamma^1 \\ T & T & 4\gamma^0 & T & 4\gamma^1 \\ 6\gamma^0 & T & T & T & T \end{array} \right) \odot x(\gamma) \end{array} \right. \quad (2.24)$$

2.4.1.3 Nouvelle formulation du modèle d'état

Nous proposons ici une reformulation du modèle 2.20 qui sera utilisée ultérieurement. L'objectif est de trouver une nouvelle forme du modèle représentant le fonctionnement d'un graphe d'événements p-temporel ne comportent que des termes (max, +). Nous adoptons ici les notations suivantes : $\leftarrow x_i = \{x_j \text{ tel que } x_j \in \bullet(\bullet x_i)\}$ est l'ensemble des transitions x_j qui se trouvent en amont des places amont de la transition x_i .

$x_i^{\rightarrow} = \{x_j \text{ tel que } x_j \in (x_i^{\bullet})\bullet\}$ est l'ensemble des transitions x_j qui se situent en aval des places aval de la transition x_i .

m_{ij} (respectivement m_{ji}) le marquage de la place qui existe entre x_j et x_i à qui on associe l'intervalle de temporisation $[a_{ij}, b_{ij}]$ avec $x_j \in \leftarrow x_i$ (respectivement $x_j \in x_i^{\rightarrow}$).

D'après le système 2.19 on en déduit que :

$$(\forall x_j \in \leftarrow x_i) x_i(k) \geq x_j(k - m_{ij}) + a_{ij} \text{ et } (\forall x_j \in \leftarrow x_i) x_i(k) \leq x_j(k - m_{ij}) + b_{ij} \iff (\forall x_j \in x_i^{\rightarrow}) x_j(k + m_{ji}) - b_{ji} \leq x_i(k)$$

Par conséquent, on obtient le modèle suivant décrit uniquement par des fonctions (max, +) :

$$\left\{ \begin{array}{l} x_i(k) \geq \bigoplus_{x_j \in \leftarrow x_i} x_j(k - m_{ij}) \otimes a_{ij} \\ x_i(k) \geq \bigoplus_{x_j \in x_i^{\rightarrow}} x_j(k + m_{ji}) \otimes (-b_{ji}) \end{array} \right. \iff x_i(k) \geq \bigoplus_{x_j \in \leftarrow x_i} a_{ij}^- \otimes x_j(k - m_{ij}) \oplus \bigoplus_{x_j \in x_i^{\rightarrow}} a_{ij}^+ \otimes x_j(k + m_{ji}) \quad (2.25)$$

avec $a_{ij}^- = a_{ij}$ et $a_{ij}^+ = -b_{ji}$, $a_{ij}^- \in \mathbb{R}^+$, $a_{ij}^+ \in \mathbb{R}^-$

Ainsi, le dernier modèle uniquement du type (max, +) donné par 2.25 est équivalent au modèle d'intervalle initial de type ((max, +), (min, +)) et qui est représenté par le système 2.19.

Si x_i est considérée comme transition d'entrée, alors, on a $\leftarrow x_i = \emptyset$. De façon similaire, si x_i est prise comme une transition de sortie, alors, on a $x_i^{\rightarrow} = \emptyset$. La prise en compte de l'ajout de ces transitions au graphe d'événement initial ne modifie pas le comportement de ce dernier. Nous détaillons maintenant les différentes étapes suivies pour arriver au nouveau modèle 2.26.

Pour chaque transition d'entrée, nous introduisons une place en amont (d'entrée), avec sa transition d'entrée notée u . Cette place ne contient pas de jetons et nous lui associons un intervalle de temporisation nul $[0, 0]$.

Pour $x_j \in u_i^{\rightarrow}$, $x_j(k) \geq u_i(k) + a_{ij}$ avec $a_{ij} = 0$ et $x_j(k) \leq u_i(k) + b_{ij}$ avec $b_{ij} = 0$ ou $u_i(k) \geq x_j(k) - b_{ij}$ (avec $b_{ij} = 0$).

D'une manière similaire, pour chaque transition de sortie, une place de sortie et sa transition en aval y sont introduites telles que la place de sortie ne contient pas de jetons et porte une temporisation nulle $[0, 0]$.

Pour tout $x_j \in \leftarrow y_i$, $y_i(k) \geq x_j(k) + a_{ji}$ avec $a_{ji} = 0$ et $y_i(k) \leq x_j(k) + b_{ij}$ avec $b_{ij} = 0$ ou $x_j(k) \geq y_i(k) - b_{ij}$.

Aussi, nous savons que pour chaque transition d'entrée u_i , $\text{card}(\leftarrow u_i) = 0$ et $\text{card}(u_i^{\rightarrow}) = 1$ et pour chaque transition de sortie y_i , $\text{card}(\leftarrow y_i) = 1$ et $\text{card}(y_i^{\rightarrow}) = 0$. D'où la représentation max-plus suivante :

$$\begin{aligned} (x_i \in u_j^{\rightarrow}) \quad x_i(k) &\geq b_{ij}^- \otimes u_j(k), & (x_j \in u_i^{\rightarrow}) \quad u_i(k) &\geq b_{ij}^+ \otimes x_j(k) \\ (x_j \in \leftarrow y_i) \quad y_i(k) &\geq c_{ij}^- \otimes x_j(k), & (x_i \in \leftarrow y_j) \quad x_i(k) &\geq c_{ij}^+ \otimes y_j(k) \end{aligned}$$

avec $b_{ij}^- = a_{ji} = 0$, $b_{ij}^+ = -b_{ij} = 0$, $c_{ij}^- = a_{ji} = 0$, $c_{ij}^+ = -b_{ij} = 0$.

Nous obtenons alors la forme matricielle en séries formelle en γ :

$$\begin{aligned} x(\gamma) &\geq B^- \otimes u(\gamma), & u(\gamma) &\geq B^+ \otimes x(\gamma) \\ y(\gamma) &\geq C^- \otimes x(\gamma), & x(\gamma) &\geq C^+ \otimes y(\gamma) \end{aligned}$$

La nouvelle représentation d'état est déduite à partir du système 2.25 auquel il faut ajouter une condition de non décroissance de x . L'introduction de $x \geq \gamma^1 x(\gamma)$ se traduit par un changement au niveau des coefficients de la diagonale de la matrice A_1^- . La nouvelle forme d'état représentant un graphe d'événements p-temporel est donnée par :

$$x(\gamma) \geq \bigoplus_{0 \leq i \leq M} A_i^- \otimes \gamma^i x(\gamma) \oplus \bigoplus_{0 \leq i \leq M} A_i^+ \otimes \gamma^{-i} x(\gamma) \quad (2.26)$$

avec : $M = \bigoplus_{i \in P} m_i$, pour $k = m(\bullet(x_i))$, si $x_j \in \leftarrow x_i$, $(A_k^-)_{ij} = e \oplus a_{ij}^-$ si $k = 1$, $i = j$ et $(A_k^-)_{ij} = a_{ij}^-$ sinon, pour $k = m((x_i)\bullet)$, $(A_k^+)_{ij} = a_{ij}^+$ si $x_j \in x_i^{\rightarrow}$.

Du fait de la spécificité de la mort des jetons dans le cas des graphes d'événements p-temporels, nous définissons le fonctionnement au plus tôt et au plus tard du système par le mode d'évolution suivant.

2.4.1.4 Fonctionnement au plus tôt et au plus tard d'un graphe d'événements p-temporel

Un fonctionnement au plus tôt (respectivement au plus tard) du système correspond à un franchissement pour chacune des transitions (sources, internes et puits) à une date au plus tôt telle qu'avant (respectivement au delà de) cette dernière, le système global (comportant toutes les transitions) perde sa propriété de fonctionnement admissible.

Nous considérons le RdP de la figure 2.11 qui est un graphe d'événements p-temporel vivant. A travers l'analyse de cet exemple, nous allons montrer comment un graphe d'événements p-temporel doit se comporter en fonctionnement au plus tôt (et au plus tard) sans qu'il perde la propriété de fonctionnement admissible.

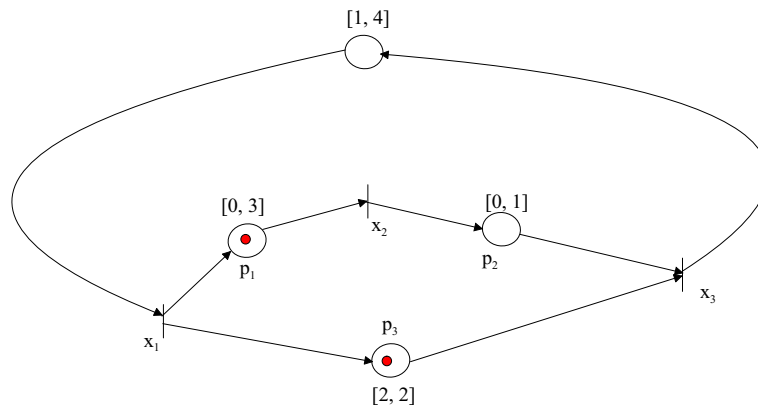


Figure 2.11 : Exemple d'un graphe d'événements p-temporel vivant

Supposons que x_2 ait un fonctionnement local au plus tôt, alors le jeton de la place p_1 peut la quitter à l'instant $t = 0$. En arrivant à p_2 , le jeton mourra avant de franchir x_4 , car il ne pourra pas attendre (il peut rester à p_2 , jusqu'à $t = 1$) le jeton qui doit venir de p_3 . On assistera alors à une désynchronisation qui mènera le système global à une situation de blocage.

De même, avec une hypothèse d'un fonctionnement au plus tard pour la transition x_2 . Le jeton de la place p_3 mourra puisque à l'instant $t = 2$ où il doit franchir x_4 , la place p_2 ne contiendra aucun jeton car le jeton de p_1 doit attendre $t = 3$ pour franchir x_2 . Le système global se retrouve dans ce cas dans une situation de blocage.

A l'opposé, franchir la transition x_2 à un instant entre $t = 1$ et $t = 2$ garantira un fonctionnement admissible ce qui montre que le graphe d'événements est vivant. Sur cet exemple, ni le fonctionnement local au plus tôt, ni le fonctionnement local au plus tard, n'aboutissent à un fonctionnement viable, bien que ce système soit vivant.

Nous concluons que la notion du fonctionnement au plus tôt (et au plus tard), pour les graphes d'événements p-temporels, ne peut être prise localement (entre une place et une transition) mais globalement. Cette notion diffère de celles des GET. En d'autres termes, l'égalité 2.7 peut ne pas être vérifiée pour un fonctionnement au plus tôt. La même remarque est valable aussi dans le cas d'un fonctionnement au plus tard. La remarque sur le fonctionnement au plus tôt est également citée dans [Bon05] et [DDC05] sans que ce comportement ne soit formalisé explicitement. En conclusion, la notion du fonctionnement au plus tôt ne peut être définie de manière analogue aux GET. L'égalité $\bigoplus_{j=1}^n a_j \otimes x_j(k - m_j) = x_i(k)$ ne représente donc pas un fonctionnement au plus tôt pour les graphes d'événements p-temporels comme c'est le cas pour les GET. De même, l'égalité $x_i(k) = \bigwedge_{j=1}^n b_j \odot x_j(k - m_j)$ ne peut traduire un fonctionnement au plus tard pour les graphes d'événements p-temporels.

2.4.2 Graphes d'événements t-temporels

Dans ce modèle, les activités sont liées aux transitions. La figure 2.12 donne un exemple d'un graphe d'événements t-temporel. Ils sont capables de représenter par exemples des systèmes informatiques qui ont des temps de réponse variables [BD91]. Un autre problème pratique pour lequel les graphes d'événements t-temporels sont un outil de modélisation intéressant est le système du "chien de garde" ou alarme. Un des objectifs de ce système est de surveiller la date de production d'une action. Suivant cette date et un paramètre donné, l'alarme se déclenche ou se désactive [Boy01]. A l'opposé des graphes d'événements p-temporels, les graphes d'événements t-temporels ne permettent pas de modéliser des systèmes pour lesquels la durée d'une tâche est comprise dans un intervalle .

Définition 2.16. (Graphe d'événements t-temporel) Un graphe d'événements t-temporel est un doublet $\langle GE, IS_t \rangle$ où GE est un graphe d'événements marqué

$$IS_t : T \longrightarrow (\mathbb{R}^+ \cup \{0\}) \times (\mathbb{R}^+ \cup \{+\infty\})$$

$$t_i \longrightarrow IS_i = [a_i, b_i] \text{ avec } 0 \leq a_i \leq b_i$$

L'intervalle statique $[a_i, b_i]$ associant une durée minimale a_i et une durée maximale b_i à chaque transition t_i .

Mise en équation des graphes d'événements t-temporels

Pour qu'une transition t_i soit franchie, il faut qu'elle reste sensibilisée durant au moins a_i unités de temps et au plus b_i unités de temps. L'intervalle de tir d'une transition est relatif à la date où la transition devient sensibilisée (validée). Si par exemple une transition t_j est validée à l'instant τ , sachant que $[a_j, b_j]$ est l'intervalle statique associé à t_j . Alors, t_j peut être tirée entre $\tau + a_j$ et $\tau + b_j$. Nous étudions maintenant le cas d'une transition de synchronisation comme l'illustre la figure 2.12. Nous constatons tout d'abord que le déclenchement de l'activation de

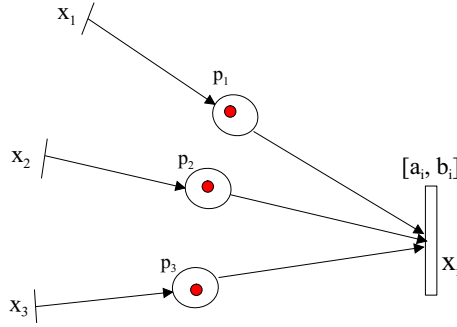


Figure 2.12 : Graphe d'événements t-temporel

l'horloge⁶ pour la transition x_i est liée directement à la condition logique⁷. Par conséquent, c'est le dernier jeton arrivant à l'une des places (p_1 , p_2 et p_3) qui marquera le début de la sensibilisation de la transition x_i . A partir de ce moment, la sensibilisation doit être maintenue continuellement pendant a_i unités de temps, et cela peut aller jusqu'à b_i unités de temps. Le franchissement peut avoir lieu entre deux dates représentant la date de tir au plus tôt, et la date de tir au plus tard. Ceci nous amène au système d'inéquations suivant :

$$\begin{cases} (x_1(k-1) + a_i) \oplus (x_2(k-1) + a_i) \oplus (x_3(k-1) + a_i) \leq x_i(k) \\ x_i(k) \leq (x_1(k-1) + b_i) \oplus (x_2(k-1) + b_i) \oplus (x_3(k-1) + b_i) \end{cases} \quad (2.27)$$

Plus généralement, dans le cas d'un graphe d'événements t-temporels, le modèle d'état correspondant est le suivant :

$$\bigoplus_{i=1}^M A_i \otimes x(k-i) \leq X(k) \leq \bigoplus_{i=1}^M B_i \otimes x(k-i) \quad (2.28)$$

2.4.3 Graphes d'événements à arcs temporels

Présentés dans [Wal83], les réseaux de Petri à arcs temporels considèrent que la temporisation est portée par les arcs comme les RdP à flux temporels qui seront introduits ultérieurement. Le problème de la mort des jetons connu pour les graphes d'événements p-temporels, surgit aussi pour les graphes d'événements à arcs temporels.

Définition 2.17. Graphes d'événements à arcs temporels Un graphe d'événements à arcs temporels est un doublet $\langle GE, Iarc \rangle$ où :

GE est un graphe d'événements marqué (places-transitions)

$Iarc : \mathcal{A} \subset (P \times T) \longrightarrow (\mathbb{R}^+ \cup \{0\}) \times (\mathbb{R}^+ \cup \{+\infty\})$

$arc_i \longrightarrow Iarc_i = [a_i, b_i]$ avec $0 \leq a_i \leq b_i$

$[a_i, b_i]$ est l'intervalle temporel statique associé à chaque arc i incident à une transition.

⁶on dit aussi la condition temporelle, c-à-d la date du commencement de la sensibilisation du x_i

⁷il traduit la condition qui doit être satisfaite au niveau du marquage

En réalité, le comportement des graphes d'événements à arcs temporels est identique aux graphes d'événements p-temporels et l'ensemble des explications peut être transposé.

Mise en équation des graphes d'événements à arcs temporels

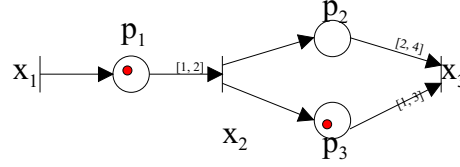


Figure 2.13 : Un exemple simple d'un graphe d'événements à arcs temporels

Nous commençons par la mise en équation d'un exemple simple (figure 2.13), avant de donner le modèle algébrique général qui peut représenter un graphe d'événements à arcs temporels. A l'état initial, aucune transition ne peut être franchie. Il faut attendre 1 unité de temps pour tirer la transition x_2 . Si x_2 est tirée à une date strictement supérieure à $t = 1$, il sera impossible de franchir x_3 puisque le jeton de la place p_3 aura dépassé son temps de séjour maximum (3 unités de temps), et sera donc mort. La synchronisation au niveau de la transition x_3 ne peut avoir lieu. Dans ce cas, le fonctionnement du graphe de la figure 2.13 est qualifié de non admissible. Le jeton qui franchit x_1 et qui arrive à la place p_1 doit y faire un séjour minimum de 1 unité de temps, et doit y quitter au plus tard après 2 unités de temps après son arrivée, d'où on a : $x_1(k-1) + 1 \leq x_2(k) \leq x_1(k-1) + 2$. De la même manière, on montre que : $(x_2(k-1) + 1) \leq x_3(k) \leq (x_2(k-1) + 3)$ et $(x_2(k) + 2) \leq x_3(k) \leq (x_2(k) + 4)$

Nous aboutissons au système d'équations suivant :

$$\begin{cases} x_1(k-1) + 1 \leq x_2(k) \leq x_1(k-1) + 2 \\ (x_2(k-1) + 1) \oplus (x_2(k) + 2) \leq x_3(k) \leq (x_2(k-1) + 3) \wedge (x_2(k) + 4) \end{cases} \quad (2.29)$$

Dans le cas général, un graphe d'événements à arcs temporels admet la même représentation d'état qu'un graphe d'événements p-temporel. D'où le modèle :

$$\begin{cases} \bigoplus_{i=1}^M A_i^- \otimes x(k-i) \oplus \bigoplus_{i=1}^M B_i^- \otimes u(k-i) \leq x(k) \leq \bigwedge_{i=1}^M A_i^+ \odot x(k-i) \wedge \bigwedge_{i=1}^M B_i^+ \odot u(k-i) \\ \bigoplus_{i=1}^M C_i^- \otimes x(k-i) \leq y(k) \leq \bigwedge_{i=1}^M C_i^+ \odot x(k-i) \end{cases} \quad (2.30)$$

2.4.4 Graphes d'événements à flux temporels

Les graphes d'événements à flux temporels se présentent comme un formalisme semblable aux graphes d'événements à arcs temporels, avec ainsi, des intervalles de temporisations sur les arcs⁸. L'introduction d'une sémantique de tir va permettre d'obtenir des formes sophistiquées.

⁸aucune politique ou sémantique de tir n'est affectée aux transitions dans le cas des graphes d'événements à arcs temporels

2.4.4.1 Définition et notations

Introduits dans [DS93], [Sén96] et [Dia01] avec pour objectif la modélisation de scénarios multimédias, les réseaux de Petri à flux temporels permettent de modéliser des synchronisations temporelles complexes. La sémantique de ces dernières, définit la règle qui montre comment les différentes parties d'un système interagissent. Autrement dit, on va pouvoir, à partir de différentes entités d'un système réel, considérer le système complet, résultat de la synchronisation des différentes parties.

En général, la synchronisation des systèmes qui présente des contraintes temporelles, possède deux aspects : la partie logique et la partie temporelle. La première correspond à une synchronisation logique sur une transition t sans le temps tandis que la deuxième est une action temporelle relative aux places d'entrée de t . Pour résoudre le problème de la synchronisation temporelle entre flux, deux principales nouveautés sont introduites [Boy01] :

- 1- une condition temporelle sur les arcs, c'est-à-dire une condition par couple (place, transition)
- 2- différents modes de synchronisations entre flux, à choisir par l'utilisateur.

La définition des règles de synchronisation (sémantiques de tir) entre plusieurs flux, est un choix qui peut favoriser un flux par rapport aux autres et peut être fait par la borne min et la borne max de l'intervalle de tir [Boy01]. On peut choisir de tirer "au plus tôt" (sémantique OU), ou bien "au plus tard" (attendre le dernier arrivé, sémantique ET), ou un flux défini par le concepteur (un flux plus important : sémantique Maître). Les règles (sémantiques) de tir seront présentées par la suite.

Définition 2.18. *Un graphe d'événements à flux temporel est un triplet $\langle GE, IT, SYN \rangle$ dans lequel :*

- GE est un graphe d'événements places-transitions,
- IT est une application qui associe, à chaque arc sortant d'une place un intervalle $[a_j, b_j]$ de validité temporelle (condition temporelle sur le tir) :

$$IT : \mathcal{A}_j \longrightarrow \mathbb{R}^+ \times (\mathbb{R}^+ \cup \{\infty\})$$

- SYN est une fonction de synchronisation et de typage des transitions. Elle associe à chaque transition un type, c'est-à-dire qu'elle définit la règle de tir associée à une transition (sémantique de tir)

$$SYN : T \longrightarrow \{\text{sémantiques}\}$$

On dit qu'un arc \mathcal{A}_j devient sensibilisé si et seulement si, sa place d'entrée p_j reçoit un jeton. Si par exemple, un arc est sensibilisé à l'instant τ , le jeton lié à ce dernier doit quitter sa place pendant l'intervalle $[\tau + a_j, \tau + b_j]$.

Par la suite, on considère que chaque arc des graphes d'événements à flux temporels considéré

dans ce mémoire, présente une contrainte temporelle.

Définition 2.19. *Soit une transition de synchronisation t d'un graphe d'événements à flux temporel. Nous considérons que $\forall p_j \in \bullet t, M(p_j) \geq 1$ et τ^f est le temps absolu de tir de t .*

1. t est de type *Et-Pur*, si et seulement si :

$$\forall p_j \in \bullet t, (\tau_j + a_j) \leq \tau^f \leq (\tau_j + b_j)$$

2. t est de type *Et-Faible*, si et seulement si :

$$\forall p_j \in \bullet t \quad (\tau_j + a_j) \leq \tau^f \quad \text{et} \quad \exists p_j \in \bullet t, \tau^f \leq (\tau_j + b_j)$$

3. t est de type *Ou-Fort*, si et seulement si on a :

$$\exists p_j \in \bullet t, \quad (\tau_j + a_j) \leq \tau^f \quad \text{et} \quad \forall p_j \in \bullet t, \quad \tau^f \leq (\tau_j + b_j)$$

4. t est de type *Ou*, si et seulement si on a :

$$\exists p_j \in \bullet t, \quad (\tau_j + a_j) \leq \tau^f \quad \text{et} \quad \exists p_j \in \bullet t, \quad \tau^f \leq (\tau_j + b_j)$$

Définition 2.20. *Une transition t de type Maître, définie à partir de l'un de ses arcs entrants, soit (p_m, t) dit Maître, est tirable au temps τ^f si et seulement si pour cet arc (p_m, t) , on a :*

$$(\tau_m + a_m) \leq \tau^f \quad \text{et} \quad \tau^f \leq (\tau_m + b_m)$$

Rappelons que la transition t est sensibilisée sous un point de vue logique.

Introduire d'autres types de règles de tir adaptées à des cas de systèmes réels reste possible pour répondre aux besoins de modélisation. Un enrichissement de la règle Maître par une composition avec d'autres sémantiques, a été proposé dans [Dia01] et [Boy01]. Ceci a conduit à la mise en évidence des sémantiques : Et-Maître, Ou-maître, Maître-Faible et Maître-Fort. Pour éclaircir la différence qui existe entre les types de sémantiques présentées ci-dessus, nous donnons la figure 2.14b qui illustre les différents types de synchronisation que l'on peut associer à la transition x de la figure 2.14a. Ainsi, d'autres sémantiques sont également possibles.

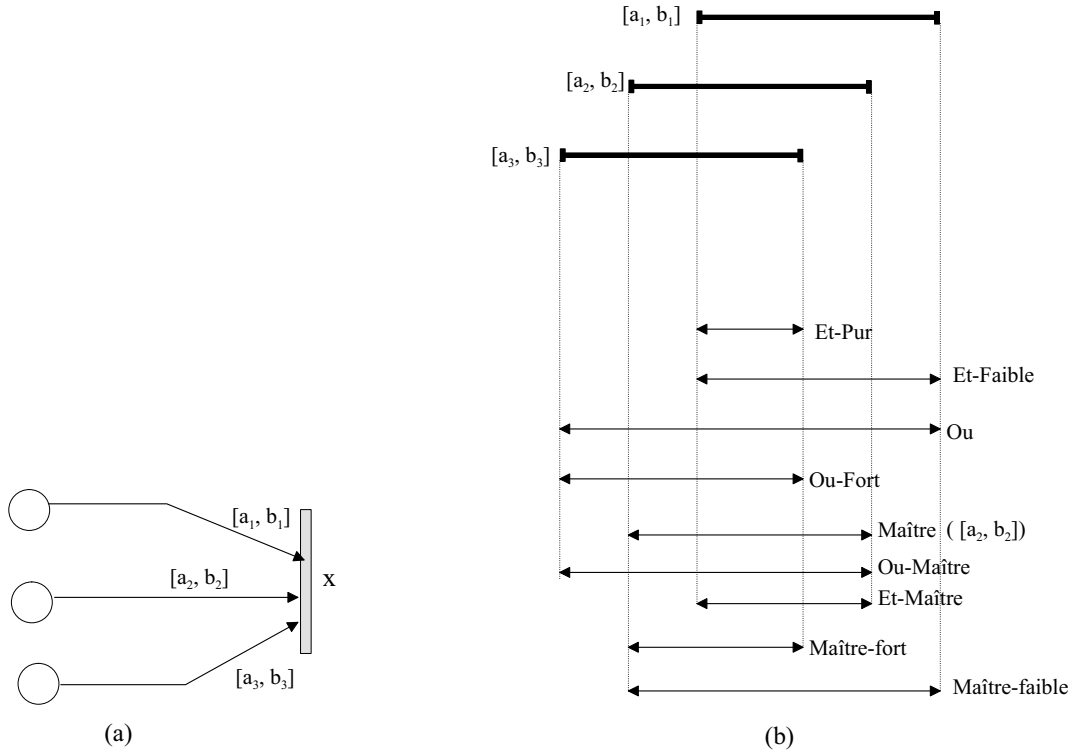


Figure 2.14 : Sémantique de synchronisation de la transition x

2.4.4.2 Mise en équation des graphes d'événements à flux temporels

Notons tout d'abord que la date de tir possible d'une transition dépend des intervalles de validité temporelle de ses arcs incidents, et de la règle de synchronisation affectée à la transition. Nous introduisons maintenant les résultats relatifs à la modélisation du comportement dynamique des graphes d'événements à flux temporels, ainsi que la présentation mathématique qui traduit les différentes règles de tir. Cette présentation se met sous forme d'équations mathématiques de type $(\max, +)$ et $(\min, +)$.

Proposition 2.21. *Soit x_i est une transition de synchronisation dans un graphe d'événements à flux temporel, alors :*

1. si x_i est de type *Et-Pur*, alors :

$$\bigoplus_{j \in s_i} x_j(k - m_j) \otimes a_j \leq x_i(k) \leq \bigwedge_{j \in s_i} x_j(k - m_j) \odot b_j \quad (2.31)$$

2. si x_i est de type *Et-Faible*, alors :

$$\bigoplus_{j \in s_i} x_j(k - m_j) \otimes a_j \leq x_i(k) \leq \bigoplus_{j \in s_i} x_j(k - m_j) \otimes b_j \quad (2.32)$$

3. si x_i est de type *Ou-Fort*, alors :

$$\bigwedge_{j \in s_i} x_j(k - m_j) \odot a_j \leq x_i \leq \bigwedge_{j \in s_i} x_j(k - m_j) \odot b_j \quad (2.33)$$

4. si x_i est de type *Ou*, alors

$$\bigwedge_{j \in s_i} x_j(k - m_j) \odot a_j \leq x_i \leq \bigoplus_{j \in s_i} x_j(k - m_j) \otimes b_j \quad (2.34)$$

avec m_j est le nombre de jetons que contient chaque p_j en aval de x_j ou bien en amont de x_i .
 s_i est le nombre d'arcs entrant en x_i .

Preuve :

1. Il suffit de considérer l'équation donnée dans le point 1 de la définition 2.19, et de l'appliquer sur l'ensemble des places (arcs) d'entrées de x_i . Sachant que τ_j qui est la date d'arrivée à la place p_j (date de sensibilisation de a_j) correspond à $x_j(k - m_j)$.
2. Similaire à la démonstration sur le point 1.
3. D'après la définition 2.19, le flux le plus rapide (arc en avance) impose le franchissement au plus tôt $\bigwedge_{j \in s_i} x_j(k - m_j) \odot a_j$. De même il impose aussi la valeur maximale de l'intervalle de tir, il arrête les autres flux, même s'ils ne sont pas finis. D'où la valeur maximale de l'intervalle de tir qui est donnée par $\bigwedge_{j \in s_i} x_j(k - m_j) \odot b_j$
4. On remarque que cette règle de tir est la plus relaxée d'après la définition 2.19. Tous les intervalles temporels sont acceptables, en tant qu'intervalles de tir. D'où le système 2.34

□

Exemple 2.4 :

La figure 2.15 illustre une modélisation de scénarios multimédias par les graphes d'événements à flux temporels. L'objectif est de synchroniser un flux de voix avec un flux de vidéo pour avoir une perception visuelle en harmonie avec le son de la voix.

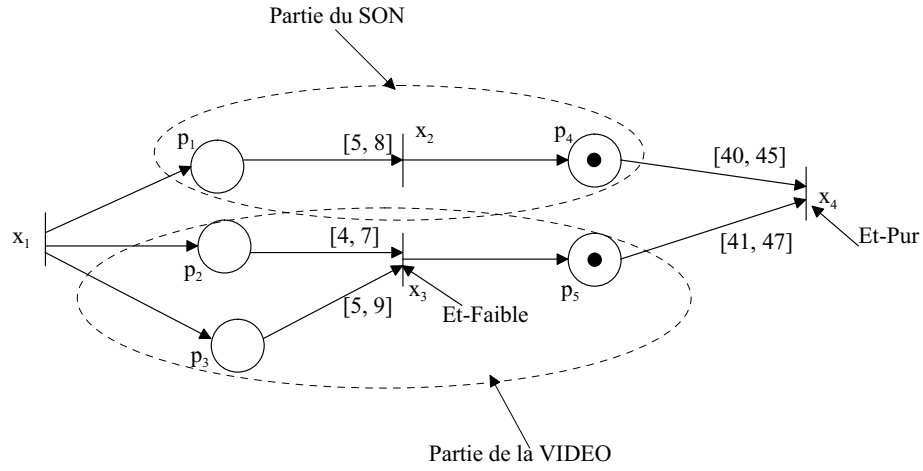


Figure 2.15 : Modélisation d'un système multimédia par un graphe d'événements à flux temporels

La préparation de la bande sonore prends entre 5 et 8 unités de temps, et elle est lancée après l'arrivée d'un jeton à la place p_1 . Pour franchir la transition x_3 qui marque le début de l'émission de la vidéo, il faut synchroniser la tâche de réglage du matériel p_2 avec celle de la préparation de la bande vidéo p_3 (re-bobinage par exemple). On attribue une sémantique Et-faible à x_3 (on peut attendre la fin des deux opérations avant le lancement de la vidéo, la première dure entre 4 et 7 unités de temps, la deuxième entre 5 et 9 unités de temps). Le marquage de p_4 et p_5 fait démarrer la voix et la vidéo, et elles durent respectivement entre 40 et 45 et entre 41 et 47 unités de temps. On a décidé que cette présentation multimédia s'arrête (franchissement de 4) dès que la vidéo ou la voix est finie et c'est pour cela que l'on associe la sémantique Et-pur à la transition x_4 . Nous supposons qu'à l'état initial, la voix et la vidéo sont prêtes à être diffusées, d'où le marquage initial du graphe de la figure 2.15 est $M_0 = (0, 0, 0, 1, 1)^t$. En appliquant les résultats établis dans la proposition 2.21 sur le graphe d'événements à flux temporels de la figure 2.15, nous obtenons :

$$5 \otimes x_1 \leq x_2(k) \leq 8 \odot x_1(k) \quad (2.35)$$

$$(4 \otimes x_1(k)) \oplus (5 \otimes x_1(k)) \leq x_3(k) \leq (7 \otimes x_1(k)) \oplus (9 \otimes x_1(k)) \quad (2.36)$$

$$(40 \otimes x_2(k-1)) \oplus (41 \otimes x_3(k-1)) \leq x_4(k) \leq (45 \odot x_2(k-1)) \wedge (47 \odot x_3(k-1)) \quad (2.37)$$

Remarque : A partir du graphe d'événements à flux temporel de la figure 2.15, nous pouvons remarquer qu'une combinaison des systèmes 2.35, 2.36 et 2.37 peut conduire à un modèle plus complexe. En général, dans un graphe d'événements à flux temporel contenant plusieurs types de sémantiques, on a le modèle d'intervalle suivant :

$$\left\{ f^-(x(k), \dots, x(k-M)) \leq x(k) \leq f^+(x(k), \dots, x(k-M)) \right. \quad (2.38)$$

avec f^- et f^+ deux fonctions du type ($\min, \max, +$).

◇

2.4.4.3 Re-formulation pour les sémantiques Et-Pur et Et-Faible

Nous présentons maintenant une autre forme de modélisation relative aux graphes d'événements à flux temporels. Cette nouvelle formulation concerne les graphes d'événements à flux temporels contenant seulement les deux types de sémantiques de tir : Et-Pur et Et-Faible.

D'une part, on montre tout d'abord que f^- est de type (max, +). La proposition 2.21 indique que pour ces deux types de sémantiques, l'équation suivante est toujours vérifiée : $\bigoplus_{j \in s_i} (x_j(k - m_j) + a_j) \leq x_i(k)$ pour toute transition x_i du graphe d'événements à flux temporels, d'où f^- est de type (max, +). Nous pouvons donc écrire :

$$f^-(x(\gamma), u(\gamma)) = A^- \otimes x(\gamma) \oplus B^- u(\gamma) \quad (2.39)$$

En revanche, pour les même type de sémantiques, la proposition 2.21 montre que l'on peut avoir les deux présentations suivantes : $x_i(k) \leq \bigwedge_{j \in P_i} (x_j(k - m_j) + b_j)$ et $x_i(k) \leq \bigoplus_{j \in P_i} (x_j(k - m_j) + b_j)$. Il en résulte alors que dans un graphe d'événements à flux temporels, toute combinaison des deux dernières formes conduit à une fonction f^+ de type (min, max, +).

D'autre part, l'équation 1.35 montre que toute fonction de type (min, max, +) admet une max-représentation. D'où l'écriture suivante :

$$f^+(x(\gamma), u(\gamma)) = \bigwedge_{i=1}^{j_i} A_i^+ \otimes x(\gamma) \oplus B_i^+ u(\gamma) \quad (2.40)$$

avec j_i est le nombre de termes (max, +).

Les matrices A^- , B^- , A_i^+ et B_i^+ appartiennent à $\overline{\mathbb{R}}_{max}[\gamma]$

Remarque : Les formes 2.39 et 2.40 intègrent l'équation d'état représentant un graphe d'événements temporisé 2.42. En effet, il suffit de reprendre le modèle d'intervalle donné par 2.39 et 2.40 et de considérer que : $j_i = 1$, $A^- = A_1^+ = A\gamma$ et $B^- = B_1^+ = B$. Ce résultat sera énoncé à la fin de ce chapitre sur la classification des graphes d'événements temporisés et temporels. On a donc l'inclusion des graphes d'événements temporisés dans les graphes d'événements à flux temporels.

◇

2.5 Classification des graphes d'événements temporisés et temporels

La capacité du modèle à saisir la réalité d'un système, le pouvoir de modélisation et la capacité de vérifier une propriété d'un système par son modèle constituent des critères importants quand on cherche à comparer les qualités et les défauts d'un modèle [Sén96]. Un point de vue peut être de partir d'un ensemble d'exemples concrets, et de mener une étude critique en

proposant des comparaisons formelles entre plusieurs modèle de RdPs temporels⁹ utilisés pour représenter ces exemples. Tout en se basant sur une analyse de leurs qualités attendues, une étude comparative de différents modèles de RdPs a été proposée dans [Boy01].

Dans ce chapitre, le point de vue est différent car nous nous basons sur les modèles algébriques des différentes sous-classes des RdPs présentées au long de ce chapitre pour réaliser une classification. L'introduction du temps sur les graphes d'événements a généré plusieurs sous-classes, que l'on peut scinder en deux sous-grandes familles selon que le comportement soit borné par dessus et dessous ou uniquement par dessous.

- Les modèles dont l'évolution est bornée par dessous et par dessus, sont modélisés à partir de temporisations définies sur un intervalle.

Si nous reprenons les modèles mathématiques obtenus lors de la modélisation de différents graphes d'événements du type intervalle, alors nous pouvons constater qu'ils ont la forme commune suivante :

$$f^-(x) \leq x \leq f^+(x) \quad (2.41)$$

Une autre forme est $f^-(x) \oplus x = x$ et $x = x \wedge f^+(x)$. Ces modèles sont non déterministes dans leur forme de base.

- Les modèles dont l'évolution est bornée par dessous, sont en général analysés et exploités dans un fonctionnement déterministe en raison d'un choix de fonctionnement au plus tôt local. Ainsi, pour la forme dateur, les graphes d'événements temporisés présentent un modèle du type $f(x) \leq x$ et l'égalité $f(x) = x$ vient de l'hypothèse du fonctionnement au plus tôt : c'est généralement ce dernier modèle avec un fonctionnement déterministe qui est utilisé dans les différents travaux.

Le modèle implicite des graphes d'événements temporisés dans l'algèbre $(\max, +)$ est donnée par : $x(\gamma) \geq A\gamma \otimes x(\gamma) \oplus B \otimes u(\gamma)$. Le dernier modèle correspond a un système dont les spécifications sont données comme suit :

1. La borne inférieure f^- est prise comme fonction $(\max, +)$ avec $f^-(x(\gamma), u(\gamma)) = A\gamma \otimes x(\gamma) \oplus B \otimes u(\gamma)$.
2. La borne supérieure f^+ est infinie.

Sous l'hypothèse d'un fonctionnement au plus tôt local, un graphe d'événements temporisé peut être représenté alors par :

$$f^-(x(\gamma), u(\gamma)) = f^+(x(\gamma), u(\gamma)) = A\gamma \otimes x(\gamma) \oplus B \otimes u(\gamma) \quad (2.42)$$

⁹le terme temporels ici recouvre les RdPs temporels et temporisés

Le tableau de la figure 2.16 donne une classification de tous les modèles vus dans ce chapitre à partir du type des fonctions $f^-()$ et $f^+()$:

		$f^+(x)$ est de type			
		$f^+(x) = +\infty$	(min, +)	(max, +)	(min, max, +)
f(x) est de type	$f^-(x) = -\infty$	pas de contraintes	graphes d'événements p-temporisés : équations aux compteurs		
	(min, +)				
	(max, +)	graphes d'événements p-temporisés : équations aux dateurs	graphes d'événements : - p-temporels - arcs temporels	graphes d'événements t-temporels	
	(min, max, +)				graphes d'événements à flux temporels

Figure 2.16 : Classification des graphes d'événements

A partir de différents graphes d'événements (temporisés et temporels), la figure 2.17 montre les différentes relations existant entre ces modèles. Ainsi, les graphes d'événements temporisés sont un cas particulier des graphes d'événements temporels. De même, les graphes d'événements à flux temporels présentent la forme la plus générale par rapport aux autres graphes d'événements temporels.

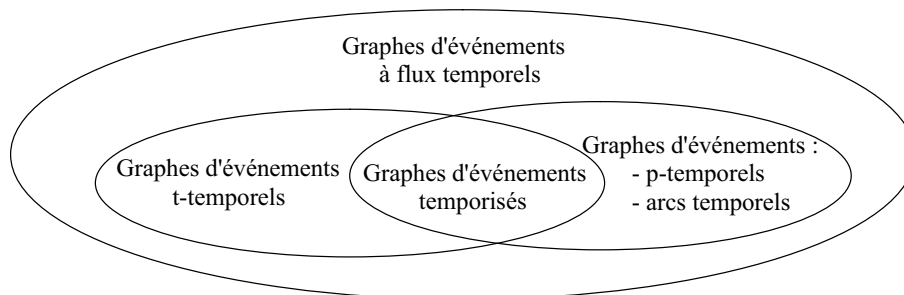


Figure 2.17 : Classification des graphes d'événements

En particulier :

- le modèle algébrique du graphe d'événements t-temporel correspond à la sémantique du Et-Faible du graphe d'événements à flux temporel, les arcs entrant, de chaque transition, ayant les mêmes valeurs d'intervalle.
- le modèle algébrique du graphe d'événements à arcs temporel est identique à celui du graphe d'événements p-temporel et correspond à la sémantique du Et-Pur du graphe d'événements à

flux temporel.

- le modèle algébrique du graphe d'événements p-temporisé en dateurs correspond au terme $(\max, +) f^-(x) \leq x$ des sémantiques Et-Pur ou Et-Faible. En fonctionnement au plus tôt, il correspond au Et-Faible avec $f^-(x) = f^+(x)$.

Symétriquement, le modèle compteur des GET correspond au terme $(\min, +) x \geq f^+(x)$ des sémantiques Et-Pur ou Et-Faible. En fonctionnement au plus grand rang, il correspond au Ou-Fort avec $f^-(x) = f^+(x)$.

En résumé, les sémantiques du Et-Pur et Et-Faible englobent tous les types de graphes d'événements lorsque les dateurs sont considérés.

2.6 Autres modèles : exemples de systèmes (min, max,+)

Comme précédemment, l'objectif est de donner des exemples de modélisation de comportement temporel sous forme de contraintes de types (min, max, +).

2.6.1 Système de production : "Descriptor systems"

Dans cet exemple [Ols91a] et [Ols93], on considère le système de production présenté dans la figure 2.18. Nous commençons tout d'abord par une description détaillée du mode de fonctionnement de ce système.

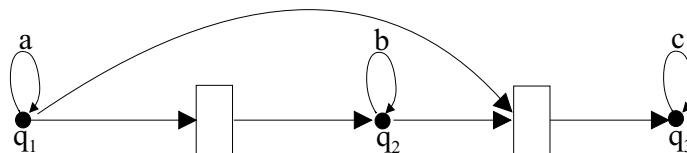


Figure 2.18 : Exemple d'un système manufacturier

On associe à chaque noeud q_1 , q_2 et q_3 respectivement, une boucle, avec une temporisation a , b et c qui indique le temps d'exécution d'une tâche. La livraison de produit aux noeuds q_2 et q_3 par q_1 se fait d'une manière simultanée. Les deux noeuds délivrent des produits à q_3 , et c'est le premier produit arrivant, qui sera traité en premier par q_3 . Le fonctionnement du système est dit FCFS¹⁰ Pendant que q_2 traite une activité, il peut recevoir au plus, un produit provenant de q_1 , qui peut être conservé dans un stock représenté par un rectangle dans la figure 2.18. Si ce stock est plein, le noeud q_1 doit attendre qu'il soit vide, avant de lui livrer un produit. Le noeud

¹⁰FCFS: First Coming First Served que l'on peut traduire en français comme le premier arrivé est le premier servi

q_3 a aussi un stock d'entrée qui peut contenir deux produits au maximum. Par conséquent, pour amorcer le traitement d'une activité, chaque noeud doit livrer son (ses) produit(s) à son (ses) noeud(s) destinataire(s).

Pour décrire mathématiquement le système étudié, on fait correspondre à chaque noeud q_i la quantité $x_i(k)$ qui représente la date de franchissement de la transition x_i (date de livraison d'un produit par un noeud q_i). Nous obtenons le modèle suivant:

$$\begin{cases} x_1(k+1) = a \otimes x_1(k) \oplus x_2(k) \oplus x_3(k) \\ x_2(k+1) = a \otimes x_1(k) \oplus b \otimes x_2(k) \oplus x_3(k) \\ x_3(k+1) = (a \otimes x_1(k) \wedge b \otimes x_2(k)) \oplus (c \otimes x_3(k)) \end{cases} \quad (2.43)$$

Afin que le dernier produit arrivant à l'entrée de q_3 puisse être traité, on introduit une transition fictive x_4 dans le réseau de la figure 2.18. En réalité, la transition x_4 est la même que x_3 , et son principal rôle est de s'occuper du dernier produit livré par q_1 et q_2 . Le système 2.43 devient :

$$\begin{cases} x_1(k+1) = a \otimes x_1(k) \oplus x_2(k) \oplus x_4(k) \\ x_2(k+1) = a \otimes x_1(k) \oplus b \otimes x_2(k) \oplus x_4(k) \\ x_3(k+1) = (a \otimes x_1(k) \wedge b \otimes x_2(k)) \oplus (c \otimes x_4(k)) \\ x_4(k+1) = x_4(k+1) = \bigoplus[(a \otimes x_1(k) \oplus b \otimes x_2(k) \oplus 2c \otimes x_3(k+1)), (c \otimes a \otimes x_1(k) \\ \wedge c \otimes b \otimes x_2(k))] \end{cases} \quad (2.44)$$

Le fonctionnement dynamique du système de la figure 2.18 est donné par 2.44. Nous constatons que ce dernier modèle comporte à la fois les trois opérations qui sont la maximisation, la minimisation et l'addition. Le modèle 2.44 se présente donc sous forme d'un système de type $(\min, \max, +)$. Le système 2.44 correspond au modèle suivant $f^-(x) \leq x \leq f^+(x)$ avec $f^-(x) = f^+(x) = f(x)$. La fonction $f(x)$ est de type $(\min, \max, +)$.

2.6.2 Système logique

Pour la conception de cet exemple, on s'est inspiré des résultats des travaux de [CDY02] et [Tha04].

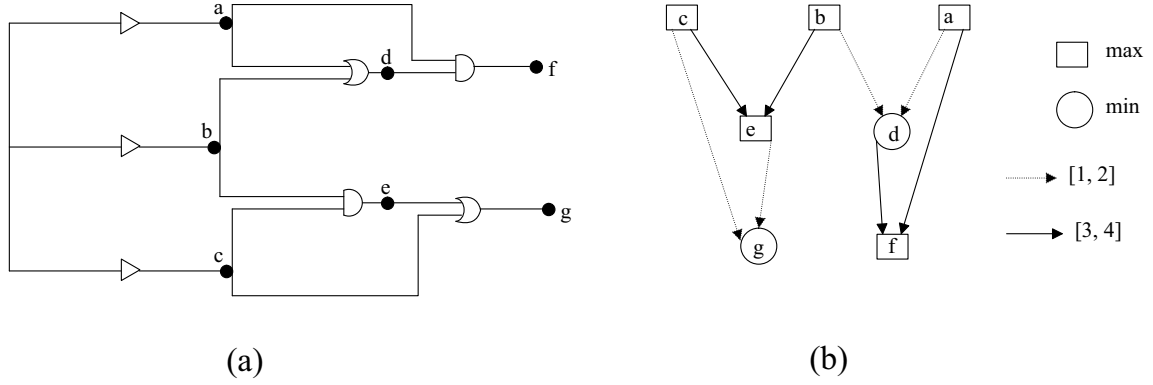


Figure 2.19 : Système à circuits logiques et son graphe de contraintes temporelles

La figure 2.19a donne l'exemple d'un système électronique composé de circuits logiques interconnectés. Nous traitons le cas d'un circuit constitué par l'interconnexion de portes logiques *ET* et *OU*. L'instant de sortie d'un signal d'une porte logique *ET*, est le maximum entre les instants d'occurrences des signaux d'entrées. En revanche, pour une porte logique *OU*, c'est le minimum entre les instants d'occurrences des signaux d'entrées. Pour chaque type de portes logiques, on associe un intervalle qui caractérise une délimitation des instants, au plus tôt et au plus tard d'obtention du signal de sortie. La figure 2.19 montre le graphe de contraintes temporelles associé au système de la figure 2.19a. Nous obtenons les inéquations suivantes :

$$(a + 1) \wedge (b + 1) \leq d \leq (a + 2) \wedge (b + 2) \quad (2.45)$$

$$(b + 3) \oplus (c + 3) \leq e \leq (b + 4) \oplus (c + 4) \quad (2.46)$$

$$(e + 1) \wedge (c + 1) \leq g \leq (e + 2) \wedge (c + 2) \quad (2.47)$$

$$(a + 3) \oplus (d + 3) \leq f \leq (a + 4) \oplus (d + 4) \quad (2.48)$$

On retrouve ainsi une forme analogue aux sémantiques du Ou-Fort et du Et-faible des graphes d'événements à flux temporels.

En injectant les équations 2.45 et 2.46 dans 2.47 et 2.48, nous arrivons à exprimer les contraintes liées à f et g en fonctions de a , b et c comme suit :

$$(a + 3) \oplus ((a + 1 + 3) \wedge (b + 1 + 3)) \leq f \leq (a + 4) \oplus ((a + 2 + 4) \wedge (b + 2 + 4)) \quad (2.49)$$

$$(c + 1) \wedge ((b + 3 + 1) \oplus (c + 3 + 1)) \leq g \leq (c + 2) \wedge ((b + 4 + 2) \oplus (c + 4 + 2)) \quad (2.50)$$

Nous aboutissons aux équations 2.45, 2.46, 2.49 et 2.50 qui caractérisent le comportement temporel du système modélisé. Elles traduisent les contraintes temporelles liées à chacune des portes logiques de la figure 2.19.

Nous concluons que le système mis sous forme d'équations de type $(\min, \max, +)$ est déjà très complexe bien que le système physique logique soit très simple et ne comporte qu'un nombre réduit de composants.

2.7 Conclusion

Ce chapitre a eu pour objectif de montrer, à travers des exemples aussi concrets et variés que possible, que la modélisation d'un certain nombre de problèmes peut être décrite par des équations ou inéquations de type $(\max, +)$, $(\min, +)$ et $(\min, \max, +)$. A notre connaissance, c'est la première fois que le lien entre les graphes d'événements temporels, et l'algèbre des dioïdes a été établi.

Certains modèles sont particulièrement compliqués, comme les systèmes logiques qui n'ont pourtant pas une réputation de complexité. On trouve cette complexité dans les graphes d'événements à flux temporels et en particulier, pour certaines sémantiques comme le OU ou le Maître. Dans ce chapitre, une classification des modèles a été proposée d'un point de vue algébrique, afin de faciliter toute future recherche dans ce domaine. Cette classification se base sur le type des fonctions obtenues pour chacune de ces sous-classes. Le graphe d'événements à flux temporel représente ainsi le modèle algébrique dont la forme est la plus générale. On pourra cependant se limiter en dateurs, aux sémantiques du Et-Faible et du Et-Pur, car celles-ci intègrent tous les types temporels des différents graphes d'événements considérés.

La description du mode de fonctionnement au plus tard a été introduite. Ce modèle est spécifique à ces graphes d'événements temporels où l'on a des contraintes sur la durée de séjour maximale des jetons dans les places. D'autre part nous avons montré que la notion du fonctionnement au plus tôt des graphes d'événements p-temporels ne peut être définie de manière analogue aux GET. En d'autres termes, nous avons montré que le fonctionnement au plus tôt (et au plus tard) pour les graphes d'événements p-temporels ne peut être prise localement (entre une place et une transition) mais globalement, sous peine de blocage du système. On ne peut donc pas transposer les notions fondamentales d'équations d'état et de fonction de transfert des GET sans une certaine adaptation, si celle-ci est possible.

Un travail d'analyse et d'exploitation de différents modèles obtenus sera l'objet des chapitres suivants.

Chapitre 3

De la validation des modèles à la vivacité des graphes d'événements p-temporels

3.1 Position du problème et contexte de l'étude

Dans ce chapitre, le premier point sera consacré à l'étude de la validation des modèles d'intervalles. Cette étude consistera à analyser la compatibilité dans plusieurs cas de systèmes de type intervalle. On établira des conditions permettant de vérifier respectivement la compatibilité des systèmes de types : $((\max, +), (\min, \max, +))$, $((\min, \max, +), (\min, +))$ et $((\max, +), (\min, +))$. Le système $((\max, +), (\min, +))$ sera considéré comme un cas particulier des deux autres systèmes. Les résultats qui seront énoncés dans l'étude de la compatibilité des systèmes $((\max, +), (\min, \max, +))$, $((\min, \max, +), (\min, +))$ seront alors adaptés et particularisés à l'étude de la compatibilité le système de type $((\max, +), (\min, +))$. Comme nous l'avons montré au chapitre 2, chacun de ces systèmes peut correspondre à un modèle (graphe d'événements) précis. En particulier, le modèle d'intervalle de type $((\max, +), (\min, +))$ correspond bien aux graphes d'événements p-temporels.

Nous exposerons ensuite un rappel sur les méthodes existantes analysant la vivacité dans les graphes d'événements temporels (en particulier p-temporels).

Enfin, nous proposerons une première approche qui utilise les résultats énoncés dans le premier point afin d'analyser la vivacité des graphes d'événements p-temporels : nous allons exploiter et appliquer alors les résultats établis dans l'étude de la compatibilité du système $((\max, +), (\min, +))$ pour traiter le problème de la vivacité. Cette approche est basée sur le calcul du vecteur spectral. Une deuxième approche spécifique aux graphes d'événements p-

temporels sera également proposée. Cette dernière approche est basée sur les séries de matrices qui seront introduites dans ce chapitre. Nous concluons ce chapitre par une illustration des deux approches.

Ce chapitre rassemble les contributions de [DAD04] et [DDA04] et quelques résultats présentés dans [DDA05a].

3.2 Validation des modèles

3.2.1 Modèle d'intervalles

Il a été montré dans le chapitre 2 que le modèle correspondant aux différentes classes de graphes d'événements temporisés et temporels peut sous mettre sous la forme initiale suivante :

$$\begin{cases} x(\gamma) = x(\gamma) \wedge f^+(x(\gamma), \dots, \gamma^m x(\gamma), u(\gamma), \dots, \gamma^m u(\gamma)) \\ x(\gamma) = x(\gamma) \oplus f^-(x(\gamma), \dots, \gamma^m x(\gamma), u(\gamma), \dots, \gamma^m u(\gamma)) \end{cases} \quad (3.1)$$

où f^+ et f^- sont deux fonctions appartenant à l'ensemble des fonctions (min, max, +) et m est le nombre de jetons présents dans chaque place.

Les composantes du vecteur u correspondent aux transitions d'entrée. Pour simplifier l'écriture, nous supposons que le vecteur x comprend aussi le vecteur y des transitions de sortie. Les fonctions $f^+(\cdot)$ et $f^-(\cdot)$ représentent respectivement les bornes supérieure et inférieure à x dont la trajectoire évolue entre ces deux bornes. En particulier, si la borne inférieure $f^-(\cdot)$ est de type (max, +) et la borne supérieure tend vers l'infini ($f^+(\cdot) = +\infty$), alors, le système classique (max, +) linéaire qui modélise les graphes d'événements temporisés, est retrouvé.

Le type du système 3.1 est défini par le type des fonctions $f^-(\cdot)$ et $f^+(\cdot)$. Le modèle 3.2 sera décrit par le couple suivant : (type de $f^-(\cdot)$, type de $f^+(\cdot)$). Le type ((min, max, +), (min, max, +)) représente le cas le plus général pour le système 3.1.

Nous introduisons ci-dessous la définition concernant la notion de compatibilité. Cette dernière, généralise la la notion d'admissibilité introduite dans la définition 2.14. La théorie que nous développerons par la suite utilisera cette notion importante.

Définition 3.1. *Soit un modèle d'intervalles défini par le couple (type de $f^-(\cdot)$, type de $f^+(\cdot)$). On dit que ce système est compatible sur un horizon fini ou infini, s'il existe une trajectoire d'état x et une trajectoire d'entrée u vérifiant les équations de ce modèle d'intervalle sur cet horizon.*

Nous introduisons maintenant une définition relative au calcul du vecteur spectral qui sera utilisée par la suite.

Définition 3.2. Soient $x \leq f(x)$ et le vecteur $X_h = (x(k), x(k+1), \dots, x(k+h))^t$ développé sur l'horizon h . La fonction algébrique $\Phi_f(X_h)$ est obtenue en développant $x(\gamma) \leq f(x(\gamma))$ algébriquement sur X_h . Cette fonction permet de décrire toutes les relations qui existent entre les composantes du vecteur X_h .

Les théorèmes donnés antérieurement 1.56, 1.57 et 1.58 sont applicables dans les cas où les systèmes ont respectivement les formes $x \leq f(x)$, $x \geq f(x)$ et $x = f(x)$ avec $f \in D^*$. Cela nous amène à considérer des systèmes qui représentent des cas particulier du modèle (type de $f^-(\cdot)$, type de $f^+(\cdot)$). Nous allons donc analyser la compatibilité dans un premier temps des systèmes de type $((\max, +), (\min, \max, +))$. L'approche pour les systèmes de type $((\min, \max, +), (\min))$ sera duale. Elle sera donnée afin de formaliser clairement les équations. Enfin, l'analyse de la compatibilité des systèmes de type $((\max, +), (\min, +))$ est un cas particulier important qui correspond au type des fonctions du modèle des graphes d'événements p-temporels.

3.2.2 Compatibilité dans les systèmes de type $((\max, +), (\min, \max, +))$

Une présentation réduite du système 3.1 par des séries formelle en γ est la suivante :

$$\begin{cases} x(\gamma) \leq f^+(z(\gamma)) \\ x(\gamma) \geq f^-(z(\gamma)) \end{cases} \quad (3.2)$$

$$\text{où } z(\gamma) = \begin{pmatrix} x(\gamma) \\ u(\gamma) \end{pmatrix}$$

Nous considérons ici le cas où la fonction f^- est du type $(\max, +)$ et f^+ du type $(\min, \max, +)$. La proposition suivante donne les conditions nécessaires et suffisantes pour que ce type de système soit compatible sur un horizon quelconque.

Proposition 3.3. Soit le modèle d'intervalle donné par 3.2 avec $f^-(z(\gamma))$ de type $(\max, +)$ seulement. Ce système est compatible si, et seulement si, le vecteur spectral de la fonction $G^+(z(\gamma))$ est supérieur ou égal à zéro.

$$G^+(z(\gamma)) = \begin{pmatrix} f^+(z(\gamma)) \wedge (-F_1^-(\gamma))^t \odot x(\gamma) \\ (-F_2^-(\gamma))^t \odot x(\gamma) \end{pmatrix}$$

$$\text{avec } F_1^-(\gamma) = \bigoplus_{i=0}^m A_i^- \otimes \gamma^i \text{ et } F_2^-(\gamma) = \bigoplus_{i=0}^m B_i^- \otimes \gamma^i$$

Preuve : D'une part on a supposé que la fonction $f^-(z(\gamma))$ est de type $(\max, +)$, d'où l'équation $x(\gamma) \geq \bigoplus_{i=0}^m A_i^- \otimes \gamma^i x(\gamma) \oplus \bigoplus_{i=0}^m B_i^- \otimes \gamma^i u(\gamma)$. Cette dernière inéquation peut se réécrire sous la forme :

$$x(\gamma) \geq F_1^-(\gamma)x(\gamma) \oplus F_2^-(\gamma)u(\gamma)$$

ce qui nous permettra d'obtenir les deux inéquations suivantes : $F_1^-(\gamma)x(\gamma) \leq x(\gamma)$ et $F_2^-(\gamma)u(\gamma) \leq x(\gamma)$ que nous réécrivons sous les formes : $f_1(x) \leq x$ et $f_2(x) \leq x$.

D'autre part, les deux fonctions f_1 et f_2 sont isotones et S.C.I. ($f_1(\oplus x) = \bigoplus_{i=0}^m A_i^- \otimes \gamma^i(\oplus x) = \bigoplus_{i=0}^m A_i^- \otimes \gamma^i x = \oplus f_1(x)$). Il en est de même pour f_2) et $f_1(\varepsilon) = \varepsilon$ et $f_2(\varepsilon) = \varepsilon$. Alors, d'après le théorème 1.28, f_1 et f_2 sont résiduables. D'où nous aboutissons au système d'inéquations suivant :

$$\begin{cases} x(\gamma) \leq f^+(z(\gamma)) \wedge F_1^-(\gamma) \setminus x(\gamma) \\ u(\gamma) \leq F_2^-(\gamma) \setminus x(\gamma) \end{cases}$$

ce qui conduit à la représentation équivalente suivante :

$$\begin{cases} x(\gamma) \leq f^+(z(\gamma)) \wedge (-F_1^-(\gamma))^t \odot x(\gamma) \\ u(\gamma) \leq (-F_2^-(\gamma))^t \odot x(\gamma) \end{cases} \quad (3.3)$$

Finalement, le système 3.3 peut se mettre sous la forme réduite suivante : $z(\gamma) \leq G^+(z(\gamma))$. Le système de type $((\max, +), (\min, \max, +))$ est réduit au système de type $(-\infty, (\min, \max, +))$, et pour l'analyser, il suffit donc d'appliquer le théorème 1.56. Si le vecteur spectral de la fonction $G^+(z(\gamma))$ est positif, alors le système $((\max, +), (\min, \max, +))$ est compatible. \square

Proposition 3.4. *Le système de type $((\max, +), (\min, \max, +))$ est compatible si et seulement si, le vecteur spectral correspondant à la fonction G^{++} est égal à zéro.*

$$G^{++}(z(\gamma)) = G^+(z(\gamma)) \wedge z(\gamma)$$

avec G^+ est défini dans la proposition 3.3.

Preuve :

La démonstration est immédiate. Il suffit de reprendre l'équation $z(\gamma) \leq G^+(z(\gamma))$, il en résulte alors que $z(\gamma) = z(\gamma) \wedge G^+(z(\gamma))$ ce qui conduit à la nouvelle forme $z(\gamma) = G^{++}(z(\gamma)) = z(\gamma) \wedge G^+(z(\gamma))$. Cela nécessite l'application du théorème 1.58 pour vérifier l'existence d'une solution finie. \square

Jusqu'à maintenant, nous avons traité le problème de compatibilité des systèmes $((\max, +), (\min, \max, +))$ sans nous occuper de l'horizon d'analyse. En effet, les résultats que nous avons introduits et qui sont énoncés par les deux proposition 3.3 et 3.4 sont valables en général, quel que soit l'horizon du travail considéré. Or, dans le cas pratique, la dimension de l'horizon est une notion primordiale, et c'est ce que nous allons illustrer dans la partie réservée à l'analyse de la vivacité. C'est pour cela que nous allons étudier par la suite la compatibilité des systèmes de type $((\max, +), (\min, \max, +))$ avec la prise en compte de l'horizon d'analyse.

Remarque : Auparavant, les équations d'état considérés sont représentées par des séries formelle en γ . Ce type de présentation nous a permis de manipuler des formes plus concises et réduites. En revanche,

dans la suite de cette section, nous aurons besoin des formes d'état plus précises tenant compte l'horizon. Ceci, nous amènera à utiliser des équations d'état en dateurs, paramétrées en k .

◇

Nous considérons les deux formes suivantes et qui peuvent se déduire à partir du système 3.2 ou bien 3.1 (en prenant $m = 1$) :

$$x(k) \leq f^+(x(k), x(k-1), u(k)) \quad (3.4)$$

$$x(k) \geq f^-(x(k), x(k-1), u(k)) \quad (3.5)$$

Les deux formes 3.4 et 3.5 peuvent représenter une modélisation d'un graphe d'événements (temporisé ou temporel) dans lequel les places existant entre une transition d'entrée et interne, ne contiennent pas de jetons. En revanche, les autres places contiennent chacune un jeton au plus.

La proposition suivante analyse la compatibilité des modèles d'intervalles de type $((\max, +), (\min, \max, +))$ sur un horizon déterminé h .

Proposition 3.5. *Soit le modèle d'intervalle du type $((\max, +), (\min, \max, +))$ avec*

$$f_j^-(x(k+j), x(k+j-1), u(k+j)) = \bigoplus_{i=0}^1 A_i^- \otimes x(k+j-i) \oplus B^- \otimes u(k+j) \quad (3.6)$$

Ce système est compatible sur l'horizon h si et seulement si, le vecteur spectral de la fonction $g_h^+(z_h(k))$ est supérieur ou égal à zéro, c'est à dire $\chi(\Phi_{g_h^+}(z_h)) \geq 0$.

$$g_h^+(z_h(k)) = F_h^+(z_h(k)) \wedge M^t \setminus z_h(k) \wedge z_h(k+1) \quad (3.7)$$

$$\text{avec } z_h(k) = \begin{pmatrix} x(k) \\ \vdots \\ x(k+h-1) \\ x(k+h) \\ u(k) \\ \vdots \\ u(k+h-1) \\ u(k+h) \end{pmatrix}, \quad M = \left(\begin{array}{c|c} M_{11} & M_{12} \\ \hline M_{21} & M_{22} \end{array} \right),$$

$$M_{11} = \begin{pmatrix} A_0^- & A_1^- & \cdots & T \\ T & \ddots & \ddots & \vdots \\ \vdots & & A_0^- & A_1^- \\ T & & \cdots & A_0^- \end{pmatrix} \quad \text{et } M_{21} = \begin{pmatrix} B^- & T & \cdots & T \\ T & \ddots & & \vdots \\ \vdots & & B^- & \\ T & & \cdots & B^- \end{pmatrix}$$

$F_h^+(z_h(k))$ est une matrice colonne constituée de deux blocs. Les coefficients du premier bloc

supérieur sont les $h+1$ fonctions f_j^+ telles que, pour chaque élément j de $z_h(k)$, on a la relation $(z_h(k))_j \leq f_j^+(x(k+j), x(k+j-1), u(k+j))$. Les éléments du deuxième bloc sont égaux à T . Les coefficients des deux matrices M_{12} et M_{22} sont tous égaux l'infini, c'est à dire $M_{12} = M_{22} = T$.

Preuve :

$$\text{Pour } 0 \leq j \leq h \text{ nous avons: } \begin{cases} x(k+j) \leq f_j^+(x(k+j), x(k+j-1), u(k+j)) \\ x(k+j-1) \leq x(k+j) \\ u(k+j-1) \leq u(k+j) \end{cases}$$

Aussi, on a :

$A_0^- \otimes x(k+j) \oplus A_1^- \otimes x(k+j-1) \oplus B^- \otimes u(k+j) \leq x(k+j)$. En appliquant le théorème 1.32, nous arrivons au système :

$$\iff \begin{cases} x(k+j) \leq A_0^- \setminus x(k+j) \\ x(k+j-1) \leq A_1^- \setminus x(k+j) \\ u(k+j) \leq B^- \setminus x(k+j) \end{cases}$$

La concaténation des deux derniers systèmes, donne la forme suivante : $\forall 0 \leq j \leq h$

$$\begin{cases} x(k+j) \leq f_j^+(x(k+j), x(k+j-1), u(k+j)) \\ \wedge A_0^- \setminus x(k+j) \wedge A_1^- \setminus x(k+j+1) \wedge x(k+j+1) \\ u(k+j) \leq B^- \setminus x(k+j) \wedge u(k+j+1) \end{cases} \quad (3.8)$$

et en particulier, pour $j = h$, on a :

$$\begin{cases} x(k+h) \leq f_h^+(x(k+h), x(k+h-1), u(k+h)) \wedge A_0^- \setminus x(k+h) \\ u(k+h) \leq B^- \setminus x(k+h) \end{cases}$$

Finalement, nous aboutissons à la forme matricielle :

$$\begin{aligned} z_h(k) &\leq F_h^+(z_h(k)) \wedge M^t \setminus z_h(k) \wedge z_h(k+1) \\ \implies z_h(k) &\leq g_h^+(z_h(k)) \end{aligned}$$

Dire que l'inéquation $z_h(k) \leq g_h^+(z_h(k))$ admet une solution finie, est équivalent à vérifier que le vecteur spectral de la fonction $g_h^+(z_h(k))$ est positif ou nul. L'étude de la compatibilité du système de type $((\max, +), (\min, \max, +))$ sur l'horizon h est réduite à l'étude d'un système de type $(-\infty, (\min, \max, +))$ sur le même horizon. □

Afin d'informatiser les équations précédentes sans faire appel à la résiduation, nous pouvons décomposer le système d'inéquations 3.8 pour obtenir une forme plus développée. Notons tout d'abord que dans l'algèbre $(\max, +)$, le signe moins habituel peut être remplacé par la relation suivante: $[X_{ij}]^{\otimes(-1)} = -[X_{ij}]$.

Pour chaque élément $(A_i^-)_{lm} \neq \varepsilon$, nous pouvons écrire $(A_i^-)_{lm} \otimes x_m(k+j-i) \leq x_l(k+j)$ et

par conséquent:

$$x_m(k+j-i) \leq (A_i^-)_{lm}^{(\otimes)-1} \otimes x_l(k+j)$$

Aussi, pour chaque $(B^-)_{lm} \neq \varepsilon$, nous aurons $(B^-)_{lm} \otimes u_m(k+j) \leq x_l(k+j)$

$$\implies u_m(k+j) \leq (B^-)_{lm}^{(\otimes)-1} \otimes x_l(k+j)$$

Pour chaque composante i des vecteurs x et u , on a :

$$\left\{ \begin{array}{l} x_i(k+j) \leq (f_j^+(\cdot))_i \wedge \bigwedge_{(A_0^-)_{li} \neq \varepsilon} (A_0^-)_{li}^{(\otimes)-1} \otimes x_l(k+j) \wedge \\ \bigwedge_{(A_1^-)_{li} \neq \varepsilon} (A_1^-)_{li}^{(\otimes)-1} \otimes x_l(k+j+1) \wedge x_i(k+j+1) \\ u_i(k+j) \leq \bigwedge_{(B^-)_{li} \neq \varepsilon} (B^-)_{li}^{(\otimes)-1} \otimes x_l(k+j) \end{array} \right. \quad (3.9)$$

3.2.3 Compatibilité dans les systèmes de type $((\min, \max, +), (\min, +))$

L'étude de la compatibilité des systèmes de type $((\min, \max, +), (\min, +))$ est dual par rapport aux systèmes $((\max, +), (\min, \max, +))$. Dans cette partie, nous allons énoncer des résultats sans démonstrations car ces dernières sont duales aux preuves déjà présentées dans le cadre de l'étude des systèmes $((\max, +), (\min, \max, +))$. Toutefois, des explications seront données à chaque fois qu'il nous le paraîtra nécessaire afin de faciliter la compréhension de ces résultats.

Proposition 3.6. *Soit le modèle d'intervalle donné par 3.2 avec $f^+(z(\gamma))$ est de type $(\min, +)$ seulement. Ce système est compatible si et seulement si, le vecteur spectral de la fonction $G^-(z(\gamma))$ est inférieur ou égal à zéro.*

$$G^-(z(\gamma)) = \begin{pmatrix} f^-(z(\gamma)) \oplus (-F_1^+(\gamma))^t \otimes x(\gamma) \\ (-F_2^+(\gamma))^t \otimes x(\gamma) \end{pmatrix}$$

$$\text{avec } F_1^+(\gamma) = \bigwedge_{i=0}^m A_i^+ \odot \gamma^i \text{ et } F_2^+(\gamma) = \bigwedge_{i=0}^m B_i^+ \odot \gamma^i$$

Preuve : La fonction $f^+(z(\gamma))$ est de type $(\min, +)$, d'où l'équation $x(\gamma) \leq \bigwedge_{i=0}^m A_i^+ \odot \gamma^i x(\gamma) \wedge \bigwedge_{i=0}^m B_i^+ \odot \gamma^i u(\gamma)$. On obtient les deux inéquations : $F_1^+(\gamma)x(\gamma) \geq x(\gamma)$ et $F_2^+(\gamma)u(\gamma) \geq x(\gamma)$. Les deux dernières inéquations peuvent se mettre sous la forme : $f_1(x) \geq x$ et $f_2(x) \geq x$. D'après le théorème 1.29, les fonctions f_1 et f_2 sont dualement résiduables puisqu'elles sont à la fois isotones et S.C.S. ($f_1(\wedge x) = \bigwedge_{i=0}^m A_i^+ \odot \gamma^i (\wedge x(\gamma)) = \wedge (\bigwedge_{i=0}^m A_i^+ \odot \gamma^i x(\gamma)) = \wedge f_1(x)$). De la même manière, on peut montrer que f_1 est S.C.S.), isotone et $f_1(T) = T$ et $f_2(T) = T$. Le reste de la démonstration est le dual de la preuve du théorème 3.3.

Le système de type $((\min, \max, +), (\min, +))$ est réduit au système de type $((\min, \max, +), +\infty)$, et pour l'analyser, il suffit donc d'appliquer le théorème 1.57.

□

Proposition 3.7. *Le système de type $((\min, \max, +), (\min, +))$ est compatible si et seulement si, le vecteur spectral correspondant à la fonction G^{--} est égal à zéro.*

$$G^{--}(z(\gamma)) = G^-(z(\gamma)) \oplus z(\gamma)$$

avec G^- est défini dans la proposition 3.6.

Proposition 3.8. *Soit le modèle d'intervalles du type $((\min, \max, +), (\min, +))$ avec*

$$f_j^+(x(k+j), x(k+j-1), u(k+j)) = \bigwedge_{i=0}^1 A_i^+ \odot x(k+j-i) \wedge B^+ \odot u(k+j) \quad (3.10)$$

Ce système est compatible sur l'horizon h si et seulement si, le vecteur spectral de la fonction $g_h^-(z_h(k))$ est inférieur ou égal à zéro, c'est à dire $\chi(\Phi_{g_h^-}(z_h)) \leq 0$.

$$g_h^-(z_h(k)) = F_h^-(z_h(k)) \oplus N^t \setminus' z_h(k) \oplus z_h(k-1) \quad (3.11)$$

$$\text{avec } N = \left(\begin{array}{c|c} N_{11} & N_{12} \\ \hline N_{21} & N_{22} \end{array} \right),$$

$$N_{11} = \begin{pmatrix} A_0^+ & A_1^+ & \cdots & \varepsilon \\ \varepsilon & \ddots & \ddots & \vdots \\ \vdots & & A_0^+ & A_1^+ \\ \varepsilon & & \cdots & A_0^+ \end{pmatrix}, \quad N_{21} = \begin{pmatrix} B^+ & \varepsilon & \cdots & \varepsilon \\ \varepsilon & \ddots & & \vdots \\ \vdots & & B^+ & \\ \varepsilon & & \cdots & B^+ \end{pmatrix}$$

et $N_{12} = N_{22} = \varepsilon$ (N_{12} et N_{22} sont deux matrices nulles). $F_h^-(z_h(k))$ est un vecteur dont les premiers $h+1$ éléments sont les fonctions f_j^- telles que $(z_h(k))_j \geq f_j^-(x(k+j), x(k+j-1), u(k+j))$ pour tout $0 \leq j \leq h$. Le reste des coefficients sont égaux à ε .

Preuve : Ici, la forme matricielle que nous cherchons est donnée comme suit :

$$\begin{aligned} z_h(k) &\geq F_h^-(z_h(k)) \oplus N^t \setminus' z_h(k) \oplus z_h(k-1) \\ &\implies z_h(k) \geq g_h^-(z_h(k)) \end{aligned}$$

L'étude de la compatibilité du système de type $((\max, +), (\min, \max, +))$ sur l'horizon h est réduite à l'étude d'un système de type $((\min, \max, +), +\infty)$ sur le même horizon. □

Comme dans les systèmes $((\max, +), (\min, \max, +))$, on associe à chacune des composantes des vecteurs x et u , une contrainte par dessous. La forme développée correspondante aux inéquations équivalentes à 3.8 est donnée par le système suivant :

$$\begin{cases} x_i(k+j) \geq (f_j^-(\cdot))_i \oplus \bigoplus_{(A_0^+)_{li} \neq T} (A_0^+)_{li}^{(\otimes)-1} \otimes x_l(k+j) \oplus \\ \bigoplus_{(A_1^+)_{li} \neq T} (A_1^+)_{li}^{(\otimes)-1} \otimes x_l(k+j) \oplus x_i(k+j-1) \\ u_i(k+j) \geq \bigoplus_{(B^+)_{li} \neq T} (B^+)_{li}^{(\otimes)-1} \otimes x_l(k+j) \end{cases} \quad (3.12)$$

3.2.4 Compatibilité dans les systèmes de type $((\max, +), (\min, +))$

Nous considérons maintenant le système (f^-, f^+) de type $((\max, +), (\min, +))$ avec :

$$\begin{cases} f^-(z(\gamma)) = \bigoplus_{i=0}^m A_i^- \otimes \gamma^i x(\gamma) \oplus \bigoplus_{i=0}^m B_i^- \otimes \gamma^i u(\gamma) \\ f^+(z(k)) = \bigwedge_{i=0}^m A_i^+ \odot \gamma^i x(\gamma) \wedge \bigwedge_{i=0}^m B_i^+ \odot \gamma^i u(\gamma) \end{cases} \quad (3.13)$$

Ce dernier modèle est considéré comme un cas particulier des systèmes $((\max, +), (\min, \max, +))$ et $((\min, \max, +), (\min, +))$.

Proposition 3.9. *Le système 3.13 est compatible si et seulement si, le vecteur spectral de la fonction $G^+(z(\gamma))$ (respectivement $G^-(z(\gamma))$) est supérieur (respectivement inférieur) ou égal à zéro.*

$G^+(z(\gamma))$ et $G^-(z(\gamma))$ sont définies respectivement dans les propositions 3.3 et 3.6.

Preuve : La démonstration est immédiate. Elle est analogue à celle de la proposition 3.3 (respectivement 3.6). □

Corollaire 3.10. *Si le système 3.13 est compatible, alors sont équivalents :*

(i) le vecteur spectral de $G^+(z(\gamma))$ est positif ou nul,

(ii) le vecteur spectral de $G^-(z(\gamma))$ est négatif ou nul.

Preuve :

(i) \implies (ii)

Supposons que le vecteur spectral de la fonction G^+ soit positif, alors le système de type $((\max, +), (\min, +))$ (3.13) est compatible. D'où, $\chi(G^-(z(\gamma))) \leq 0$ d'après la proposition 3.9.

(i) \longleftarrow (ii)

Nous supposons maintenant qu'on a $\chi(G^-(z(\gamma))) \leq 0$, le système 3.13 est alors compatible, et d'après la proposition 3.9, le vecteur spectral de la fonction G^+ est positif. □

A travers la proposition 3.9, nous avons indiqué que le système initial donné par 3.13 peut admettre deux représentations possibles $G^-(z(\gamma)) \leq z(\gamma)$ ou $z(\gamma) \leq G^+(z(\gamma))$. L'analyse de la compatibilité du système 3.13 est réduit à l'analyse de l'un des deux systèmes de type $(-\infty, (\min, +))$ ou $((\max, +), +\infty)$ qui sont équivalents.

Afin de prendre en compte l'horizon d'analyse nous allons considérer le modèle $((\max, +), (\min, +))$ avec :

$$\begin{cases} f_j^-(x(k+j), x(k+j-1), u(k+j)) = \bigoplus_{i=0}^1 A_i^- \otimes x(k+j-i) \oplus B^- \otimes u(k+j) \\ f_j^+(x(k+j), x(k+j-1), u(k+j)) = \bigwedge_{i=0}^1 A_i^+ \odot x(k+j-i) \wedge B^+ \odot u(k+j) \end{cases} \quad (3.14)$$

Remarque : Le système 3.14 peut être déduit facilement du système 3.13. Une conséquence est une éventuelle augmentation de la dimension du vecteur d'état.

◇

Proposition 3.11. *Le système 3.14 est compatible sur l'horizon h si et seulement si, $\chi(\Phi_{g_h^+}(z_h)) \geq 0$ (respectivement $\chi(\Phi_{g_h^-}(z_h)) \leq 0$).*

Les fonctions $g_h^+(z_h(k))$ et $g_h^-(z_h(k))$ sont données respectivement par 3.7 et 3.11.

Corollaire 3.12. *Soit le modèle d'intervalle donné par 3.14, si ce système est compatible sur l'horizon h , alors les deux points suivants sont équivalents :*

- (i) $\chi(\Phi_{g_h^+}(z_h)) \geq 0$ avec $g_h^+(z_h(k)) = F_h^+(z_h(k)) \wedge M^t \setminus z_h(k) \wedge z_h(k+1)$,
- (ii) $\chi(\Phi_{g_h^-}(z_h)) \leq 0$ avec $g_h^-(z_h(k)) = F_h^-(z_h(k)) \oplus N^t \setminus' z_h(k) \oplus z_h(k-1)$.

Les démonstrations pour la proposition 3.11 et le corollaire 3.12 sont analogues à celles données respectivement pour la proposition 3.9 et le corollaire 3.10.

3.3 Vivacité dans les graphes d'événements p-temporels

3.3.1 État de l'art

3.3.1.1 Approche classe d'états

L'approche basée sur le graphe des marquages accessibles dans le cadre de l'étude des réseaux de Petri généralisés a souvent montré une incapacité à l'analyse et la vérification de la vivacité dans le cas général [Mur89]. Cette méthode a pour objectif de trouver toutes les séquences possibles et tous les états possibles. Le fait d'introduire du temps sur un réseau de Petri rend cette analyse encore plus difficile. L'analyse énumérative (on dit aussi l'analyse par énumération) est une méthode qui permet de mettre en évidence les propriétés d'un système modélisé par des réseaux de Petri p-temporels. Elle est basée sur l'approche dite "classe d'états" qui nécessite la construction d'un graphe de couverture (arbre de couverture) : elle permet de caractériser la situation du réseau à tout moment donné. L'intérêt de l'analyse énumérative est de limiter le nombre d'états accessibles à partir d'un état donné.

Nous exposons brièvement les principes de l'approche classe d'états. Nous savons que le passage de l'état initial à un état accessible, se fait par le franchissement d'une séquence de transitions σ . Plusieurs instants possibles de tir d'une transition x_i de cette séquence peuvent exister, d'où le grand nombre d'états possibles que l'on peut obtenir, et qui correspond au franchissement d'une même transition x_i . Pour remédier à ce problème, on va définir alors un ensemble d'états qui regroupe tous les états accessibles depuis l'état initial, par le franchissement à tous les instants possibles qui correspondent à la même séquence σ . Ainsi, la classe d'états est définie par cet ensemble d'états associée à la séquence σ .

Une classe d'état est définie comme un couple $\langle M, D \rangle$, où M est le marquage commun à tous les états de cette classe. Les intervalles de tir potentiels associés aux jetons, sont définis dans le domaine D . Les démarches qui permettent d'établir les conditions de franchissement d'une transition à partir d'une classe d'états donnée, et ainsi de calculer la classe suivante, sont données en détail dans [Kha97] et [BD91]. La construction du graphe des classes vient juste après le calcul de toutes les classes d'états possibles. Grâce à ce graphe, on pourra connaître tous les états accessibles possibles, déterminer les séquences de tir réalisables et enfin vérifier les propriétés d'un graphe d'événements p-temporel. En résumé, l'approche classe d'états, permet de connaître toutes les évolution possibles d'un système.

Toutefois, cette approche présente des difficultés pour la construction de l'arbre de couverture, puisque le calcul d'une classe d'état suivante est souvent très difficile. En plus, on a souvent un risque d'explosion combinatoire du nombre d'états possibles. Aussi, il a été démontré dans [BD91] (dans le cadre d'études sur les RdPs t-temporels) que si les bornes des intervalles temporels ne sont pas finies et/ou le réseau n'est pas borné alors, le nombre de classes d'états est infini. Ce même résultat a été confirmé dans les travaux de [Kha97] pour les RdPs p-temporels.

3.3.1.2 Approche état entier

Plus commode pour vérifier et analyser les propriétés d'un graphe d'événement p-temporels, l'approche dite "état entier" a été proposée dans [Kha97] et [Pop91]¹. Elle est caractérisée par le fait que le franchissement des transitions d'une séquence se fait à des instants entiers. Bien que les dates de franchissement des transitions peuvent être réelles, l'approche état entier ne s'intéresse qu'aux dates entières. L'examen de départ d'un jeton d'une place à une date entière devient systématique. Le calcul de l'état suivant devient alors plus facile (voir la figure 3.1).

Nous rappelons ici un résultat énoncé dans [Kha97] relatif à l'analyse de la vivacité des réseaux de Petri p-temporels. Ce résultat stipule qu'un réseau est vivant si et seulement si, dans le graphe des états entiers, pour chaque transition, il existe un arc associé et un chemin qui relie

¹l'approche état entier dans [Pop91] a pour objectif l'étude des RdPs t-temporels

cet arc, à tous les autres arcs associés aux autres transitions du réseau.

En revanche, l'approche état entier présente aussi des limitations. Tout d'abord, elle ne traite que les RdPs p-temporels dont les bornes des intervalles associés aux places, sont finies. Ensuite, pour les intervalles temporels, la différence entre la borne minimale et maximale, ne doit pas être très grande. Autrement dit, il faut considérer des intervalles réduits. En plus, cette méthode ne permet pas de suivre tous les états possibles et accessibles.

Exemple 3.1 : .

On considère ici un exemple déjà exposé dans [Kha97] d'un graphe d'événements p-temporel.

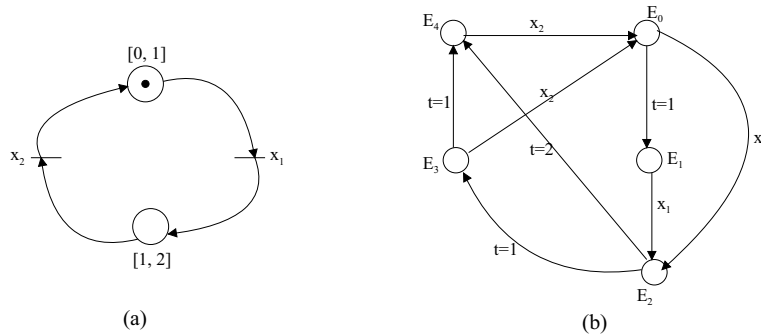


Figure 3.1 : Un graphe d'événements p-temporel et le graphe des états entiers correspondant

Cet exemple nous permet de montrer l'application de cette approche afin d'analyser la vivacité du graphe de la figure 3.2a. Les notations suivantes sont utilisées dans cet exemple :

- E_i : un état entier i décrivant le graphe
- M_i : marquage du graphe correspondant à l'état E_i
- G_i : l'âge des jetons correspondant au marquage M_i
- : le symbole "-" indique qu'il n'y a pas de jetons dans la place associée

Les états obtenus sont :

$$E_0 = \left(M_0 = (1, 0), G_0 = \begin{pmatrix} 0 \\ - \end{pmatrix} \right), E_1 = \left(M_1 = (1, 0), G_0 = \begin{pmatrix} 1 \\ - \end{pmatrix} \right),$$

$$E_2 = \left(M_2 = (0, 1), G_2 = \begin{pmatrix} - \\ 0 \end{pmatrix} \right), E_3 = \left(M_3 = (0, 1), G_3 = \begin{pmatrix} - \\ 1 \end{pmatrix} \right)$$

et $E_4 = \left(M_4 = (0, 1), G_4 = \begin{pmatrix} - \\ 2 \end{pmatrix} \right)$

A partir de l'analyse du graphe de la figure 3.1b, nous pouvons conclure que le graphe d'événements p-temporel de la figure 3.1a est vivant.

En résumé, les approches précédentes sont basées sur des combinatoires conduisent à des explosions du nombre d'états. Il est donc nécessaire de développer une approche spécifique aux graphes d'événements p-temporels.

3.3.2 Re-formulation du modèle

Dans le cadre d'une approche que nous proposerons dans cette partie pour l'analyse de la vivacité, il est intéressant d'obtenir un modèle plus concis à partir d'un graphe de départ quelconque. Plus précisément, on veut aboutir à un graphe d'événements p-temporel dans lequel toute place qui relie deux transitions internes, contient exactement un jeton, et les places liées aux transitions d'entrée et de sortie ne contiennent pas de jetons.

Une contrainte aux transformations nécessaires est de modifier le modèle du graphe d'événements p-temporel sans cacher ou supprimer d'éventuels problèmes dans le modèle non simplifié : l'image du graphe d'événements p-temporel doit encore être fidèle. Pour répondre à ce problème, nous allons présenter maintenant une démarche que nous avons adoptée pour la re-formulation des graphes d'événements p-temporels dont le modèle est donné par le système 2.26.

Étape 1 : réduction de l'horizon de l'équation ARMA.

La réduction de l'horizon de l'équation ARMA est interprétée graphiquement par un éclatement de chaque place, qui contient un nombre de marques strictement supérieur à un, en un ensemble de places contenant chacune, exactement, une marque. Évidemment, des transitions intermédiaires (fictives) seront aussi introduites entre les places nouvellement créés. Après cet éclatement, seule la place d'origine gardera son intervalle de temporisation et nous associons des intervalles de temps nuls $([0, 0])$ au reste des places que nous venons de créer. Le comportement du graphe reste identique après cette transformation. En effet, les $m - 1$ jetons distribués sur les p_{jm-1} places (voir figure 3.2b) dont les temporisations sont nulles, seront obligés de quitter ces places dès l'instant $t = 0$. Tous ces jetons rejoindront la place p_{jm} et pourront à nouveau franchir x_i entre les instants a_j et b_j comme c'était prévu initialement, c'est à dire, avant la transformation du graphe 3.2a en 3.2b.

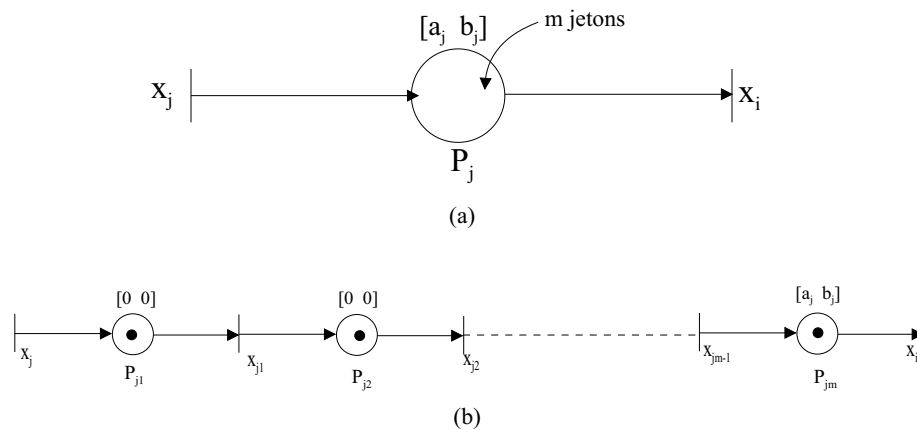


Figure 3.2 : Deux graphes d'événements p-temporels

Le graphe d'événements p-temporel de la figure 3.2a est équivalent au graphe de la figure 3.2b. En effet, Pour que le graphe de la figure 3.2a respecte un fonctionnement admissible, $x_i(\gamma)$ doit satisfaire : $a_j \otimes x_j(\gamma)\gamma^{m_j} \leq x_i(\gamma) \leq b_j \odot x_j(\gamma)\gamma^{m_j}$ Or, un fonctionnement admissible du graphe de la figure 3.2b s'exprime par le système suivant :

$$\left\{ \begin{array}{l} 0 \otimes x_j(\gamma)\gamma^1 \leq x_{j_1}(\gamma) \leq 0 \odot x_j(\gamma)\gamma^1 \\ 0 \otimes x_{j_1}(\gamma)\gamma^1 \leq x_{j_2}(\gamma) \leq 0 \odot x_{j_1}(\gamma)\gamma^1 \\ \vdots \\ 0 \otimes x_{j_{m-2}}(\gamma)\gamma^1 \leq x_{j_{m-1}}(\gamma) \leq 0 \odot x_{j_{m-2}}(\gamma)\gamma^1 \\ a_j \otimes x_{j_{m-1}}(\gamma)\gamma^1 \leq x_i(\gamma) \leq b_j \odot x_{j_{m-1}}(\gamma)\gamma^1 \end{array} \right.$$

Par une simple substitution de $x_{j_{m-1}}$ dans la dernière équation du système précédent par sa formule donnée par l'avant dernière équation du même système, on aboutit à : $a_j \otimes x_{j_{m-2}}(\gamma)\gamma^2 \leq x_i(\gamma) \leq b_j \odot x_{j_{m-2}}(\gamma)\gamma^2$. Nous procédons ensuite à des substitutions successives de la même manière et en parcourant tout le système, on exprime $x_i(\gamma)$ en fonction de $x_j(\gamma)$. On retrouve le modèle : $a_j \otimes x_j(\gamma)\gamma^{m_j} \leq x_i(\gamma) \leq b_j \odot x_j(\gamma)\gamma^{m_j}$. Les deux graphes 3.2a et 3.2b sont représentés par le même modèle mathématique, d'où la conclusion sur l'équivalence entre les deux derniers systèmes.

Nous passons maintenant à la deuxième étape de cette procédure de re-formulation.

Étape 2 : suppression de la partie statique.

Supprimer la partie statique du système, signifie transformer le graphe de telle façon que l'on obtienne un modèle décrivant un graphe dans lequel les places (hormis celles d'entrée et de sortie), contiennent exactement un jeton. Notons que lors de la transformation effectuée dans l'étape 1, une extension du vecteur d'état se produit. Ce nouveau vecteur d'état sera donné par \mathcal{X} . La nouvelle présentation d'état donnée par :

$$\left\{ \begin{array}{l} \mathcal{X} \geq \gamma^0 \mathbf{A}_0^- \mathcal{X} \oplus \gamma^1 \mathbf{A}^- \mathcal{X} \\ \mathcal{X} \geq \gamma^0 \mathbf{A}_0^+ \mathcal{X} \oplus \gamma^{-1} \mathbf{A}^+ \mathcal{X} \end{array} \right. \quad (3.15)$$

A travers un exemple pédagogique simple, nous allons essayer de supprimer la partie statique par deux approches différentes.



Figure 3.3 : Graphe d'événements p-temporels et un graphe simplifié

Prenons le graphe d'événements p-temporels donné par la figure 3.3a. Nous allons essayer de simplifier le graphe de la figure 3.3a dont le modèle d'état est de la forme :

$$\begin{cases} x \geq \gamma^0 A_0^- x \oplus \gamma^1 A_1^- x \\ x \geq \gamma^0 A_0^+ x \oplus \gamma^{-1} A_1^+ x \end{cases} \quad (3.16)$$

avec $x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$, $A_0^- = \begin{pmatrix} \varepsilon & \varepsilon \\ 4 & \varepsilon \end{pmatrix}$, $A_1^- = \begin{pmatrix} \varepsilon & 0 \\ \varepsilon & \varepsilon \end{pmatrix}$, $A_0^+ = \begin{pmatrix} \varepsilon & -3 \\ \varepsilon & \varepsilon \end{pmatrix}$ et $A_1^+ = \begin{pmatrix} \varepsilon & \varepsilon \\ -10 & \varepsilon \end{pmatrix}$

Une première approche consiste à transposer directement les résultats relatifs à la simplification des GET (voir paragraphe 2.3.3) sur les graphes d'événements p-temporels. En appliquant le corollaire 1.36 sur chacune des inégalités du système 3.16, nous obtenons un nouveau modèle 3.17 correspondant au graphe de la figure 3.3b.

$$\begin{cases} x \geq \gamma^1 (A_0^-)^* \otimes A_1^- x \\ x \geq \gamma^{-1} (A_0^+)^* \otimes A_1^+ x \end{cases} \quad (3.17)$$

avec $(A_0^-)^* \otimes A_1^- = \begin{pmatrix} \varepsilon & 0 \\ \varepsilon & 4 \end{pmatrix}$ et $(A_0^+)^* \otimes A_1^+ = \begin{pmatrix} -13 & \varepsilon \\ -10 & \varepsilon \end{pmatrix}$

Analysons maintenant la figure 3.3a. Sans passer par la construction du graphe de calcul, nous pouvons déjà constater qu'initialement, une synchronisation au niveau de la transition x_2 est impossible à cause du marquage initial et de l'incompatibilité des temporisations des places p_2 et p_3 . En effet, le jeton qui franchira x_1 créera en même temps un jeton dans chacune des places p_2 et p_3 , et puisque $[1, 3] \cap [4, 5] = \emptyset$ les deux jetons vont mourir. Un phénomène de désynchronisation sera donc observé.

Maintenant, la constatation que l'on peut faire sur le graphe simplifié 3.3b, est qu'il n'y a plus de problème lié à la mort de jetons, et que ce graphe est vivant. Le problème de mort de jetons qui surgissait dans le graphe figure 3.3a a complètement disparu après sa re-formulation. Remarquons aussi que l'on a affecté à la borne min (respectivement max) de l'intervalle associé à la place p_3 de la figure 3.3b, la valeur 0 puisque $x_1(k) \geq 0 + x_1(k-1)$ est toujours vérifié.

(respectivement p_2 la valeur T , car à partir du modèle 3.17 on a $x_2(k) \geq \varepsilon + x_2(k+1)$, d'où $x_2(k) \leq Tx_2(k-1)$).

Nous pouvons donc conclure que l'approche de simplification immédiate que nous avons présentée ci-dessus, n'est pas valable sur les graphes d'événements p-temporels. Ainsi, d'éventuelles propriétés peuvent disparaître au cours de cette transformation en utilisant cette approche. Le résultat suivant (proposition 3.13) donne les conditions de simplification d'un graphe d'événements p-temporel sans qu'il ne perde ses propriétés.

En posant $\mathbf{A}_0^- \oplus \mathbf{A}_0^+ = \mathbf{A}_0$, le système 3.15 peut se réécrire :

$$\mathcal{X} \geq \gamma^1 \mathbf{A}^- \mathcal{X} \oplus \gamma^0 \mathbf{A}_0 \mathcal{X} \oplus \gamma^{-1} \mathbf{A}^+ \mathcal{X} \quad (3.18)$$

Proposition 3.13. *Pour qu'un graphe d'événements p-temporels modélisé par 2.26 admette un graphe simplifié dont le modèle d'état est donnée par le système suivant :*

$$\mathcal{X} \geq (\gamma^1 \mathcal{A}^- \oplus \gamma^{-1} \mathcal{A}^+) \mathcal{X} \quad (3.19)$$

il faut que \mathbf{A}_0^* existe dans \mathbb{R}_{max} et soit donc différent de T . Le terme T représente ici est une matrice dont tous les coefficients tendent vers l'infini $+\infty$.

Preuve : Si \mathbf{A}_0^* existe, alors la plus petite solution de l'équation 3.18 est donnée par :

$$\mathcal{X} = (\mathbf{A}_0)^* [\gamma^1 \mathbf{A}^- \mathcal{X} \oplus \mathcal{X} \oplus \gamma^{-1} \mathbf{A}^+ \mathcal{X}].$$

Chaque trajectoire \mathcal{X} du système peut alors vérifier l'équation suivante :

$$\mathcal{X} \geq (\mathbf{A}_0)^* [\gamma^1 \mathbf{A}^- \mathcal{X} \oplus \gamma^{-1} \mathbf{A}^+ \mathcal{X}] \quad (3.20)$$

On pose $\mathcal{A}^- = \mathbf{A}_0^* \mathbf{A}^-$ et $\mathcal{A}^+ = \mathbf{A}_0^* \mathbf{A}^+$. D'où le nouveau système 3.19.

□

En d'autres termes, nous pouvons passer d'un graphe modélisé par le système 3.18 à un graphe dont le modèle est donné par 3.19 sous la condition que la matrice \mathbf{A}_0^* converge. Cette condition est en réalité une condition de vivacité du graphe d'événements p-temporel.

Reprenons le graphe de la figure 3.3a. Pour pouvoir appliquer la proposition 3.13, nous considérons le modèle suivant décrivant le fonctionnement du graphe 3.3a :

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \gamma^1 \begin{pmatrix} \varepsilon & 0 \\ \varepsilon & \varepsilon \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \oplus \gamma^0 \begin{pmatrix} \varepsilon & -3 \\ 4 & \varepsilon \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \oplus \gamma^{-1} \begin{pmatrix} \varepsilon & \varepsilon \\ -10 & \varepsilon \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \quad (3.21)$$

avec $\mathbf{A}_0 = A_0^- \oplus A_0^+ = \begin{pmatrix} \varepsilon & -3 \\ 4 & \varepsilon \end{pmatrix}$. Le calcul de $(\mathbf{A}_0)^*$ donne $(\mathbf{A}_0)^* = \begin{pmatrix} T & T \\ T & T \end{pmatrix}$, d'où il est impossible de simplifier le graphe de la figure 3.3a d'après la proposition 3.13.

Nous considérons maintenant un deuxième cas où nous modifions l'intervalle de temporisation associé à la p_3 du graphe de la figure 3.3a. Le nouvel intervalle de temps associé à p_3 est : $[2, 5]$. Nous gardons le même modèle donné par le système 3.16, seuls les coefficients de la matrice A_0^- changent : $A_0^-(2, 1) = 2$. Avec ce changement, on n'assistera pas au phénomène de désynchronisation et le nouveau modèle préservera la vivacité des jetons. Le calcul du \mathbf{A}_0^* donne : $(\mathbf{A}_0)^* = \begin{pmatrix} 0 & -3 \\ 2 & 0 \end{pmatrix}$, d'où d'après la proposition 3.13, il est possible de simplifier le nouveau graphe de la figure 3.3a. Le nouveau modèle est donné par :

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \geq (\gamma^1 \mathbf{A}_0^* \otimes A_1^- \oplus \gamma^{-1} \mathbf{A}_0^* \otimes A_1^+) \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \quad (3.22)$$

avec $(\mathbf{A}_0)^* \otimes A_1^- = \begin{pmatrix} \varepsilon & 0 \\ \varepsilon & 2 \end{pmatrix}$ et $(\mathbf{A}_0)^* \otimes A_1^+ = \begin{pmatrix} -13 & \varepsilon \\ -10 & \varepsilon \end{pmatrix}$

Le nouveau graphe correspondant au nouveau modèle est le même que celui de la figure 3.3b. Seule la temporisation associée à la place p_2 change de valeur. La nouvelle temporisation est : $[2, T]$.

Après cette phase de reformulation et ce premier résultat sur la vivacité statique des graphes d'événements p -temporels, nous passons maintenant à l'étude de la vivacité dynamique de ces derniers. Pour cela, nous allons analyser le système 3.19 en vérifiant l'existence d'une trajectoire admissible. Ceci, fera l'objet d'une étude détaillée, par la suite.

Un graphe d'événements p -temporel est vivant, alors toutes ses transitions le sont aussi. Cela veut dire, qu'il existe toujours un marquage M tel qu'une transition x_i soit franchissable d'après la définition de la vivacité. Or, franchir une transition à un instant donné se traduit par un \mathcal{X}_i fini pour un événement (horizon) donné. Cela est vrai pour toutes les transitions sur tout l'horizon considéré. D'où, une trajectoire \mathcal{X} finie sur un horizon h .

3.3.3 Approche basée sur le graphe de calcul et le vecteur spectral

Nous proposons ici une approche basée sur l'exploitation du graphe de calcul relatif à un graphe d'événements p -temporel. Cette méthode utilise ce graphe de calcul afin de déterminer le vecteur spectral. Nous donnons dans un premier temps, un exemple pédagogique afin de montrer la construction du graphe de calcul pour un horizon de calcul donné $[k, k + 3]$.

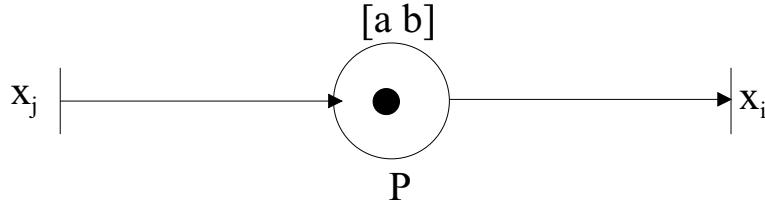


Figure 3.4 : Exemple d'un un graphe d'événements p-temporels

A partir du graphe de la figure 3.4, on obtient :

$$a \otimes \gamma^1 x_j(\gamma) \leq x_i(\gamma) \leq b \odot \gamma^1 x_j(\gamma) \quad (3.23)$$

Deux possibilités sont envisageables pour re-formuler le système 3.23. Nous aboutissons au premier système suivant :

$$\begin{cases} x_i(\gamma) \leq b\gamma^1 x_j(\gamma) \\ \gamma^1 x_j(\gamma) \leq -ax_i(\gamma) \end{cases} \quad (3.24)$$

La deuxième possibilité est donnée par :

$$\begin{cases} x_i(\gamma) \geq a\gamma^1 x_j(\gamma) \\ \gamma^1 x_j(\gamma) \geq -bx_i(\gamma) \end{cases} \quad (3.25)$$

Pour construire un graphe de calcul, on fait correspondre au système d'inéquations 3.24 (respectivement 3.25), le graphe donné par la figure 3.5b (respectivement 3.5a), qui traduit l'ensemble des contraintes modélisant le graphe de la figure 3.5.

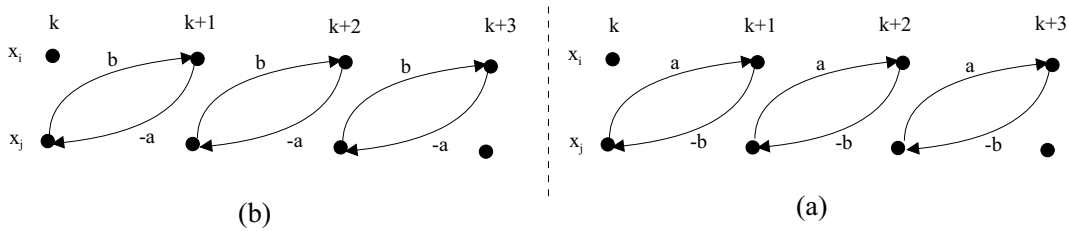


Figure 3.5 : Graphes de calcul sur l'horizon $[k, k+3]$ correspondant au graphe de la figure 3.4

Le calcul du vecteur spectral reposera sur le calcul du poids moyen des circuits dans l'un des graphes 3.5b ou 3.5a.

Le choix d'une re-formulation du modèle précédent, afin de le faire correspondre à un ensemble de contraintes par dessus, ou par dessous est arbitraire. D'où, une modélisation mathématique qui peut se mettre sous l'une des deux formes générales suivantes :

$$z_h(k) \leq g_h^+(z_h(k)), \quad z_h(k) \geq g_h^-(z_h(k))$$

où les fonctions $g_h^+(z_h(k))$ et $g_h^-(z_h(k))$ sont définies respectivement par 3.7 et 3.11. Comme il a été montré dans le chapitre 2, la modélisation des graphes d'événements p-temporels correspond aux systèmes de type $((\max, +), (\min, +))$, cette écriture est toujours possible. On peut donc appliquer la proposition 3.11 et le corollaire 3.12.

Nous introduisons maintenant un résultat qui analyse la vivacité de chacune des transitions du graphe d'événements p-temporel.

Proposition 3.14. *Dans un graphe d'événements p-temporel, les deux propositions suivantes sont équivalentes :*

- (i) *La transition x_i est vivante sur un horizon h si, $\chi_{i+jn}(\Phi_{g_h^+}(z_h)) \geq 0$.*
 - (ii) *La transition x_i est vivante sur un horizon h si, $\chi_{i+jn}(\Phi_{g_h^-}(z_h)) \leq 0$.*
- avec n est le nombre de transitions du graphe d'événements et $j \in [0, h]$.*

Preuve : Nous montrons tout d'abord la proposition (i).

On a $z_h(k) \leq g_h^+(z_h(k))$, et d'après le théorème 1.56, vérifier l'existence d'une solution finie pour cette dernière équation est équivalent à ce que le vecteur spectral de $g_h^+(z_h(k))$ soit positif. En particulier, $x_i(k+h)$ sera réalisable si et seulement si $\chi_{i+jn}(\Phi_{g_h^+}(z_h)) \geq 0$, puisque $\chi_{i+jn}(\Phi_{g_h^+}(z_h)) \in \chi(\Phi_{g_h^+}(z_h))$. On conclut que x_i est vivante sur l'horizon h .

La démonstration de la proposition (ii) est duale à (i).

La démonstration de l'équivalence entre les deux points (i) et (ii) est une application directe du corollaire 3.12.

□

Remarque : la valeur de $\chi_{i+jn}(\Phi_{g_h^+}(z_h))$ correspond au minimum (respectivement $\chi_{i+jn}(\Phi_{g_h^-}(z_h)) \leq 0$ correspond au maximum) de tous les poids moyens des circuits, en amont du noeud i , représentant la transition x_i sur l'horizon h .

◇

Exemple 3.2 :

Nous considérons le graphe d'événements p-temporel suivant pour illustrer la démarche. La figure 3.7 montre le graphe de calcul associé au graphe d'événements p-temporel de la figure 3.6.

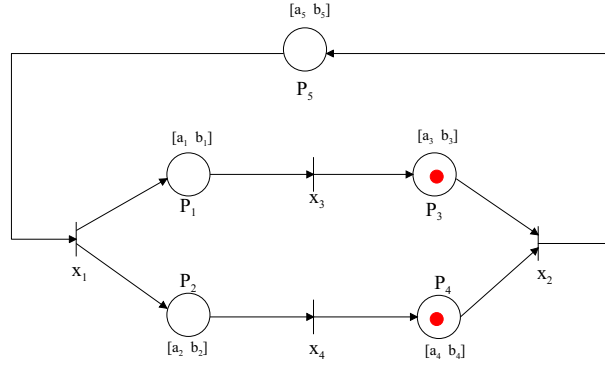


Figure 3.6 : Graphe d'événements p-temporel non vivant

Initialement, nous pouvons vérifier facilement que le graphe d'événements sous-jacent (sans considération des intervalles de temps associés aux places) au graphe de la figure 3.6, est vivant (chaque circuit contient un jeton). En associant des temporisations particulières à chaque place, nous pouvons noter que, malgré une vivacité assurée pour le graphe sous-jacent, le graphe temporel peut être dans un état de blocage total. Montrant différents comportements, plusieurs cas peuvent donc se présenter si l'on agit sur les limites des intervalles relatifs aux places.

La première étape de cette approche est de modéliser le système de la figure 3.6 par des équations d'état récurrentes sous la forme :

$$\left\{ \begin{array}{l} x_2(k) + a_5 \leq x_1(k) \leq x_2(k) + b_5 \\ (x_3(k-1) + a_3) \oplus (x_4(k-1) + a_4) \leq x_2(k) \\ \leq (x_3(k-1) + b_3) \wedge (x_4(k-1) + b_4) \\ x_1(k) + a_1 \leq x_3(k) \leq x_1(k) + b_1 \\ x_1(k) + a_2 \leq x_4(k) \leq x_1(k) + b_2 \end{array} \right. \quad (3.26)$$

La deuxième étape consiste à décomposer le système d'équations précédent 3.26 et le mettre sous la forme $x \leq g_h^+(x)$. Ainsi, nous obtenons le système :

$$\left\{ \begin{array}{l} x_1(k) \leq (x_2(k) + b_5) \wedge (x_3(k) - a_1) \wedge (x_4(k) - a_2) \\ x_2(k) \leq (x_1(k) - a_5) \wedge (x_3(k-1) + b_3) \wedge (x_4(k-1) + b_4) \\ x_3(k) \leq (x_1(k) + b_1) \wedge (x_2(k-1) - a_3) \\ x_4(k) \leq (x_1(k) + b_2) \wedge (x_2(k+1) - a_4) \end{array} \right.$$

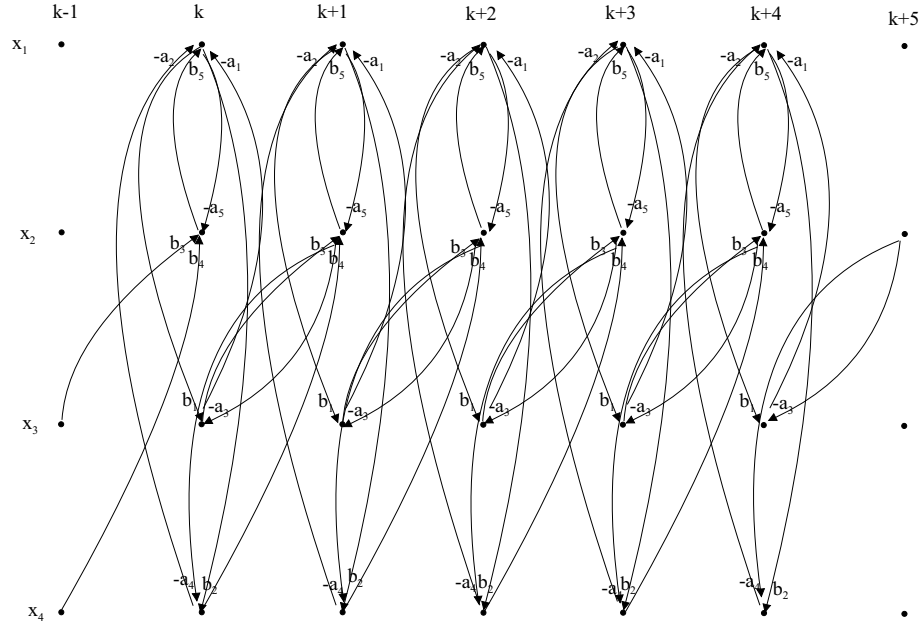


Figure 3.7 : Graphe de calcul associé au graphe de la figure 3.6

Cas 1 :

Nous considérons maintenant ce premier cas dans lequel nous fixons les bornes des intervalles comme suit :

$$[a_1 \ b_1] = [0, 1], [a_2 \ b_2] = [5, 6], [a_3 \ b_3] = [0, 1], [a_4 \ b_4] = [0, 1] \text{ et } [a_5 \ b_5] = [3, 4].$$

Nous calculons le vecteur spectral de g_h^+ à partir du graphe de la figure 3.7. Nous aboutissons aux résultats donnés ci-dessous :

$$\begin{aligned} \chi_2(\Phi_{g^+}(X_1)) &= \frac{1}{2} & \chi_1(\Phi_{g^+}(X_1)) &= \frac{1}{2} \\ \chi_6(\Phi_{g^+}(X_2)) &= -\frac{3}{4} & \chi_5(\Phi_{g^+}(X_2)) &= -\frac{3}{4} \end{aligned}$$

En résumé : $\chi(\Phi_{g^+}(X_1)) \geq 0$ et $\chi(\Phi_{g^+}(X_2)) < 0$

En appliquant la proposition 3.14, nous déduisons que le système est vivant sur l'horizon $k = 1$. En revanche, au delà de cet horizon, c'est-à-dire après $k = 2$, on détecte la mort des jetons. Le système perd ces derniers, et la propriété de vivacité n'est plus assurée.

Cas 2 :

Nous considérons un deuxième cas où les nouvelles temporisations sont données par :

$$[a_1 \ b_1] = [3, 4], [a_2 \ b_2] = [3, 4], [a_3 \ b_3] = [1, 2], [a_4 \ b_4] = [6, 7] \text{ et } [a_5 \ b_5] = [4, 5].$$

Nous exploitons aussi le graphe de calcul de la figure 3.7. Le calcul du vecteur spectral ($\chi_2(\Phi_{g^+}(X_1)) = -\frac{12}{5}$, $\chi_1(\Phi_{g^+}(X_1)) = -\frac{12}{5}$) et l'application de la proposition 3.14 montrent bien que la synchronisation de la transition x_2 ne peut être faite. Les deux jetons vont mourir, le système de la figure 3.6 se retrouve dans un état de blocage dès le premier tir car : $\chi(\Phi_{g^+}(X_1)) < 0$.

Nous avons montré, en utilisant cette approche basée sur le calcul du vecteur spectral, qu'elle est efficace pour repérer d'éventuels dysfonctionnements liés à une transition comme la mort de jetons, résultat d'une désynchronisation. En revanche, la complexité du problème de la recherche des circuits dans le graphe de calcul, augmente progressivement avec l'horizon du calcul. D'autre part, il n'est pas possible de conclure sur un horizon arbitraire grand. Ce problème sera exposé dans un deuxième exemple (voir paragraphe 3.4) ce qui montrera la nécessité de remédier au problème de la recherche des circuits dans un graphe de calcul dont l'horizon est grand.

3.3.4 Approche utilisant des séries de matrices

Comme nous l'avons souligné auparavant, un fonctionnement admissible (voir définition 2.14) garantit la vivacité du système. L'étude de tel fonctionnement peut se faire à travers l'étude des modèles algébriques qui le présentent. En l'occurrence, il s'agit ici d'étudier l'existence de trajectoires d'état admissibles. Cette notion de vivacité concerne l'évolution dynamique d'un graphe d'événements p-temporel qui garantit une évolution sans avoir de morts de jetons qui à leur tour, peuvent causer éventuellement le blocage du système.

Soit la représentation d'état suivante :

$$\mathcal{X} \geq (\gamma^1 \mathcal{A}^- \oplus \gamma^{-1} \mathcal{A}^+) \mathcal{X} \quad (3.27)$$

Nous introduisons dans un premier temps la définition d'un graphe dynamique $G_h(A, B)$

Définition 3.15. Graphe dynamique

Soit deux matrices A et B dans $\mathbb{R}_{max}^{n \times n}$. Le graphe dynamique induit par ces matrices sur un horizon h que l'on notera $G_h(A, B)$ est défini de la manière suivante : le noeud j_k correspond à la colonne j et le noeud i_k correspond la ligne i , pour tout k allant de 0 à h . Il existe un arc (j_{k-1}, i_k) de poids A_{ij} si $A_{ij} \neq \varepsilon$. De même, il existe un arc (j_{k+1}, i_k) de poids B_{ij} si $B_{ij} \neq \varepsilon$.

Nous donnons maintenant quelques résultats concernant l'étude et la vérification de l'existence des trajectoires finies ainsi que leur interprétation graphique.

Proposition 3.16. Soit une série de matrices $w_k = \mathcal{A}^- \otimes (w_{k-1})^* \otimes \mathcal{A}^+$ avec $w_1 = \mathcal{A}^- \otimes \mathcal{A}^+$. Une condition nécessaire de l'existence dans \mathbb{R}_{max} d'une trajectoire d'état sur un horizon infini, est que les matrices w_k aient des circuits de poids négatif ou nul.

Preuve :

Soit k , $\mathcal{X}(k) \geq \mathcal{A}^- \otimes \mathcal{X}(k-1) \oplus \mathcal{A}^+ \otimes \mathcal{X}(k+1)$ et
 $\mathcal{X}(k+1) \geq \mathcal{A}^- \otimes \mathcal{X}(k) \oplus \mathcal{A}^+ \otimes \mathcal{X}(k+2) =$
 $\mathcal{A}^- \otimes [\mathcal{A}^- \otimes \mathcal{X}(k-1) \oplus \mathcal{A}^+ \otimes \mathcal{X}(k+1)] \oplus \mathcal{A}^+ \otimes \mathcal{X}(k+2) \geq \mathcal{A}^- \otimes \mathcal{A}^- \otimes \mathcal{X}(k-1) \oplus \mathcal{A}^- \otimes \mathcal{A}^+ \otimes$
 $\mathcal{X}(k+1) \oplus \mathcal{A}^+ \otimes \mathcal{X}(k+2)$ (distributivité de \otimes par rapport à \oplus).

Or, $\mathcal{X}(k+1) \geq \mathcal{A}^- \otimes \mathcal{A}^+ \otimes \mathcal{X}(k+1) \oplus \mathcal{A}^+ \otimes \mathcal{X}(k+2) \oplus \mathcal{A}^- \otimes \mathcal{A}^- \otimes \mathcal{X}(k-1)$

Une condition nécessaire pour l'existence de $\mathcal{X}(k+1) \neq T$ est que tous les circuits du graphe $G(\mathcal{A}^- \otimes \mathcal{A}^+)$ n'aient pas de circuits de poids positifs. Par conséquent, $\mathcal{X}(k+1) \geq (\mathcal{A}^- \otimes \mathcal{A}^+)^* [\mathcal{A}^+ \otimes \mathcal{X}(k+2) \oplus \mathcal{A}^- \otimes \mathcal{A}^- \otimes \mathcal{X}(k-1)]$.

$$w_1 = (\mathcal{A}^- \otimes \mathcal{A}^+) \text{ et } \beta_1^- = \mathcal{A}^- \otimes \mathcal{A}^- \otimes \mathcal{X}(k-1)$$

Une récurrence est appliquée. Supposons alors que la propriété H_d soit vérifiée telle que H_d est donnée par : $\mathcal{X}(k+d) \geq w_d \otimes \mathcal{X}(k+d) \oplus \mathcal{A}^+ \otimes \mathcal{X}(k+d+1) \oplus \beta_d^-$ et $w_d = (\mathcal{A}^- \otimes w_{d-1}^* \otimes \mathcal{A}^+)$ avec tous les circuits de $G(w_d)$ ayant des poids non-positifs.

En ce qui concerne l'initialisation de la récurrence, la propriété H_1 est vérifiée car on a déjà :

$$\mathcal{X}(k+1) \geq \mathcal{A}^- \otimes \mathcal{A}^+ \otimes \mathcal{X}(k+1) \oplus \mathcal{A}^+ \otimes \mathcal{X}(k+2) \oplus \mathcal{A}^- \otimes \mathcal{A}^- \otimes \mathcal{X}(k-1)$$

Maintenant, nous allons montrer que la propriété H_{d+1} est vérifiée. On a :

$$\mathcal{X}(k+d) \geq (w_d)^* \otimes [\mathcal{A}^+ \otimes \mathcal{X}(k+d+1) \oplus \beta_d^-]$$

$$\mathcal{X}(k+d+1) \geq \mathcal{A}^- \otimes \mathcal{X}(k+d) \oplus \mathcal{A}^+ \otimes \mathcal{X}(k+d+2) \geq$$

$$\mathcal{A}^- \otimes [(w_d)^* \otimes [\mathcal{A}^+ \otimes \mathcal{X}(k+d+1) \oplus \beta_d^-]] \oplus \mathcal{A}^+ \otimes \mathcal{X}(k+d+2) \geq$$

$$\mathcal{A}^- \otimes (w_d)^* \otimes \mathcal{A}^+ \otimes \mathcal{X}(k+d+1) \oplus \mathcal{A}^- \otimes (w_d)^* \otimes \beta_d^- \oplus \mathcal{A}^+ \otimes \mathcal{X}(k+d+2)$$

Si $w_{d+1} = (\mathcal{A}^- \otimes (w_d)^* \otimes \mathcal{A}^+)$ est tel que tous les circuits dans le graphe $G(w_{d+1})$ ont un poids négatif et $\beta_{d+1}^- = \mathcal{A}^- \otimes (w_d)^* \otimes \beta_d^-$

$$\text{alors, } \mathcal{X}(k+d+1) \geq (w_{d+1})^* \otimes [\mathcal{A}^+ \otimes \mathcal{X}(k+d+2) \oplus \beta_{d+1}^-]$$

En conclusion, H_{d+1} est vérifiée, d'où le résultat énoncé dans la proposition 3.16. □

La proposition suivante donne une interprétation graphique des séries de matrices w_k .

Proposition 3.17. *La matrice $(w_k)^*$ représente le poids maximal de tous les chemins entre le noeud i_k et le noeud j_k dans le graphe dynamique induit $G_k(\mathcal{A}^-, \mathcal{A}^+)$ développé sur un horizon k . Chaque composante de la diagonale $(w_k)_{ii}^*$ représente le maximum entre le circuit de plus grand poids et zéro.*

Preuve :

$$\mathcal{X}(k)_i \geq (\mathcal{A}^-)_{il} \otimes \mathcal{X}(k-1)_l \geq (\mathcal{A}^-)_{il} \otimes (\mathcal{A}^+)_{lj} \mathcal{X}(k)_j$$

$(w_1)_{ij} = (\mathcal{A}^- \otimes \mathcal{A}^+)_{ij}$ représente le poids le plus grand entre les chemins allant de j_k à i_k ($j_k \rightarrow l_{k-1}$) et ($l_{k-1} \rightarrow i_k$) et $(w_1)^*$ représente le poids le plus grand des chemins (zéro s'il n'y a pas de chemin) allant successivement d'un noeud correspondant à un événement de numéro k à $k-1$ et de $k-1$ à k , ensuite de k à $k-1$ et de $k-1$ à k , et ainsi de suite.

$(w_2)_{ij} = (\mathcal{A}^- \otimes (w_1)^* \otimes \mathcal{A}^+)_{ij}$ représente le poids le plus grand entre les chemins allant de j_k à i_k en passant successivement par des noeuds correspondant à des événements de numéros k , $k-1$ à $k-2$ et de $k-2$, $k-1$ à k .

Par conséquent, chaque circuit dans le graphe de calcul associé sur l'horizon qui va de $k-1$ à $k+1$, est défini par un chemin qui va de i_{k+1} à i_{k+1} et passant éventuellement dans les noeuds d'indices k et $k-1$, est contenu dans $(w_2)^*$. Le même argument est répété sur un horizon plus

large. □

Proposition 3.18. *La série de matrices $w_0 = \varepsilon$ et $w_k = \mathcal{A}^- \otimes (w_{k-1})^* \otimes \mathcal{A}^+$ est non-décroissante pour tout $k \geq 1$.*

Preuve : Nous allons démontrer cette proposition par récurrence. Supposons alors que la propriété $H_k : w_{k-1} \geq w_{k-2}$ est vérifiée et montrons que H_{k+1} l'est aussi.

D'abord on a :

$w_2 \geq w_1 = \mathcal{A}^- \otimes \mathcal{A}^+$ parce que

$$\begin{aligned} w_2 &= \mathcal{A}^- \otimes (w_1)^* \otimes \mathcal{A}^+ = \mathcal{A}^- \otimes \mathcal{A}^+ \oplus \mathcal{A}^- \otimes (w_1) \otimes \mathcal{A}^+ \oplus \dots \\ &= w_1 \oplus \mathcal{A}^- \otimes (w_1) \otimes \mathcal{A}^+ \oplus \dots \geq w_1 \end{aligned}$$

D'où l'initialisation de la récurrence est vérifiée, elle peut être faite aussi par : $w_1 \geq w_0 = \varepsilon$

Montrons maintenant que $w_k \geq w_{k-1}$.

$$\begin{aligned} w_k &= \mathcal{A}^- \otimes (w_{k-1})^* \otimes \mathcal{A}^+ = \\ &= \mathcal{A}^- \otimes \mathcal{A}^+ \oplus \mathcal{A}^- \otimes (w_{k-1}) \otimes \mathcal{A}^+ \oplus \mathcal{A}^- \otimes (w_{k-1})^2 \otimes \mathcal{A}^+ \oplus \dots \end{aligned}$$

Comme $w_{k-1} \geq w_{k-2}$ et par l'isotonie du produit, $w_k \geq \mathcal{A}^- \otimes \mathcal{A}^+ \oplus \mathcal{A}^- \otimes (w_{k-2}) \otimes \mathcal{A}^+ \oplus \mathcal{A}^- \otimes (w_{k-2})^2 \otimes \mathcal{A}^+ \oplus \dots = \mathcal{A}^- \otimes (w_{k-2})^* \otimes \mathcal{A}^+ = w_{k-1}$, d'où la propriété H_{k+1} est vérifiée. En conclusion la série de matrices w_k est non-décroissante pour tout $k \geq 1$. □

Comme la série w_k est non-décroissante, des circuits de poids positif peuvent être créés dans $G(\mathcal{A}^-, \mathcal{A}^+)$ sur un horizon h , qui vont entraîner la non-existence de trajectoires sans mort de jetons sur un horizon infini. A l'opposé, s'il existe k_1 tel que $w_k = w_{k-1}$ pour $k \geq k_1$ avec $(w_k)_{ij} \in \mathbb{R}_{max}$, les poids des circuits de $G_h(\mathcal{A}^-, \mathcal{A}^+)$ sont seulement négatifs sur chaque horizon h et on peut conclure que le graphe d'événements p-temporel est vivant sur un horizon infini.

3.4 Illustration

Nous présentons maintenant un exemple qui nous permettra d'illustrer les deux approches introduites ci-dessus et d'étudier le problème de la vivacité des graphes d'événements p-temporels. Cet exemple nous permettra aussi de montrer l'avantage que présente l'approche utilisant les séries de matrices par rapport à celle utilisant le vecteur spectral.

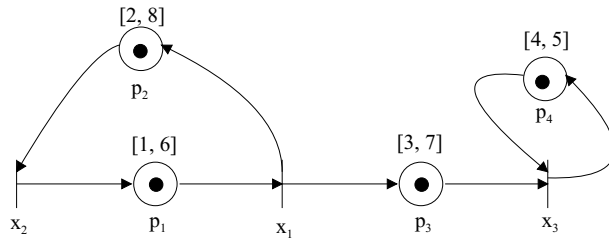


Figure 3.8 : Graphe d'événements p-temporel vivant

Approche basée sur le vecteur spectral et le graphe de calcul

Une première modélisation possible pour décrire le graphe d'événements de la figure 3.8 est donnée comme suit :

$$\begin{cases} 1 + x_2(k - 1) \leq x_1(k) \leq 6 + x_2(k - 1) \\ 2 + x_1(k - 1) \leq x_2(k) \leq 8 + x_1(k - 1) \\ (3 + x_1(k - 1)) \oplus (4 + x_3(k - 1)) \leq x_3(k) \leq (7 + x_1(k - 1)) \oplus (5 + x_3(k - 1)) \end{cases} \quad (3.28)$$

Le dernier système 3.28, peut se mettre sous la forme suivante :

$$\begin{cases} x_1(k) \leq (6 + x_2(k - 1)) \wedge (-2 + x_2(k + 1)) \wedge (-3 + x_3(k + 1)) \\ x_2(k) \leq (8 + x_1(k - 1)) \wedge (-1 + x_1(k + 1)) \\ x_3(k) \leq ((7 + x_1(k - 1)) \oplus (5 + x_3(k - 1))) \wedge (-4 + x_3(k + 1)) \end{cases} \quad (3.29)$$

A partir du système 3.29, nous construisons le graphe de calcul suivant (voir figure 3.9).

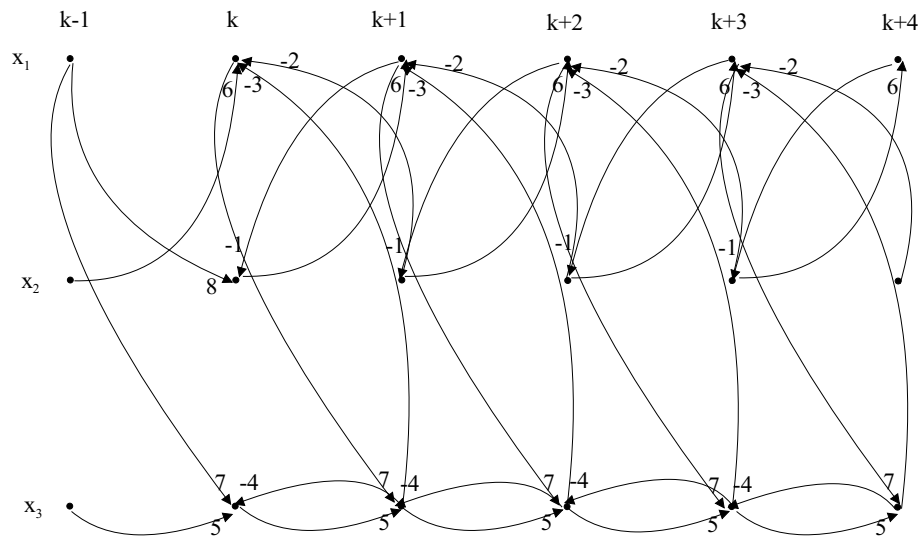


Figure 3.9 : Graphe de calcul associé au graphe de la figure 3.8

D'après ce graphe, $\chi(\Phi_{g^+}(X_3)) \geq 0$. Par conséquent, l'application de la proposition 3.14 implique que le graphe de la figure 3.8 est vivant sur l'horizon $h = 3$. En revanche, la difficulté principale qui nous est posée, est de repérer les circuits dans le graphe de calcul (voir figure 3.9) pour tout h supérieur à 3 : l'approche ne permet pas de conclure quand la taille du graphe est infinie.

Approche basée sur les séries de matrices

Une deuxième possibilité pour modéliser le graphe de la figure 3.28 est donnée par :

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} \geq \begin{pmatrix} \varepsilon & 1 & \varepsilon \\ 2 & \varepsilon & \varepsilon \\ 3 & \varepsilon & 4 \end{pmatrix} \gamma^1 \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} \oplus \begin{pmatrix} \varepsilon & -8 & -7 \\ -6 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & -5 \end{pmatrix} \gamma^{-1} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}.$$

Les résultats de simulations donnent :

$$w_1 = \begin{pmatrix} -5 & \varepsilon & \varepsilon \\ \varepsilon & -6 & -5 \\ \varepsilon & -5 & -1 \end{pmatrix}; w_2 = \begin{pmatrix} -5 & \varepsilon & -9 \\ \varepsilon & -6 & -5 \\ -7 & -5 & -1 \end{pmatrix}; w_3 = \begin{pmatrix} -5 & -19 & -9 \\ -18 & -6 & -5 \\ -7 & -5 & -1 \end{pmatrix} \text{ et } w_4 = w_3.$$

Nous remarquons que $w_4 = w_k$ pour tout $k \geq 4$, d'où, on peut dire qu'il existe des trajectoires finies (admissibles) sans mort de jetons sur un horizon infini, puisque w_k converge quand k tend vers l'infini. Par conséquent, en appliquant la proposition 3.16, il existe une trajectoire d'état fini. Le graphe de la figure 3.8 est alors vivant sur un horizon infini.

3.5 Conclusion

Dans ce chapitre, nous avons analysé dans un premier temps, la compatibilité dans des systèmes de type intervalle (f, g) . Des conditions sur la compatibilité des systèmes $((\max, +)$, $(\min, \max, +)$, $((\min, \max, +)$, $(\min, +)$) et $((\max, +)$, $(\min, +)$) sont alors développées. Établir ces conditions a nécessité une exploitation des résultats relatifs à la théorie de la résiduation ainsi que l'application de la théorie spectrale des fonctions $(\min, \max, +)$. Les résultats que nous avons trouvés dans le cas où f et g sont respectivement de type $(\max, +)$ et $(\min, +)$, ont été appliqués dans le cadre de l'analyse de la vivacité des graphes d'événements p-temporels.

Nous avons aussi abordé le problème de re-formulation des graphes d'événements p-temporels. Il s'est avéré que la méthode classique, qui permet de simplifier le modèle des GET, ne peut pas être appliquée sans précautions dans le cas des graphes d'événements p-temporels. En effet, à travers un exemple, nous avons montré que le problème initial de la non-vivacité du graphe d'événements p-temporel, est disparu après simplification par cette démarche. Nous avons donc introduit une condition de vivacité "statique" qui permet de vérifier si l'on peut re-formuler le graphe d'événements p-temporel sous la forme $\mathcal{X} \geq \gamma^1 \mathbf{A}^- \mathcal{X} \oplus \gamma^{-1} \mathbf{A}^+ \mathcal{X}$.

Ensuite, nous avons proposé deux approches pour vérifier et analyser la vivacité des graphes d'événements p-temporels. La première méthode utilisant le vecteur spectral et le graphe de calcul. Sans outils informatiques opérationnels à l'heure actuelle, cette approche présente des difficultés liées au repérage manuel des circuits dans le graphe de calcul : même pour des graphes d'événements p-temporels simples, le graphe de calcul présente une structure complexe faisant introduire des rétroactions ; d'autre part, sa taille est infinie. C'est pour cela que nous avons cherché un autre moyen permettant la prise en compte de tous les circuits dans le graphe de calcul, quelque soit l'horizon de calcul. Une approche originale basée sur le calcul matriciel est ainsi proposée. Cette approche part du graphe de calcul en l'interprétant sous forme de séries de matrices. Cette démarche présente les avantages suivants par rapport à celle basée directement sur le calcul du vecteur spectral : le graphe de calcul est représenté par des matrices de tailles réduites ; de plus, le calcul est itératif et exploite les résultats des calculs précédents.

Chapitre 4

Estimation d'état optimale des graphes d'événements temporisés et temporels. Application à la détection de défaillances

4.1 Introduction

Dans ce chapitre, nous considérons le problème de l'estimation d'état. Une première approche serait d'utiliser la connaissance du modèle et des conditions initiales afin de caractériser le vecteur d'état à tout moment donné et d'avoir sa trajectoire d'état. Ceci peut être réalisé par une itération de l'équation d'état pour les graphes d'événements temporisés. Mais les perturbations génèrent rapidement une mauvaise détermination de ce vecteur ou trajectoire d'état, et ainsi, empêchent le contrôle du système (processus). Cette approche se complique encore plus avec les modèles d'intervalles où les trajectoires ne peuvent être déduites simplement.

Une approche plus aboutie est de reconstruire l'état x du système à partir des données connues de l'entrée u et la sortie y (voir figure 4.1). Le modèle (en fonctionnement normal ou nominal) est supposé connu et invariant à partir d'un numéro d'événement de début, noté k_s . L'équation d'état, ou une forme analogue, est donc utilisable pour $k \geq k_s$. En général, l'état estimé est différent de l'état réel que l'on pourrait obtenir par des mesures effectuées directement sur le système en raison de l'imperfection de l'estimation.

Ce chapitre est structuré de la manière suivante :

Dans un premier temps, un rappel sur l'estimation et la détection de défaillances, est donné

en considérant les systèmes continus. Les principes donnés apporteront une vision générale du problème posé. Ensuite, on propose un estimateur optimal pour les graphes d'événements à flux temporels et on étudie l'existence d'une estimation. On donne ensuite deux algorithmes de calcul de la plus grande estimation spécifiques aux graphes d'événements temporisés et à flux temporels. On montre enfin que l'approche proposée peut être appliquée au problème de la détection de défaillances. Ce chapitre se termine par l'analyse des résultats de simulation obtenus sur différents graphes d'événements.

Les résultats énoncés dans ce chapitre regroupent les contributions dans [DDA02] [DDA03a] et [DDA05c]

4.2 Estimation et détection de défaillances pour les systèmes continus

4.2.1 Rappels sur les observateurs

Un estimateur ou reconstituteur d'état est un système (voir figure 4.1) ayant comme entrées les entrées et les sorties du processus réel, et dont la sortie est une estimation de l'état de ce processus.

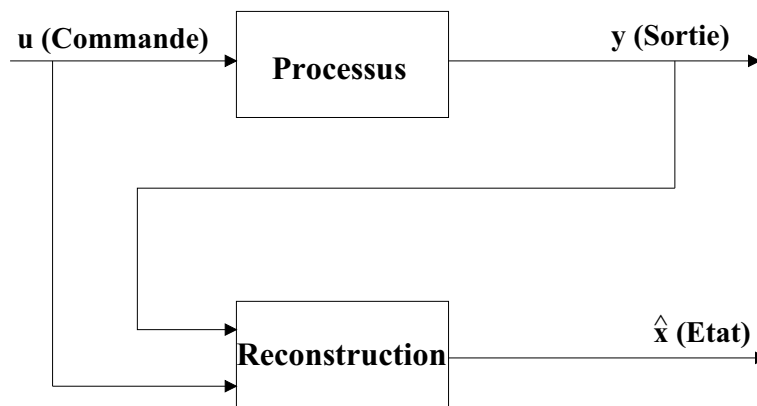


Figure 4.1 : Reconstruction d'état à partir de l'entrée et de la sortie

La réalisation de l'estimation dépend de la nature du modèle du système réel étudié : continu ou discret, déterministe ou stochastique. Notons que, dans le cas où le modèle du processus est déterministe, l'estimateur sera appelé observateur tandis que dans le cas où le modèle est stochastique, le reconstituteur d'état sera appelé filtre. Les observateurs sont des outils bien connus des automaticiens à des fins de commande en boucle fermée dans les processus continus. Nous nous limiterons par la suite au cas déterministe qui est plus proche de notre problème initial.

Rappelons la philosophie de l'estimation dans les systèmes continus. Le problème de l'observabilité prend origine dans le fait que l'on considère souvent des systèmes possédant n variables d'état et p sorties, dans lesquels le nombre des sorties est inférieur à celui des variables d'état ($p < n$). La question se pose sur la possibilité de reconstruire l'état à un instant, à partir de l'observation de la sortie pendant un temps fini.

Dans le problème de l'estimation, les seules grandeurs accessibles du système sont les variables d'entrée et de sortie. A l'opposé, la trajectoire de l'état est l'inconnue. En cohérence avec cette dernière donnée, il est important de rappeler qu'en estimation, les conditions initiales de l'évolution du système dynamique sont inconnues. Tout le problème est alors de s'affranchir de ces conditions initiales par une résolution adaptée au modèle défini et naturellement à l'algèbre utilisée.

Observateurs en continu

Le modèle correspond aux classiques équations d'état et de mesure, définies dans l'algèbre habituelle :

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) \\ x(t=0) = x_0 \end{cases} \quad (4.1)$$

La résolution du problème par Luenberger l'a conduit à l'observateur suivant, où la décroissance de fonctions exponentielles se traduira par la convergence asymptotique à zéro de l'erreur d'observation $\tilde{x} = x - \hat{x}$, avec comme point de départ $\tilde{x} = x_0 - \hat{x}_0$. La variable \hat{x}_0 est une estimation de l'état initial.

$$\begin{cases} \dot{\hat{x}}(t) = A\hat{x}(t) + Bu(t) + K[y(t) - C\hat{x}(t)], & \hat{x}(0) = \hat{x}_0 \\ \hat{y}(t) = C\hat{x}(t) \end{cases} \quad (4.2)$$

L'estimation n'est alors correcte qu'après un certain temps de convergence, fixé par la dynamique de l'observateur. Ce sera le cas si la matrice $\tilde{A} = A - K.C$ est de Hurwitz, c'est à dire que les valeurs propres de \tilde{A} sont à partie réelle négative.

Observateurs en discret

Les équations du modèle sont les classiques équations d'état et de mesure, en discret, définies dans l'algèbre habituelle :

$$\begin{cases} x(k+1) = Ax(k) + Bu(k) \\ y(k) = Cx(k) \\ x(0) = x_0 \end{cases} \quad (4.3)$$

La résolution du problème se traduit par l'observateur défini par les équations suivantes.

$$\hat{x}(k+1) = (A - KC)\hat{x}(k) + Bu(k) + Ky(k), \quad \hat{x}(0) = \hat{x}_0 \quad (4.4)$$

$$\hat{y}(k) = C\hat{x}(k) \quad (4.5)$$

La matrice K est la matrice de contre-réaction. Afin de permettre la convergence de l'observateur et de s'affranchir des conditions initiales, les valeurs propres de $(A - KC)$ doivent être dans le cercle unité.

Remarque : Dans [Har04], un observateur dans l'algèbre $(\text{Max}, +)$ a été présenté. Le point de départ est la structure de l'observateur de Luenberger mais avec les transformations suivantes : l'addition $+$ du modèle et la soustraction $-$ du comparateur, sont remplacées par le maximum noté \oplus ; la multiplication \times est remplacée par l'addition noté \otimes . Permettant de rester dans l'algèbre $(\text{max}, +)$, ces transformations permettent d'être cohérent avec les structures $(\text{max}, +)$ développées en boucle fermée par ailleurs, et donc une insertion de l'observateur dans une structure de commande analogue à celle du retour d'état. Une amélioration des performances est constatée sur un exemple par rapport à une structure dite "retour de sortie". Comme il est imposé que le modèle de l'observateur soit exprimé par un graphe d'événements temporisé, l'approche est non-optimale comme l'illustre l'exemple élémentaire du voyageur dans le paragraphe 4.3. Remarquons que la condition initiale n'est pas exprimée dans le modèle initial et donc qu'une perspective est son intégration dans une approche plus générale.

◇

4.2.2 Rappels sur la détection de défaillances

De manière classique, les principales étapes de la surveillance sont la détection, l'isolation, le diagnostic et la reconfiguration. De nombreuses méthodes de détection de défaillances ont été développées depuis les années 70 [Wil76]. Dans la classe des méthodes qui n'utilisent pas le modèle du processus, on peut citer une méthode répandue en milieu industriel qui consiste, via un observateur (opérateur ou dispositif automatique), à comparer en ligne les grandeurs à des seuils prédéterminés. Ces grandeurs peuvent être des mesures des capteurs ou des mesures prédéfinies. Dans tous les cas, cette méthode présente des limitations. En effet, des fausses alarmes sont provoquées si les variables retournent à un niveau normal en l'absence de réaction d'un opérateur. Il existe une autre classe de méthodes de détection de défaillances basées sur un modèle linéaire du processus où la modélisation du processus physique est un préalable à ces méthodes. Parmi ces méthodes, on distingue en général, l'approche basée sur les observateurs (ou filtres) de celle basée sur l'espace de parité [Ger88]. Les deux méthodes sont basées sur la généralisation des résidus (des indicateurs de défauts) pour la détection de défaillances. Pour les deux approches, l'estimation d'état est effectuée de manière explicite (les observateurs) ou non (l'espace de parité). Citons également les approches d'identification qui consistent à identifier en ligne les divers paramètres du système et à comparer ces estimations aux valeurs nominales

des paramètres.

Observateurs et génération de résidus

Utilisant les observateurs présentés ci-dessus, le principe général de cette méthode consiste à comparer les variables de sortie estimées avec les mêmes variables de sortie mesurées. On utilisera alors ici l'erreur d'estimation sur les sorties comme indicateur de défauts ou résidu (voir figure 4.2).

$$\varepsilon(k) = y(k) - \hat{y}(k) \quad (4.6)$$

avec $y(k)$ et $\hat{y}(k)$ représentent respectivement la sortie mesurée et la sortie estimée.

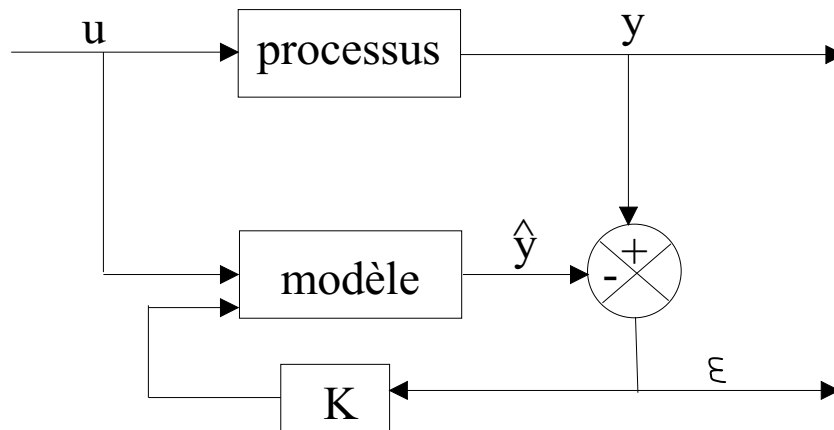


Figure 4.2 : Génération des résidus à l'aide d'observateurs

Espace de parité et génération de résidus

Le principe de l'approche "espace de parité" est de tester la cohérence d'un modèle avec le processus physique. La réécriture des équations du modèle, de manière à obtenir des relations particulières, appelées Relations de Redondance Analytique (RRA), est le principe de cette approche. Ces relations (RRA) mettent en rapport directement des grandeurs connues et accessibles. Les résidus sont obtenus en substituant dans ces relations (RRA) les variables connues par leurs valeurs réelles, prélevées sur le système en fonctionnement. Une défaillance est détectée lorsque le résidu sort d'un domaine prédéfini.

Nous considérons le modèle d'état linéaire donné par le système 4.3. Nous écrivons ces équations sur une fenêtre d'observation $[k, k + s]$, nous obtenons alors la forme suivante :

$$y(k, k + s) = O_s x(k) + COM_s(B)u(k, k + s) \quad (4.7)$$

avec $y(k, k+s) = \begin{pmatrix} y(k) \\ y(k+1) \\ \vdots \\ y(k+s) \end{pmatrix}$, $u(k, k+s) = \begin{pmatrix} u(k) \\ u(k+1) \\ \vdots \\ u(k+s) \end{pmatrix}$, O_s est la matrice d'observabilité

d'ordre s dont la forme est $O_s = [C^t(CA)^t \dots (CA^s)^t]^t$ et $COM_s(B)$ est la matrice de contrôle (de commande) d'ordre s

$$COM_s(B) = \begin{pmatrix} 0 & \dots & 0 \\ CB & 0 & \dots & \vdots \\ \vdots & \ddots & \ddots & \\ CA^{s-1}B & \dots & CB & 0 \end{pmatrix}.$$

L'espace de parité d'ordre s noté P_s est l'espace engendré par les vecteurs lignes de la matrice P définie par :

$$P \times O_s = 0 \tag{4.8}$$

Après avoir multiplié à gauche et à droite l'équation 4.7, on obtient

$$0 = P.y(k, k+s) - P.COM_s(B).u(k, k+s) \tag{4.9}$$

Comme l'état $x(k)$ est éliminé, ces égalités représentent les RRA. En tenant compte des erreurs de modélisation et des bruits, le vecteur des résidus prendra la forme :

$$r(k+s) = P.y(k, k+s) - P.COM_s(B).u(k, k+s) \tag{4.10}$$

La figure 4.3 décrit schématiquement la méthode de l'espace de parité.

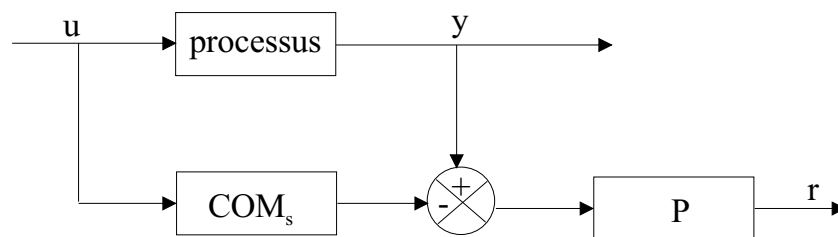


Figure 4.3 : Principe de la méthode espace de parité

Remarque : Le calcul des observateurs et des Relations de Redondance Analytique dépendent de la notion d'observabilité. On dit qu'un système est complètement observable sur un horizon fini $[k_s, k_f]$ si de l'observation de sa sortie y sur cet horizon on peut déduire l'état du système $x(k)$ pour $k \in [k_s, k_f]$. En revanche, si l'observation de la sortie sur cet horizon nous donne qu'une partie de l'état du système, alors ce dernier est inobservable. De manière classique, pour dire qu'un système est observable, il suffit de s'assurer que le rang du sous espace généré par $[C^t, A^t C^t, \dots, (A^{n-1})^t C^t]$ soit égal à n . Notons que

dans les travaux de [Gaz97] et [PW97] des approches concernant les conditions d'observabilité pour les système $(\max, +)$ ont été développées.

◇

Après avoir rappelé et présenté ces quelques méthodes d'estimation et de détection de défaillances pour les systèmes continus, nous proposons maintenant d'introduire une approche d'estimation d'état dans les graphes d'événements temporels. Préalablement, nous donnons la philosophie de cet observateur, dans l'algèbre $(\max, +)$ et les algèbres dites "topicales ou exotiques".

4.3 Estimation en fonctionnement nominal dans les graphes d'événements

Pour illustrer simplement le principe de l'estimation d'état dans les graphes d'événements, prenons l'exemple élémentaire d'un voyageur. En connaissant sa date d'arrivée x_{Ar} et la durée du voyage T , on peut estimer sa date de départ par une simple soustraction $x_{Ar} - T$. Cet observateur élémentaire n'est donc pas un graphe d'événements (ou un réseau de Petri) mais est analogue aux équations "backward" utilisées en commande optimale.

Sous un point de vue plus mathématique, les observateurs développés par Luenberger comme le filtre de Kalman-Bucy, sont le résultat de raisonnements algébriques spécifiques aux modèles considérés et plus généralement à l'algèbre classique. Le résultat est par exemple une structure en boucle fermée pour l'observateur de Luenberger en discrétisé. L'objectif est également dans ce chapitre, de développer un observateur. Cependant, l'algèbre considérée ici, est fondamentalement différente comme le montre le théorème suivant, ce qui n'est pas sans conséquences.

Théorème 4.1. [BCOQ92] *Tout groupe idempotent est réduit à l'élément zéro.*

En d'autres termes, la propriété d'idempotence de l'opération maximum notée \oplus , que l'on trouve dans l'algèbre des dioïdes est opposée à la propriété de symétrie de l'addition habituelle $+$ qui est largement exploitée en automatique classique. C'est cette opposition qui permet de clairement distinguer les anneaux des dioïdes. Les auteurs Gondran et Minoux [GM01] présentent un arbre montrant les différentes relations entre les différents concepts.

La conséquence est que, les résolutions dans les algèbres des dioïdes sont fondamentalement différentes de celles de l'algèbre habituelle. L'étude des différents moyens de résolution montre un caractère original devant traiter des espaces de solutions également particuliers du type non convexes. On peut citer :

- l'algorithme MD de Mc Millan et Dill [MD92]
- l'algorithme de E. Walkup et Borriello [Wal95] et [WB98] avec l'amélioration de Y. Cheng et

D-Z. Zheng [CZ02] et [CZ05]

- l'algorithme alternée de R.A. Cuninghame-Green et P. Butkovic [CGB03]

Ces algorithmes ne présentent aucune analogie évidente avec les résolutions plus classiques comme la programmation linéaire. Une exception est la symétrisation développée par S. Gaubert où une certaine analogie existe avec la résolution de Kramer. Cependant, cette approche permet plus de gérer des combinatoires, qu'elle n'est un outil de résolution. Elle permet en particulier, une formulation élégante du théorème de Cayley-Hamilton [BCOQ92].

Le point de départ de la démarche qui suit, sera donc une méthode de résolution relative au domaine. Nous choisissons par la suite l'algorithme général MD en raison de sa proximité avec la résolution par l'étoile de Kleene. Sa mise en oeuvre est simple et demande une écriture du problème sous forme de point fixe. Comme pour les observateurs développés pour les systèmes continus, la structure de calcul de l'estimateur que nous allons introduire dans ce chapitre sera la conséquence de la résolution, et ne sera pas fixée préalablement.

Exploitant la théorie relative au vecteur spectral et au point fixe, nous présentons maintenant l'analyse de l'existence d'une estimation et le calcul de la solution optimale.

4.3.1 Synthèse d'un estimateur

L'objectif dans cette partie est de trouver la plus petite valeur de la borne supérieure de $x(k)$ sachant que les valeurs de l'entrée $u(k)$ et de la sortie $y(k)$ sont connues pour k allant de k_s à k_f . Les événements k_s et k_f correspondent aux numéros de début et de fin de l'horizon. Le modèle est supposé connu sur ce même horizon d'observation.

Remarque : Dans la suite de cette section, les résultats relatifs aux graphes d'événements à flux temporels que l'on va présenter sont valables aussi pour les différents graphes d'événements temporels (p-temporels, temporisés, ...).

◇

Il a été montré auparavant (chapitre 2) que l'évolution d'un système représenté par un graphe d'événements (temporisé ou temporel) est décrite en général par un modèle d'intervalle dont la forme est :

$$f^-(x(\gamma), u(\gamma)) \leq x(\gamma) \leq f^+(x(\gamma), u(\gamma)) \quad (4.11)$$

Les fonctions f^- et f^+ sont de type (min, max, +). On considère ici que la sortie obéit au système suivant :

$$y(\gamma) = C \otimes x(\gamma) \quad (4.12)$$

Comme on a choisi de travailler avec seulement deux sémantiques différentes le Et-Pur et le Et-Faible, la fonction f^- donnée par l'équation 4.11 sera alors de type (max, +) seulement (voir

paragraphe 2.4.4.3). Nous avons alors le système suivant modélisant un graphe d'événements à flux temporels avec les sémantiques Et-Pur et Et-Faible :

$$\begin{cases} A^- \otimes x(\gamma) \oplus B^- u(\gamma) \leq x(\gamma) \leq \bigwedge_{i=1}^{j_i} A_i^+ \otimes x(\gamma) \oplus B_i^+ \otimes u(\gamma) \\ y(\gamma) = C \otimes x(\gamma) \end{cases} \quad (4.13)$$

On pourra donc exploiter le fait que la fonction f^- est résiduable. Ceci nous amène à introduire la proposition qui suit et qui donne une formulation du problème de l'estimation sous une forme du type point fixe.

Proposition 4.2. *Pour un graphe d'événements à flux temporel avec les sémantiques Et-Pur et Et-Faible, le problème d'estimation du plus grand $x(\gamma)$ peut être re-formulé comme suit : en allant de k_s à k_f , chercher la plus grande solution de l'inéquation suivante :*

$$x(\gamma) \leq h(x(\gamma)) \quad (4.14)$$

avec

$$h(x(\gamma)) = \left(\gamma^{-1}x(\gamma) \wedge (A^- \setminus x(\gamma)) \wedge (C \setminus y(\gamma)) \wedge \bigwedge_{i=1}^{j_i} (A_i^+ \otimes x(\gamma) \oplus B_i^+ \otimes u(\gamma)) \right)$$

tout en respectant les contraintes $u(\gamma) \leq B^- \setminus x(\gamma)$ et $y(\gamma) \leq Cx(\gamma)$

Preuve : Reprenons le système 4.13. Comme la multiplication matricielle \otimes est résiduable (voir le paragraphe 1.4.2.1), nous aboutissons alors au système suivant :

$$\begin{cases} x(\gamma) \leq (A^- \setminus x(\gamma)) \wedge \bigwedge_{i=1}^{j_i} A_i^+ \otimes x(\gamma) \oplus B_i^+ \otimes u(\gamma) \\ u(\gamma) \leq B^- \setminus x(\gamma) \\ y(\gamma) \leq C \otimes x(\gamma) \\ C \otimes x(\gamma) \leq y(\gamma) \end{cases}$$

D'où le résultat : $x(\gamma) \leq A^- \setminus x(\gamma) \wedge C \setminus y(\gamma) \wedge \bigwedge_{i=1}^{j_i} A_i^+ \otimes x(\gamma) \oplus B_i^+ \otimes u(\gamma)$

avec $u(\gamma) \leq B^- \setminus x(\gamma)$ et $y(\gamma) \leq C \otimes x(\gamma)$

Nous imposons d'autre part la non-décroissance de la trajectoire de l'état estimé $x(\gamma) \leq \gamma^{-1}x(\gamma)$

□

Notons d'une part que l'équation 4.14 est de type (min, max, +). D'autre part, l'équation 4.14 peut se décomposer en deux termes : le premier $\gamma^{-1}x \wedge (A^- \setminus x(\gamma)) \wedge (C \setminus y(\gamma))$ présente une partie backward ; le deuxième terme $\bigwedge_{i=1}^{j_i} (A_i^+ \otimes x(\gamma) \oplus B_i^+ \otimes u(\gamma))$ correspond à la partie forward si les exposants en γ dans les matrices A_i^+ and B_i^+ sont positifs. La concomitance de ces deux

phénomènes rend le problème d'estimation particulièrement complexe. Nous devons de manière générale, résoudre un problème du type point fixe donné par :

$$x \leq f(x) \quad (4.15)$$

avec la fonction f est de type (min, max, +) qui est définie dans la grammaire suivante :

$$f = b, x_1, x_2, \dots, x_n \mid f \otimes a \mid f \wedge f \mid f \oplus f \quad (4.16)$$

les scalaires a et b sont deux réels. La forme 4.16 présente une forme différente par rapport à la forme d'une fonction homogène donnée par 1.30.

4.3.2 Existence d'une estimation optimale

La fonction h donnée formellement par l'équation 4.14 est une fonction non homogène puisqu'elle peut se mettre sous la forme 4.16. Par conséquent, l'utilisation du vecteur spectral pour vérifier l'existence d'une solution du système 4.15 est non applicable. Ceci nous conduit à proposer une forme plus adéquate qui permettra l'application de la théorie spectrale.

La théorie spectrale s'applique aux cas des fonctions homogènes. Comme la forme 4.16 ne présente pas cette dernière propriété, nous allons procéder à une relaxation. En d'autres termes, faire une homogénéisation de la fonction f donnée par 4.15 en introduisant une variable x_0 que l'on va associer à b tel que b soit substitué par $b \otimes x_0$ dans la forme 4.16. Ainsi, si le problème initial est relatif au système $x \leq f(x)$, le but est alors de trouver la solution optimale $y = (x_0, x_1, \dots, x_n)^t$ (si y existe) telle que $x_0 = 0$ et $x_i \leq f_i(x_0, x_1, \dots, x_n)^t$ pour tout $i \neq 0$ avec f_i une fonction (min, max, +) de type $(n + 1, 1)$.

Si nous introduisons l'inéquation triviale suivante $x_0 \leq x_0$, on obtient alors une nouvelle formulation qui consiste à chercher le plus grand $y = (x_0, x_1, \dots, x_n)^t$ tel que $x_0 \leq x_0$ et $x_i \leq f_i(x_0, x_1, \dots, x_n)^t$ pour tout $i \neq 0$ avec $x_0 = 0$. En d'autres termes, le nouveau problème du type point fixe à résoudre est donné comme suit :

$$y \leq F(y) \quad (4.17)$$

avec F une fonction (min, max, +) de type $(n + 1, n + 1)$. Le problème donné par 4.15 est ramené par cette opération dite de relaxation, à une nouvelle formulation donnée par 4.17.

Remarque : Notons que dans la suite de ce chapitre, chaque fonction non homogène f écrite en lettres minuscules, correspond à une fonction homogène F notée en lettres majuscules.

◇

La proposition suivante nous permettra de transposer les résultats relatifs aux fonctions non homogènes aux cas de fonctions homogènes.

Proposition 4.3. Soient les deux fonctions f et F respectant respectivement les deux systèmes 4.15 et 4.17. Alors on a :

$$\{x \text{ tel que } x \leq f(x)\} = \{x \text{ tel que } x_i = (-y_0) \otimes y_i \text{ pour } y \text{ vérifiant } y \leq F(y)\}$$

Preuve : Soit x tel que $x \leq f(x)$. On peut vérifier facilement que le vecteur $y = (0, x_1, \dots, x_n)^t$ est une solution pour $x_0 \leq x_0$, $x_i \leq f_i((x_0, x_1, \dots, x_n)^t)$ pour tout $i \neq 0$ avec $x_0 = 0$ ou $y \leq F(y)$ avec $y_0 = 0$. Inversement, soit y une solution de $y \leq F(y)$. La propriété d'homogénéité implique que $\lambda \otimes y$ est aussi une solution. En particulier, on peut prendre $\lambda = (-y_0)$ et la solution $x_i = (-y_0) \otimes y_i$ vérifie $x_0 = 0$ et $x_i \leq f_i(x_0, x_1, \dots, x_n)^t$ pour tout $i \neq 0$

□

Par conséquent, la résolution et l'analyse de l'existence d'une solution, peuvent être appliquées au système homogène. On transpose ensuite la solution obtenue du système initial par multiplication \otimes de chaque composante du vecteur x par $-y_0$. Autrement dit, la nouvelle origine des temps y_0 peut être décalée ou choisie arbitrairement.

Nous supposons qu'une fonction non homogène de type (min, max, +) admet la forme suivante :

$$f_i(x) = \bigwedge_{v \in V(i)} A_{iv}x \quad (4.18)$$

pour tout i de 1 à n et $f_0(x) = 0$ avec $x = (x_0, x_1, \dots, x_n)^t$. $\{V(1), \dots, V(n)\}$ est un ensemble fini d'indices et A_{iv} est un vecteur ligne dont les coefficients appartiennent à \mathbb{R}_{\max} . On dit qu'une politique est une application $\pi : \{0, \dots, n\} \rightarrow \bigcup_{0 \leq i \leq n} V(i)$, tel que $\pi(i) \in V(i)$, pour tout $0 \leq i \leq n$. La politique matricielle correspondante $A[\pi]$ est définie par : $A[\pi]_i = A_{i\pi(i)}$. Chaque fonction h appartenant à $rec(S)$ définit une politique π_h . Alors, $h : x \in \mathbb{R}^n \rightarrow h(x) = A[\pi_h] \otimes x$.

Définition 4.4. (Structure CPSN)

Soient la fonction $h \in rec(S)$ et sa politique correspondante π_h . Cette structure est dite CPSN (Circuits de Poids Strictement Négatif), si dans le graphe correspondant à $A[\pi_h]$, tous les circuits ont un poids strictement négatif. La résolution du système a une unique solution donnée par la première colonne de $A^*[\pi_h]$.

Afin de considérer une fonction homogène, nous introduisons les expressions $V(0)$ et A_{00} , avec $V(0) = \{0\}$ et $A_{00} = (e, \dots, \varepsilon)$. Dans ce cas, on dit que la fonction h et sa politique correspondante ont une structure CPSN si cela est vrai dans le cas non-homogène.

Nous considérons maintenant des structures qui satisfont $\chi(\Phi_f(X_I)) = 0$. Ceci mène à introduire les deux propositions 4.5 et 4.6.

Proposition 4.5. Soit une fonction (min, max, +) $f(x) = \bigwedge_{g \in G} g(x)$ avec $g(x)$ une (max, +) application seulement. Toute solution d'une structure (max, +) avec $\chi(g) = 0$ vérifie $f(x) \leq x$.

Preuve : $f(x) = \bigwedge_{g \in G} g(x)$ avec $g(x)$ une application (max, +). Si $\chi(g) = 0$, alors il existe une solution notée α qui vérifie $\alpha = g(\alpha)$. Ainsi, $f(\alpha) \leq g(\alpha) = \alpha$. □

Nous considérons maintenant le cas particulier d'une structure CPSN.

Proposition 4.6. La solution d'une structure CPSN (max, +) $A(\pi)$ pour $x_0 = 0$ est une borne supérieure de l'espace de solutions de $x \leq f(x)$ et $x = f(x)$.

Preuve : Soit $f(x) = \bigwedge_{g \in G} g(x)$ avec $g(x)$ une application (max, +) seulement. La structure (max, +) avec $A(\pi)$ pour $x_0 = 0$ a une seule solution unique notée α . Elle est aussi la plus grande solution vérifiant $x \leq g_\pi(x)$ et $x = g_\pi(x)$. Comme toute solution doit vérifier $x \leq g_i(x)$, elle doit vérifier en particulier $x \leq g_\pi(x)$. Par conséquent, toute solution est inférieure ou égale à α . □

Nous introduisons maintenant la proposition 4.7 qui donne une nouvelle formulation au problème d'estimation. La première forme est énoncée dans la proposition 4.2.

Proposition 4.7. Il existe une solution finie qui satisfait le système 4.14 sur un horizon l si et seulement si, $\chi(\Phi_H(X_l)) \geq 0$ avec H une fonction homogène et de type (min, max, +), qui est définie formellement par :

$$\begin{pmatrix} x_0 \\ x(\gamma) \end{pmatrix} \leq H \begin{pmatrix} x_0 \\ x(\gamma) \end{pmatrix} \quad (4.19)$$

avec

$$H \begin{pmatrix} x_0 \\ x(\gamma) \end{pmatrix} = \begin{pmatrix} x_0 \wedge u(\gamma) \setminus (B^- \setminus x(\gamma)) \wedge y(\gamma) \setminus Cx(\gamma) \\ \gamma^{-1}x(\gamma) \wedge (A^- \setminus x(\gamma)) \wedge (C \setminus (y(\gamma) \otimes x_0)) \wedge \\ \bigwedge_{i=1}^{j_1} (A_i^+ \otimes x(\gamma) \oplus B_i^+ u(\gamma) \otimes x_0) \end{pmatrix}$$

Preuve : En introduisant x_0 , le terme $\gamma^{-1}x \wedge (A^- \setminus x(\gamma)) \wedge (C \setminus y(\gamma)) \wedge \bigwedge_{i=1}^{j_i} (A_i^+ \otimes x(\gamma) \oplus B_i^+ \otimes u(\gamma))$ du système 4.14 est substitué alors par :

$$(\gamma^{-1}x(\gamma)) \wedge (A^- \setminus x(\gamma)) \wedge (C \setminus (y(\gamma) \otimes x_0)) \wedge \bigwedge_{i=1}^{j_i} (A_i^+ \otimes x(\gamma) \oplus B_i^+ \otimes u(\gamma) \otimes x_0)$$

A partir des contraintes $u(\gamma) \leq B^- \setminus x(\gamma)$ et $y(\gamma) \leq Cx(\gamma)$, nous en déduisons $u(\gamma) \otimes x_0 \leq B^- \setminus x(\gamma)$ et $y(\gamma) \otimes x_0 \leq Cx(\gamma)$ qui sont équivalents respectivement à $x_0 \leq u(\gamma) \setminus (B^- \setminus x(\gamma))$ et $x_0 \leq y(\gamma) \setminus Cx(\gamma)$.

Par conséquent, le système défini par 4.14 avec les deux contraintes citées ci-dessus peuvent aussi être défini par le système 4.19 sous la condition $x_0 = 0$.

Comme H est une fonction homogène, le vecteur spectral $\chi(\Phi_H(X_l))$ peut être calculé, et ainsi on peut appliquer le théorème 1.56. Si le vecteur spectral satisfait les conditions requises d'existence d'une solution, le système 4.14 pourra avoir une solution finie. □

Remarque :

1. Notons que la multiplication à gauche d'un vecteur par un scalaire dans le chapitre 1, donne $y(\gamma) \otimes x_0$ et $u(\gamma) \otimes x_0$. Les deux écritures peuvent se mettre aussi sous la forme $x_0 \otimes y(\gamma)$ et $x_0 \otimes u(\gamma)$.
2. L'existence d'une solution pour le système 4.19 nécessite la possibilité de prendre $x_0 = 0$ en raison de l'homogénéité de H . ◇

L'application H peut se réécrire sous la forme :

$$H = H_1 \wedge H_2 \wedge \dots \quad (4.20)$$

avec H_1, H_2, \dots qui sont toutes des applications (max, +) seulement.

Soit alors l'application H_1 correspondant aux équations "backward" relatifs aux graphes d'événements temporisés. On a :

$$\begin{pmatrix} x_0 \\ x(\gamma) \end{pmatrix} \leq H_1 \begin{pmatrix} x_0 \\ x(\gamma) \end{pmatrix} \quad (4.21)$$

avec

$$H_1 \begin{pmatrix} x_0 \\ x(\gamma) \end{pmatrix} = \begin{pmatrix} x_0 \\ (A^- \setminus x(\gamma)) \wedge (C \setminus (y(\gamma) \otimes x_0)) \end{pmatrix} \quad (4.22)$$

Définition 4.8. Observabilité

Un graphe d'événements est structurellement observable si, à partir de chaque transition interne, il existe au moins un chemin qui mène à une transition de sortie.

L'observabilité structurelle donne une condition pour observer en effet dans une sortie dont l'origine vient d'au moins une transition interne. Cette notion va nous permettre d'introduire les propositions 4.9 et 4.10. Les résultats énoncés dans ces dernières considèrent un horizon infini.

Proposition 4.9. *Un graphe d'événements structurellement observable à flux temporel dont les sémantiques sont le Et-Pur et le Et-Faible vérifie $\chi(\Phi_{H_1}(X_{+\infty})) = 0$. La plus grande solution de la partie backward h_1 vérifie le système $h(x) \leq x$ et donne une borne supérieure finie pour l'ensemble des solutions de $x \leq h(x)$ et $x = h(x)$.*

Preuve : La fonction $h(x)$ est une max-représentation, d'où l'écriture $h(x) = h_1(x) \wedge h_2(x) \wedge h_3(x) \wedge \dots$ avec chaque application h_i qui est max-plus seulement. L'application homogène correspondante à $h(x)$ est donnée par $H(x) = H_1(x) \wedge H_2(x) \wedge H_3(x) \wedge \dots$.

Un circuit dont le poids est égal à zéro, est associé à x_0 qui lui, à son tour, est associé à chacune des valeurs connues, en particulier les sorties. Si nous assumons que le système est observable, alors, de chaque transition interne il existe au moins un chemin qui part vers une transition de sortie dans le graphe d'événements. Dans le graphe associé, les sens des chemins est opposé et pour chaque noeud qui correspond à une transition x_i pour tout événement k , il existe un chemin qui part de x_0 vers $x_i(k)$. La définition du vecteur spectral associé à H_1 montre que $\chi(\Phi_{H_1}(X_{+\infty})) = 0$. Par conséquent, il existe x_1 tel que $x_1 = H_1(x_1)$. Cette solution vérifie aussi $H(x_1) = H_1(x_1) \wedge H_2(x_1) \wedge H_3(x_1) \wedge \dots \leq H_1(x_1) = x_1$. En particulier, la plus grande solution de la partie "backward" $h_1(x) = x$ pour $x_0 = 0$ que l'on peut noter x_2 vérifie $h(x_2) \leq x_2$.

Comme l'ensemble des séries formelles en γ est un treillis complet et h_1 est une fonction monotone, alors le théorème 1.33 de Knaster et Tarski peut être appliqué. Ce dernier prouve que x_2 qui est la plus grande solution de $x = h_1(x)$ est aussi la plus grande solution vérifiant $x \leq h_1(x)$. Ce même résultat peut être prouvé en considérant l'ensemble $\{x \leq h_1(x)\}$ qui est un treillis complet (voir le théorème 1.35). De plus, comme l'inéquation $x \leq h(x)$ est équivalente à l'ensemble des inéquations $x \leq h_i(x)$, chaque variable x doit satisfaire chacune de ces inéquations et en particulier $x \leq h_1(x)$. Par conséquent, chaque solution x de $x \leq h(x)$ est inférieure ou égale à x_2 . Finalement, cet ensemble comporte l'ensemble des solutions vérifiant $x = h(x)$. □

Proposition 4.10. *Dans un graphe d'événements structurellement observable à flux temporel dont les sémantiques sont le Et-Pur et le Et-Faible, il existe au moins une solution finie satisfaisant l'égalité :*

$$\begin{pmatrix} x_0 \\ x(\gamma) \end{pmatrix} = H \begin{pmatrix} x_0 \\ x(\gamma) \end{pmatrix} \quad (4.23)$$

ou il n'existe pas de solution pour le système 4.14.

Preuve : A partir de la proposition 4.9, on peut déduire que le vecteur spectral du système complet est inférieur ou égal à zéro : $\chi(\Phi_{H_1}(X_{+\infty})) = 0$. L'association avec de nouveaux termes peut créer des circuits de poids strictement négatifs et dans ce cas, il n'existe pas de solution. □

4.4 Algorithmes de calcul du plus grand état

Nous rappelons que le principe de l'algorithme MD (voir paragraphe 1.5.2) est d'affecter des valeurs infinies à l'initialisation, et de procéder à une minimisation de ses valeurs en propageant les contraintes du système. La solution finale est obtenue après satisfaction de toutes les contraintes par les valeurs des variables. Les itérations sont de la forme :

$$x_{k+1} \leftarrow f(x_k) \wedge x_k \quad (4.24)$$

Remarque : Le nombre k dans le terme 4.24 représente le numéro de l'itération et ne correspond pas au k ème date de tir de l'état.

◇

Avec le même esprit que celui de l'algorithme donné ci-dessus (4.24), nous introduisons un algorithme spécifique aux graphes d'événements à flux temporels. Cet algorithme peut aussi être appliqué au cas des graphes d'événements temporisés, p-temporels, ... Nous supposons que chacune des places internes du GET contient un jeton. Par conséquent, le modèle algébrique correspondant nous permet d'obtenir des itérations définies sur un horizon court $[k, k + 1]$. Cela permettra une écriture simplifiée de l'algorithme.

4.4.1 Cas d'un graphe d'événements à flux temporels

Nous considérons ici que la notation $M(i, \cdot)$ représente la ligne i d'une matrice M , et $M(\cdot, j)$ représente la colonne j de la matrice M .

Nous spécifions le modèle relatif aux graphes d'événements à flux temporels comme suit :

$$f_i^+(x(\gamma), u(\gamma)) = \bigwedge_{l=1}^{j(i)} h_l(x(\gamma), u(\gamma))$$

avec $j(i)$ le nombre d'applications max-plus seulement h_l correspondant à chaque composante (transition) x_i . La fonction h_l a la forme :

$$h_l(x(\gamma), u(\gamma)) = A_l^+(i, \cdot) \otimes x(\gamma) \oplus B_l^+(i, \cdot) \otimes u(\gamma)$$

L'élément $A_l^+(i, j)$ égal $\varepsilon\gamma$ (respectivement $B_l^+(i, j) = \varepsilon\gamma$) s'il n'existe pas de places qui relie directement la transition interne x_j avec une autre transition interne x_i (respectivement, une transition d'entrée u_j avec une transition interne x_i). Aussi, on a $A_l^+(i, \cdot) = \gamma(T, T, \dots, T)$ et $B_l^+(i, \cdot) = \gamma(T, T, \dots, T)$ pour chaque composante x_i telle que $l > j(i)$.

Remarque :

- Les matrices A_l^+ et B_l^+ peuvent contenir à la fois des coefficients dont les valeurs temporelles sont : ε et T . La manipulation de ces matrices est possible puisque nous considérons que la structure algébrique du travail est un dioïde complet.
- Dans les algorithmes 1 et 2, les vecteurs $\mu(k)$ et $\lambda(k)$ représentent le vecteur d'état $x(k)$. Les variables k_s et k_f représentent respectivement les événements de début et de fin d'un horizon d'observation $[k_s, k_f]$.
- La monotonie est assurée si l'on ajoute l'identité à la matrice A^- .

◇

Algorithme 1 :Pas 0 : initialisation $\mu_i(k_f) \leftarrow T, \lambda_i(k_f) \leftarrow T$ Répéter jusqu'à : $\lambda_i(k) = \mu_i(k)$ pour $1 \leq i \leq n$ et $k_s \leq k \leq k_f$ Pas 1 : $\lambda_i(k) \leftarrow \mu_i(k) \wedge [A^-(., i) \setminus \lambda(k+1)] \wedge [C(., i) \setminus y(k)]$ de $k = k_f$ à k_s Pas 2 : $\mu_i(k_s) \leftarrow \lambda_i(k_s)$

$$\mu_i(k) \leftarrow \lambda_i(k) \wedge \bigwedge_{l=1}^{j(i)} [A_l^+(i, .) \otimes \mu(k-1) \oplus B_l^+(i, .) \otimes u(k)] \quad \text{de } k = k_{s+1} \text{ à } k_f$$

Adapté aux graphes d'événements, l'algorithme 1 ci-dessus converge vers le plus grand état en un nombre d'itérations fini, pour les graphes d'événements à flux temporels. Il est cependant difficile d'effectuer une analyse théorique du nombre d'itérations exactes comme pour beaucoup d'algorithmes dans ce domaine [CZ05]. On peut affirmer qu'il est en pratique, pseudo-polynomial comme l'algorithme général MD. La mise en oeuvre de l'algorithme est facile et a besoin seulement de la mémorisation des matrices du modèle, et de la trajectoire d'évaluation puisqu'il est basé sur l'itération d'une boucle "backward/forward".

4.4.2 Cas d'un graphe d'événements temporisé

Nous déduisons de l'algorithme 1 précédent un algorithme spécifique pour l'estimation du plus grand état dans les graphes d'événements temporisés. La convergence est très rapide dans le cas particulier des graphes d'événements temporisés, ce que nous allons démontrer ci-dessous.

Dans le but de garantir la monotonie des trajectoires, la condition de monotonie $x(k) \leq x(k+1)$ est directement introduite dans le modèle, ce qui permet de ne pas ajouter d'inéquations dans le système à résoudre :

$$x(\gamma) = A'\gamma \otimes x(\gamma) \oplus B \otimes u(\gamma) \text{ avec } A' = I_{\mathcal{D}} \oplus A$$

Algorithme 2 :

Pas 0 : initialisation $\lambda(k_f + 1) \leftarrow T$

Répéter jusqu'à : $\lambda_i(k) = \mu_i(k)$ pour $1 \leq i \leq n$ et $k_s \leq k \leq k_f$

Pas 1 : $\lambda(k) \leftarrow [A' \setminus \lambda(k+1)] \wedge [C \setminus y(k)]$ de $k = k_f$ à k_s

Pas 2 : $\mu(k_s) \leftarrow \lambda(k_s)$

$$\mu(k) \leftarrow \lambda(k) \wedge [A' \mu(k-1) \oplus Bu(k)] \quad \text{de } k = k_{s+1} \text{ à } k_f$$

Proposition 4.11. *Pour un graphe d'événements temporisé, l'algorithme 2 converge en une seule itération.*

Preuve : si l'on peut montrer que $\mu(k) \leq [A' \setminus \mu(k+1)] \wedge [C \setminus y(k)]$, on montrera également qu'une nouvelle itération de la boucle comme dans l'algorithme général, est inutile.

Si f est résiduable, alors il existe h tel que $f \circ h \leq I_{\mathcal{D}}$ et $h \circ f \geq Id$ d'après le théorème 1.26. Comme $A \otimes (A \setminus x) \leq I_{\mathcal{D}}$ et $A \setminus [A \otimes x] \geq I_{\mathcal{D}}$ alors, $A' \setminus [A' \otimes \mu(k)] \geq \mu(k)$ et $A' \setminus [A' \otimes \mu(k) \oplus B \otimes u(k+1)] \geq \mu(k)$ par isotonie de la résiduation à gauche de la multiplication \setminus .

De plus, le pas 2 implique que $\mu(k) \leq \lambda(k)$. Il en résulte, $\mu(k) \leq \lambda(k) \wedge A' \setminus [A' \otimes \mu(k) \oplus B \otimes u(k+1)]$. Il reste à montrer l'égalité suivante pour terminer la démonstration :

$$A' \setminus \mu(k+1) \wedge C \setminus y(k) = \lambda(k) \wedge A' \setminus [A' \mu(k) \oplus Bu(k+1)]$$

De simples substitutions, et la propriété de distributivité de la résiduation \setminus par rapport au \wedge sont appliquées par la suite.

$$\begin{aligned} A' \setminus \mu(k+1) \wedge C \setminus y(k) &= \\ A' \setminus [\lambda(k+1) \wedge [A' \mu(k) \oplus Bu(k+1)]] \wedge C \setminus y(k) &= \\ A' \setminus \lambda(k+1) \wedge C \setminus y(k) \wedge A' \setminus [A' \mu(k) \oplus Bu(k+1)] &= \\ \lambda(k) \wedge A' \setminus [A' \mu(k) \oplus Bu(k+1)] & \end{aligned}$$

Finalement, comme $\mu(k) \leq [A' \setminus \mu(k+1)] \wedge [C \setminus y(k)]$, l'algorithme n'aura pas besoin d'une nouvelle minimisation par le pas 1. D'autre part, la minimisation par $\mu(k)$ dans le pas 1 est inutile car l'algorithme démarre implicitement à T et n'effectue pas de nouvelle minimisation par le pas 1.

□

En conclusion, le terme "répéter jusqu'à" peut être supprimé ce qui donne l'algorithme suivant :

Algorithme 3 :

Pas 0 : initialisation $\lambda(k_f + 1) \leftarrow T$

Pas 1 : $\lambda(k) \leftarrow [A' \setminus \lambda(k + 1)] \wedge [C \setminus y(k)]$ de $k = k_f$ à k_s

Pas 2 : $\mu(k_s) \leftarrow \lambda(k_s)$

$\mu(k) \leftarrow \lambda(k) \wedge [A' \mu(k - 1) \oplus Bu(k)]$ de $k = k_{s+1}$ à k_f

La figure 4.4 explique le principe de l'estimateur qui permet déterminer le plus grand état. La partie gauche du graphe représente le terme "forward" de l'estimateur et la partie droite correspond au terme "backward". Cette figure montre bien l'existence d'une certaine symétrie entre la multiplication \otimes et la maximisation \oplus d'un côté et la résiduation \setminus et la minimisation \wedge de l'autre côté.

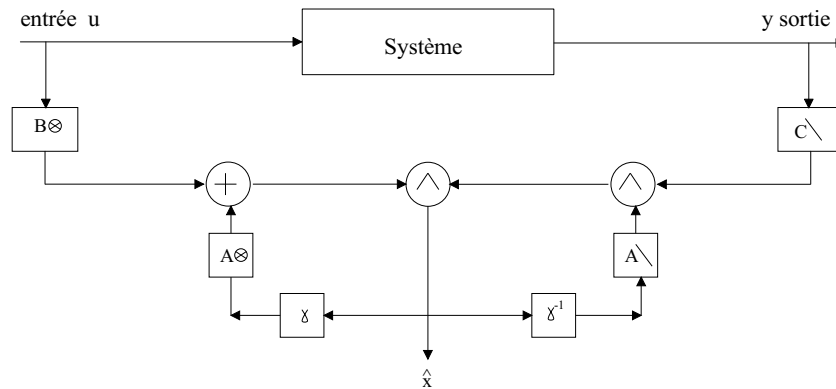


Figure 4.4 : Principe de l'estimation dans le cas d'un GET

4.5 Illustration

Le schéma de la figure 4.5 résume le principe de la démarche de détection de défaillance relative aux graphes d'événements temporisés et temporels, introduite dans ce chapitre.

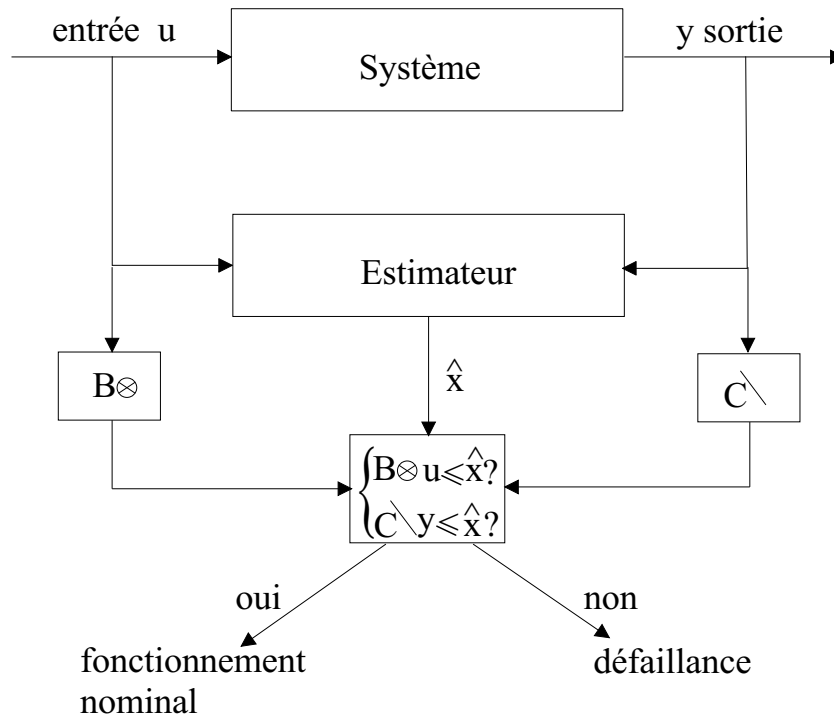


Figure 4.5 : Principe de détection de défaillance dans un graphe d'événements temporisé ou temporel

Les contraintes $u(\gamma) \leq B^-x(\gamma)$ et $y(\gamma) \leq Cx(\gamma)$ doivent être vérifiées par le comportement nominal du système. Ces deux contraintes vérifient la cohérence entre le modèle et les données connues. Elles permettent en particulier de vérifier que le graphe d'événements suit son modèle nominal.

Dans un but d'illustrer clairement l'approche d'estimation et de détection de défaillances présentée dans ce chapitre, des exemples simples de graphes d'événements temporisés et temporels sont considérés.

Tous les tests de simulation sont réalisés à l'aide du logiciel Scilab en utilisant la tool-box Max-plus, développée à l'INRIA. Les estimations sont faites sur des horizons de calcul arbitraires.

4.5.1 Cas d'un graphe d'événements temporisé

Le modèle nominal M_1 représentant le graphe de la figure 4.6 est donné par les matrices A_1 , B_1 et C_1 .

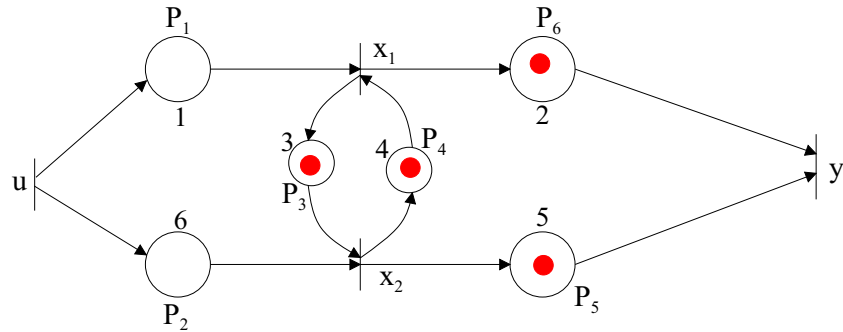


Figure 4.6 : Graphe d'événements temporisé

$$A_1 = \begin{pmatrix} \varepsilon & 4 \\ 3 & \varepsilon \end{pmatrix}, B_1 = \begin{pmatrix} 1 \\ 6 \end{pmatrix} \text{ et } C_1 = \begin{pmatrix} 2 & 5 \end{pmatrix}$$

Estimation

Dans le tableau suivant, x est obtenu par des itérations directes de l'équation d'état partant d'une entrée u connue et l'état $x(11) = \begin{pmatrix} 1 \\ 6 \end{pmatrix}$.

k	11	12	13	14	15	16	17
x	$\begin{pmatrix} 1 \\ 6 \end{pmatrix}$	$\begin{pmatrix} 10 \\ 9 \end{pmatrix}$	$\begin{pmatrix} 13 \\ 13 \end{pmatrix}$	$\begin{pmatrix} 17 \\ 16 \end{pmatrix}$	$\begin{pmatrix} 20 \\ 20 \end{pmatrix}$	$\begin{pmatrix} 24 \\ 23 \end{pmatrix}$	$\begin{pmatrix} 27 \\ 27 \end{pmatrix}$
u	0	3	6	9	12	16	19
y	11	14	18	21	25	28	32
\hat{x}	$\begin{pmatrix} 6 \\ 6 \end{pmatrix}$	$\begin{pmatrix} 10 \\ 9 \end{pmatrix}$	$\begin{pmatrix} 13 \\ 13 \end{pmatrix}$	$\begin{pmatrix} 17 \\ 16 \end{pmatrix}$	$\begin{pmatrix} 20 \\ 20 \end{pmatrix}$	$\begin{pmatrix} 24 \\ 23 \end{pmatrix}$	$\begin{pmatrix} 27 \\ 27 \end{pmatrix}$
\hat{y}	11	14	18	21	25	28	32
$B \otimes u$	$\begin{pmatrix} 1 \\ 6 \end{pmatrix}$	$\begin{pmatrix} 4 \\ 9 \end{pmatrix}$	$\begin{pmatrix} 7 \\ 12 \end{pmatrix}$	$\begin{pmatrix} 10 \\ 15 \end{pmatrix}$	$\begin{pmatrix} 13 \\ 18 \end{pmatrix}$	$\begin{pmatrix} 17 \\ 22 \end{pmatrix}$	$\begin{pmatrix} 20 \\ 25 \end{pmatrix}$

Le vecteur \hat{x} (Respectivement \hat{y}) représente l'état (respectivement, la sortie) estimé à partir des données connues u et y . Notons que le vecteur $x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$

Les contraintes $y \leq \hat{y} = C \otimes \hat{x}$ et $B \otimes u \leq \hat{x}$ sont vérifiées sur l'horizon de travail choisi [11, 17]. De plus, $\hat{y}(k) = y(k)$ et $\hat{x}(k) \geq x(k)$. En particulier, $\hat{x}(k) = x(k)$ pour $12 \leq k \leq 17$.

Défaillance

Nous considérons maintenant deux exemples de défaillances.

Défaillance 1 : augmentation de la durée d'une temporisation

Dans ce premier cas, la temporisation de la place p_4 est augmentée à 12 quand $k = 15$. Le nouveau modèle noté M_2 après cette défaillance est le suivant :

$$A_2 = \begin{pmatrix} \varepsilon & 12 \\ 3 & \varepsilon \end{pmatrix}, B_2 = B_1 \text{ et } C_2 = C_1$$

k	11	12	13	14	15	16	17
x	$\binom{1}{6}$	$\binom{10}{9}$	$\binom{13}{13}$	$\binom{17}{16}$	$\binom{28}{20}$	$\binom{32}{31}$	$\binom{43}{35}$
u	0	3	6	9	12	16	19
y	11	14	18	21	30	36	45
\hat{x}	$\binom{6}{6}$	$\binom{10}{9}$	$\binom{13}{13}$	$\binom{17}{16}$	$\binom{20}{20}$	$\binom{24}{23}$	$\binom{27}{27}$
\hat{y}	11	14	18	21	25	28	32
$B \otimes u$	$\binom{1}{6}$	$\binom{4}{9}$	$\binom{7}{12}$	$\binom{10}{15}$	$\binom{13}{18}$	$\binom{17}{22}$	$\binom{20}{25}$

A partir des données connues (entrée et sortie) sur l'horizon $k \in [11, 17]$, la procédure de détection de défaillance est donnée comme suit : la contrainte $y \leq \hat{y} = C \otimes \hat{x}$ n'est plus vérifiée sur cet horizon $[11, 17]$ ($y(k) \not\leq \hat{y}(k)$ pour $k = 15, 16$ et 17). Dans ce cas, la détection est réalisée à $k = 17$ ce qui correspond à une date minimale de 45 correspondant à la dernière donnée utilisée.

Défaillance 2 : diminution de la durée d'une temporisation

Un deuxième cas consiste à baisser la temporisation de la place p_4 à la valeur 1. On obtient alors un nouveau modèle nominal M_3 donné après cette défaillance par : $A_3 = \begin{pmatrix} \varepsilon & 1 \\ 3 & \varepsilon \end{pmatrix}$, $B_3 = B_1$ et $C_3 = C_1$

k	11	12	13	14	15	16	17
x	$\binom{1}{6}$	$\binom{10}{9}$	$\binom{13}{13}$	$\binom{17}{16}$	$\binom{17}{20}$	$\binom{21}{22}$	$\binom{23}{25}$
u	0	3	6	9	12	16	19
y	11	14	18	21	25	27	30
\hat{x}	$\binom{5}{4}$	$\binom{8}{8}$	$\binom{12}{11}$	$\binom{15}{15}$	$\binom{19}{18}$	$\binom{22}{22}$	$\binom{26}{25}$
\hat{y}	9	13	16	20	23	27	30
$B \otimes u$	$\binom{1}{6}$	$\binom{4}{9}$	$\binom{7}{12}$	$\binom{10}{15}$	$\binom{13}{18}$	$\binom{17}{22}$	$\binom{20}{25}$

De même, la procédure de détection de défaillances est la suivante : les contraintes $y \leq \hat{y} = C \otimes \hat{x}$ et $B \otimes u \leq \hat{x}$ ne sont plus vérifiées sur l'horizon $k \in [11, 17]$ ($y(k) \not\leq \hat{y}(k)$ pour k allant de 11 à 15 et $B \otimes u(k) \not\leq \hat{x}(k)$ pour k allant de 11 à 13). Dans ce cas, la détection de défaillances est réalisée après $t = 30$ unités de temps. Comme l'approche utilise une minimisation "backward", les contraintes avant $k = 15$ sont affectées même si la défaillance arrive à $k = 15$. L'introduction de nouvelles données pour $k = 18$ peut produire une minimisation pour l'estimation de x , $\hat{x}(k)$ pour $k \in [11, 18]$.

4.5.2 Cas d'un graphe d'événements à flux temporel

Nous traitons maintenant l'exemple d'un graphe d'événements à flux temporel présenté par la figure 4.7 et dont le modèle nominal est donné par les matrices A^- , B^- , A_1^+ , A_2^+ , B_1^+ , B_2^+ et C .

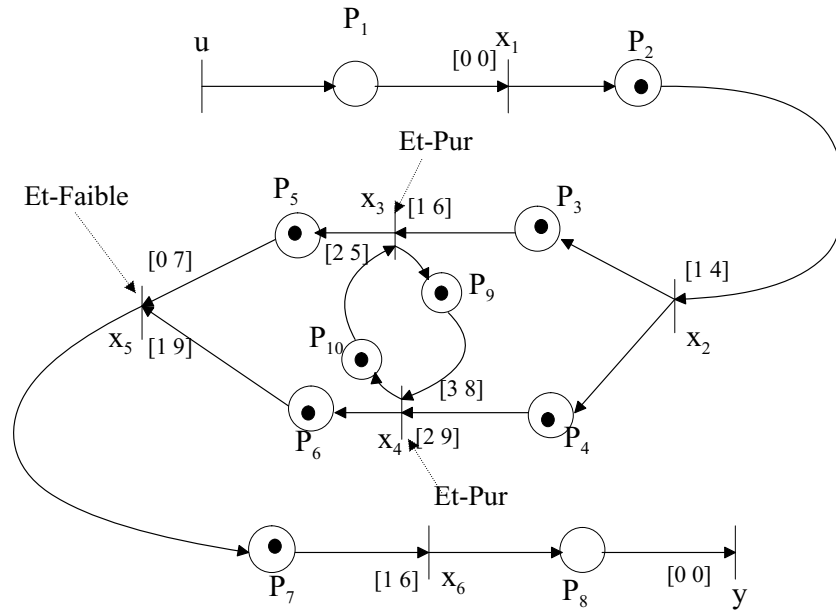


Figure 4.7 : Graphe d'événements à flux temporels avec les sémantiques ET et Et-Faible

$$A^- = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ 1 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & 1 & \varepsilon & 2 & \varepsilon & \varepsilon \\ \varepsilon & 2 & 3 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & e & 1 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & 1 & \varepsilon \end{pmatrix}, A_1^+ = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ 4 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & 6 & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & 9 & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 7 & 9 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & 6 & \varepsilon \end{pmatrix},$$

$$A_2^+ = \begin{pmatrix} T & T & T & T & T & T \\ T & T & T & T & T & T \\ \varepsilon & \varepsilon & \varepsilon & 5 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 8 & \varepsilon & \varepsilon & \varepsilon \\ T & T & T & T & T & T \\ T & T & T & T & T & T \end{pmatrix}, B^- = \begin{pmatrix} e \\ \varepsilon \\ \varepsilon \\ \varepsilon \\ \varepsilon \\ \varepsilon \end{pmatrix}, B_1^+ = \begin{pmatrix} e \\ \varepsilon \\ \varepsilon \\ \varepsilon \\ \varepsilon \\ \varepsilon \end{pmatrix}, B_2^+ = \begin{pmatrix} T \\ T \\ \varepsilon \\ \varepsilon \\ T \\ T \end{pmatrix} \text{ et}$$

$$C = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & e \end{pmatrix}.$$

Estimation

Dans un premier temps, nous calculons la trajectoire admissible du système sur l'horizon $[1, 10]$, et qui vérifie le modèle algébrique donné par le modèle 4.13. Les résultats de calcul sont regroupés dans le tableau suivant :

k	1	2	3	4	5	6	7	8	9	10
u	3	5	7	9	11	13	17	19	22	24
x_1	2	5	7	9	11	13	17	19	22	24
x_2	1	3	6	8	10	12	14	18	20	23
x_3	2	5	7	10	12	15	17	20	22	25
x_4	3	5	8	10	13	15	18	20	23	25
x_5	1	4	6	9	11	14	16	19	21	24
x_6	1	5	8	11	13	16	18	20	25	27
y	1	5	8	11	13	16	18	20	25	27

Dans un deuxième temps, l'observateur utilise l'entrée u et la sortie y pour estimer le plus grand état \hat{x} . Les résultats relatifs à ce dernier sont regroupés dans le tableau suivant :

k	1	2	3	4	5	6	7	8	9	10
\hat{x}_1	3	5	7	9	11	13	17	19	22	24
\hat{x}_2	6	9	9	11	13	15	17	21	23	26
\hat{x}_3	5	8	10	13	15	18	21	23	27	29
\hat{x}_4	6	8	11	13	16	18	21	25	27	32
\hat{x}_5	4	7	10	12	15	17	19	24	26	36
\hat{x}_6	1	5	8	11	13	16	18	20	25	27
$C \otimes \hat{x}$	1	5	8	11	13	16	18	20	25	27
$B^- \setminus \hat{x}$	3	5	7	9	11	13	17	19	22	24

Les contraintes $y \leq C \otimes \hat{x} = \hat{y}$ et $B^- \otimes u \leq \hat{x}$ sont vérifiées sur l'horizon $[1, 10]$.

De plus, $\hat{y}(k) = y(k)$ et $\hat{x} \geq x$. En particulier, $\hat{x}_6(k) = x_6(k)$ pour k allant de 1 à 10 et $\hat{x}_1(k) = x_1(k)$ pour $k \in [2, 10]$.

Défaillance

Nous considérons maintenant une défaillance. Les temporisations associées à l'arc partant de la place p_6 vers la transition x_5 , ne sont plus définies par l'intervalle $[1, 9]$ mais par l'intervalle $[7, 9]$. Nous obtenons alors un nouveau modèle M_2 à cause de cette défaillance, car l'un des coefficients de la matrice A^- a changé : $A^-(5, 4) = 7$. Le nouveau modèle M_2 est utilisé pour calculer la nouvelle trajectoire y . Ignorant le nouveau modèle M_2 , l'observateur utilise le modèle nominal M_1 et les nouvelles données pour estimer une nouvelle trajectoire \hat{x} . Les résultats du calcul sont donnés par le tableau suivant :

k	1	2	3	4	5	6	7	8	9	10
\hat{x}_1	3	5	7	9	11	12	16	18	21	23
\hat{x}_2	7	9	9	11	13	15	17	21	23	26
\hat{x}_3	6	8	11	13	16	18	21	23	27	29
\hat{x}_4	6	9	11	14	16	19	21	25	27	32
\hat{x}_5	7	10	18	20	23	25	28	30	34	36
\hat{x}_6	1	8	11	24	26	29	31	34	36	40
y	1	8	11	28	31	33	35	37	39	41
$C \otimes \hat{x}$	1	8	11	24	26	29	31	34	36	40
$B^- \setminus \hat{x}$	3	5	7	9	11	12	16	18	21	23

Sachant que les données relatives à l'entrée et à la sortie sont connues pour tout $k \in [1, 10]$, la détection de défaillances donne les résultats suivants : les contraintes $y \leq C \otimes \hat{x} = \hat{y}$ et $B^- \otimes u \leq \hat{x}$ ne sont pas vérifiées sur l'horizon $[4, 10]$ ($y(k)$ n'est pas inférieur ou égal à $\hat{y}(k)$ pour k allant de 4 à 10 et $B^- \otimes u(k)$ n'est pas aussi inférieur ou égal à $\hat{x}(k)$ pour k appartenant à $[6, 10]$).

4.6 Conclusion

Dans ce chapitre, nous avons proposé une approche pour estimer le plus grand état dans les graphes d'événements à flux temporels pour les sémantiques Et-Pur et Et-Faible. Il a été montré aussi qu'une seule forme du modèle d'intervalle suffit pour décrire les différents types de graphes d'événements, puisque les GET sont inclus dans les graphes d'événements à flux temporels. De même, les graphes d'événements p-temporels correspondent à la sémantique du Et-Pur. Le modèle algébrique obtenu a été transformé afin de re-formuler le problème de l'estimation sous la forme d'un problème du point fixe $x \leq f(x)$ avec f une fonction (\min , \max , $+$).

L'existence d'une solution exige la cohérence entre le modèle et les données exactes qui décrivent ce dernier. La nouvelle formulation proposée a introduit deux contraintes relatives aux données émanant de l'entrée et de la sortie. Ces contraintes ne sont pas utilisées pour l'estimation en ligne de la trajectoire d'état, mais elles permettent cependant de vérifier la cohérence des données. L'existence d'une solution finie peut aussi être analysée grâce à la théorie spectrale. Les résultats relatifs à la théorie du point fixe ont permis de chercher une borne supérieure de l'estimation d'état sous les conditions de l'observabilité structurelle.

L'estimateur optimal introduit dans ce chapitre a été exploité pour développer une méthode de détection de défaillances. Cette méthode est basée aussi sur la connaissance du modèle d'état. Les deux contraintes relatives aux données émanant de l'entrée et de la sortie sont donc utilisées comme moyen de test pour savoir si une défaillance a lieu ou non. La vérification de ces deux

contraintes signifie que le modèle suit son fonctionnement nominal. Dans le cas contraire, c'est l'occurrence d'une défaillance. Des exemples de GET et graphes d'événements à flux temporels ont été introduits pour bien illustrer l'application de l'estimateur dans le domaine de la détection de défaillances.

Chapitre 5

Commande optimale des graphes d'événements temporisés et temporels

5.1 Introduction

Dans ce chapitre, on s'intéresse au problème de la commande optimale des graphes d'événements temporisés et temporels. Le problème de la commande optimale se présente de la manière suivante : supposons que certains événements puissent être désignés comme contrôlables, ou en d'autres termes que le tir des transitions correspondantes puisse être retardé jusqu'à une certaine date notée $u(k)$, ceci par un système de supervision. Supposons que nous désirions ralentir le système de manière qu'aucun événement ne puisse arriver plus tard qu'une séquence de dates Z .

L'objectif de la commande optimale consiste alors à calculer la plus grande entrée et le co-état correspondant dans le graphe d'événements à partir d'une consigne connue $z(k)$ comme le montre la figure 5.1.

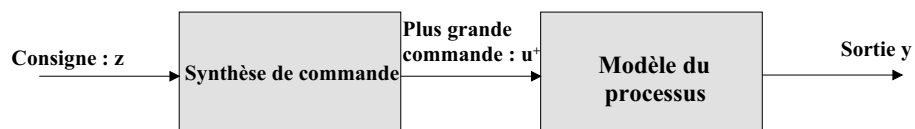


Figure 5.1 : Principe de la commande optimale

Ce chapitre suit la structure suivante :

La première partie traite des GET. Nous analysons la méthode classique dite "Backward".

Nous introduisons une approche s'appuyant sur une méthode de propagation de contraintes. Nous terminons cette partie par une comparaison des résultats entre cette approche et la méthode utilisant les équations "Backward".

La seconde partie concerne les graphes d'événements à flux temporels. La modélisation de ces derniers donne des formes complexes, et il faut chercher des moyens spécifiques pour traiter ce genre de modèles. On cherche ici une commande et un co-état admissibles, c'est à dire qui respectent les contraintes temporelles imposées par le modèle. Dans un premier temps, nous vérifierons l'existence d'une commande. Dans un deuxième temps, un algorithme spécifique aux graphes d'événements à flux temporels est introduit, donnant les valeurs optimales du co-état et de la commande.

Comme l'approche est analogue à celle de l'estimation, nous ne donnerons que les grandes lignes de l'approche afin de ne pas alourdir la présentation.

Ce chapitre est adapté des articles [DDA03b], [DAD03] et [DDA05b]. D'autres résultats sur la commande des graphes d'événements p-temporels et des GET sous contraintes, définies par un graphe d'événements p-temporel, sont fournis dans [DDA05a], et constitue un prolongement de ce présent travail.

5.2 Commande des graphes d'événements temporisés

5.2.1 Re-formulation de la méthode "Backward"

La méthode dite "Backward" établie dans [BCOQ92] et [CGQ93] permet l'évaluation d'une séquence de dates d'entrées afin de mieux gérer les dates de sortie. La méthode Backward présente une solution au problème de la commande sous forme d'un système dans lequel la résiduation (\setminus) et l'opération $\min(\wedge)$ sont utilisées.

Il est connu que, pour le système pour lequel les équations du type dateur donnent la solution minimale (solution au plus tôt), la plus grande solution (solution au plus tard) est donnée explicitement par les équations récursives appelées "Backward" où le co-vecteur joue le rôle du vecteur d'état. Les équations d'état et les équations récursives donnent respectivement les dates au plus tôt et au plus tard des tâches. La différence entre le co-état et l'état représente le temps disponible pour le tir des transitions. L'existence d'une différence négative empêche le respect des dates désirées. Cette approche exige donc la connaissance des valeurs du vecteur d'état.

Nous énonçons maintenant les résultats relatifs à la méthode "Backward" avant de reformuler cette dernière. Cette reformulation permettra de faire un comparatif avec la méthode

proposée dans cette partie.

Le modèle d'état représentant un graphe d'événements temporisé par des séries formelles en γ est donné par le système suivant :

$$\begin{cases} X(\gamma) &= A\gamma X(\gamma) \oplus BU(\gamma) \\ Y(\gamma) &= CX(\gamma) \leq Z(\gamma) \end{cases} \quad (5.1)$$

La résolution du modèle 5.1 aboutit au système dit Système Adjoint :

$$\begin{cases} \xi(\gamma) &= A\backslash\gamma^{-1}\xi(\gamma) \wedge C\backslash Z(\gamma) \\ \hat{u}(\gamma) &= B\backslash\xi(\gamma) \end{cases} \quad (5.2)$$

ξ est appelé le co-état.

En développant ce dernier système, nous pourrions obtenir les équations classiques dites "Backward" paramétrées en k .

$$\begin{cases} \xi(k) &= A\backslash\xi(k+1) \wedge C\backslash z(k) \\ \hat{u}(k) &= B\backslash\xi(k) \end{cases} \quad (5.3)$$

Dans le cadre de la commande des GET dans l'algèbre $(\max, +)$, le système 5.3 représente le résultat classique et connu de l'approche appelée communément méthode "Backward". Nous proposons maintenant une re-formulation de ce résultat.

Proposition 5.1. *la méthode "Backward" résout le problème suivant : calculer le plus grand co-état et la plus grande commande notés respectivement $\xi(k)$ et $\hat{u}(k)$ tels que :*

$$\begin{cases} Ax(k) \oplus Bu(k+1) &\leq x(k+1) \\ Cx(k) &\leq z(k) \end{cases} \quad (5.4)$$

Preuve :

Re-formulée autrement, la résolution du système 5.2 est équivalente à la recherche des plus grandes valeurs de $\theta(k)$ et $u(k)$, qui vérifient les inéquations suivantes et qui correspondent respectivement à $\xi(k)$ et $\hat{u}(k)$.

$$\begin{cases} \theta(k) &\leq A\backslash\theta(k+1) \\ \theta(k) &\leq C\backslash z(k) \\ u(k) &\leq B\backslash\theta(k) \end{cases}$$

Par définition de la résiduation :

$$\begin{aligned} \Leftrightarrow \begin{cases} A\theta(k) &\leq \theta(k+1) \\ C\theta(k) &\leq z(k) \\ Bu(k) &\leq \theta(k) \end{cases} &\Leftrightarrow \begin{cases} A\theta(k) &\leq \theta(k+1) \\ Bu(k+1) &\leq \theta(k+1) \\ C\theta(k) &\leq z(k) \end{cases} \\ &\Leftrightarrow \begin{cases} \theta(k+1) &\geq A\theta(k) \oplus Bu(k+1) \\ C\theta(k) &\leq z(k) \end{cases} \end{aligned}$$

En d'autres termes, la méthode "Backward" résout le problème suivant : calculer le plus grand état et la plus grande commande notés respectivement $\xi(k)$ et $\hat{u}(k)$ tels que :

$$\begin{cases} (1) & A\theta(k) \oplus Bu(k+1) \leq \theta(k+1) \\ (2) & C\theta(k) \leq z(k) \end{cases} \quad (5.5)$$

□

Autrement dit, le dernier système 5.5 ne donne pas une description d'un fonctionnement au plus tôt qui est représenté par les égalités strictes des équations d'état habituelles. Nous pouvons donc conclure que le problème résolu est relaxé par rapport au problème fondamental, et que le co-état calculé n'est pas obligatoirement admissible, c'est à dire ne vérifie pas obligatoirement l'équation d'état.

5.2.2 Synthèse de la commande par la méthode de propagation de contraintes MD

L'objectif du problème que nous abordons ici est formulé de la manière suivante. Étant donnée une trajectoire de sortie désirée $z(k)$, on cherche à calculer la plus grande commande $u(k)$ tel que la réponse du graphe à cette entrée produit une sortie $y(k)$ vérifiant la contrainte suivante :

$$y(k) \leq z(k) \quad (5.6)$$

En raison de la propriété d'isotonie matricielle, obtenir le plus grand $u(k)$ entraîne le calcul du plus grand état et de la plus grande sortie en raison de respectivement l'équation d'état et de l'équation relative à la sortie.

La solution peut donc être obtenue par le calcul des plus grands u et x notés respectivement u^+ et x^+ vérifiant le système des contraintes suivant :

$$\begin{cases} x(k) = Ax(k-1) \oplus Bu(k) \\ y(k) = Cx(k) \\ y(k) \leq z(k) \end{cases} \quad (5.7)$$

avec $k \in [k_s \ k_f]$

En résumé, pour le problème de commande traité ici, l'ensemble de la séquence de production désirée $z(k)$ est supposé connu, tandis que la commande $u(k)$ et l'état $x(k)$ sont les variables à déterminer. Nous qualifions ce problème de fondamental.

La proposition suivante nous permet d'introduire une nouvelle formulation du problème précédent.

Proposition 5.2. *Le problème de la commande peut être re-formulé de la manière suivante. Calculer les plus grands u et x vérifiant les contraintes suivantes :*

$$\begin{cases} x(k) \leq [Ax(k-1) \oplus Bu(k)] \wedge [A \setminus x(k+1)] \wedge [C \setminus z(k)] \\ u(k) \leq B \setminus x(k) \end{cases} \quad (5.8)$$

Preuve :

Si nous considérons uniquement l'état, en remplaçant $y(k)$ par son expression (on suppose que le système fonctionne au plus tôt), le système 5.7 peut se réécrire :

$$\begin{cases} x(k) & = & Ax(k-1) \oplus Bu(k) \\ Cx(k) & \leq & z(k) \end{cases}$$

$$\Leftrightarrow \begin{cases} x(k) \leq Ax(k-1) \oplus Bu(k) \\ Ax(k-1) \oplus Bu(k) \leq x(k) \\ Cx(k) \leq z(k) \end{cases} \Leftrightarrow \begin{cases} x(k) \leq Ax(k-1) \oplus Bu(k) \\ Ax(k-1) \leq x(k) \\ Bu(k) \leq x(k) \\ Cx(k) \leq z(k) \end{cases}$$

En exploitant les résultats de la résiduation :

$$\begin{cases} x(k) \leq C \setminus z(k) \\ x(k) \leq Ax(k-1) \oplus Bu(k) \\ x(k) \leq A \setminus x(k+1) \\ u(k) \leq B \setminus x(k) \end{cases} \quad (5.9)$$

$$\Leftrightarrow \begin{cases} x(k) \leq [Ax(k-1) \oplus Bu(k)] \wedge [A \setminus x(k+1)] \wedge [C \setminus z(k)] \\ u(k) \leq B \setminus x(k) \end{cases}$$

□

On remarquera que l'expression obtenue utilise les opérations min, max, ainsi que la résiduation. La variable $x(k)$ est la minimisation d'un terme backward $[A \setminus x(k+1)] \wedge [C \setminus z(k)]$ comme dans l'approche classique et d'un terme forward $Ax(k-1) \oplus Bu(k)$ correspondant à l'équation d'état.

Après avoir reconstruit le système d'équations modélisant le processus ainsi que les contraintes qu'il doit respecter, nous pouvons appliquer l'algorithme MD de propagation de contraintes sur ce nouveau système afin de calculer les plus grandes valeurs de $x(k)$ et de $u(k)$ recherchées.

A partir du système 5.8, nous développons l'algorithme 1 donné ci-dessous à partir de l'algorithme général MD de propagation de contraintes. Cet algorithme est donc spécifique aux GET.

Algorithme 1:

Pas 0 : initialisation $\lambda(k_f) \leftarrow T$, $\mu(k_f) \leftarrow T$

Répéter tant que $\lambda(k)$, $\mu(k)$ et $u(k)$ ne sont pas constants

Pas 1 : $\lambda(k) \leftarrow [A \setminus \lambda(k+1)] \wedge [C \setminus z(k)]$ pour k allant de k_f à k_s

Pas 2 : $u(k) \leftarrow B \setminus \lambda(k)$

Pas 3 : $\mu(k_s) \leftarrow \lambda(k_s)$

Pas 4 : $\mu(k) \leftarrow \lambda(k) \wedge [A\mu(k-1) \oplus Bu(k)]$ pour k allant de k_s à k_f

Pas 5 : $u(k) \leftarrow B \setminus \mu(k)$

Bien que l'algorithme général ait la caractéristique d'être pseudo-polynomial, la proposition suivante montre que l'algorithme 1 que nous venons de présenter, converge en un seul coup.

Proposition 5.3. *Dans l'algorithme 1, la forme suivante est vérifiée :*

$$\mu(k) \leq [A \setminus \mu(k+1)] \wedge [C \setminus z(k)]$$

Preuve : On a :

$$\begin{aligned} [A \setminus \mu(k+1)] \wedge [C \setminus z(k)] &= A \setminus [\lambda(k+1) \wedge [A\mu(k) \oplus Bu(k+1)]] \wedge [C \setminus z(k)] \\ &= A \setminus \lambda(k+1) \wedge A \setminus [A\mu(k) \oplus Bu(k+1)] \wedge [C \setminus z(k)] \\ &= A \setminus \lambda(k+1) \wedge C \setminus z(k) \wedge A \setminus [A\mu(k) \oplus Bu(k+1)] \\ &= \lambda(k) \wedge A \setminus [A\mu(k) \oplus Bu(k+1)] \end{aligned}$$

D'une part, d'après le théorème 1.26, on peut écrire : $A \otimes (A \setminus x) \leq x$ et $A \setminus [A \otimes x] \geq x$

Ainsi, $A \setminus [A \otimes \mu(k)] \geq \mu(k)$ et $A \setminus [A\mu(k) \oplus Bu(k+1)] \geq A \setminus [A\mu(k)] \geq \mu(k)$ par isotonie de l'opération de résiduation \setminus .

D'autre part, $\mu(k) \leq \lambda(k)$ par construction. Nous concluons que : $\mu(k) \leq [A \setminus \mu(k+1)] \wedge [C \setminus z(k)]$.

D'où une convergence de l'algorithme 1 en un seul coup. D'autre part, la minimisation par $\mu(k)$ dans le pas 1 est inutile car l'algorithme démarre implicitement à T et n'effectue pas de nouvelle minimisation par le pas 1. □

Les propositions suivantes 5.4 et 5.5 permettront d'apporter des améliorations à l'algorithme 1.

En particulier, la proposition 5.4 suivante montre que la minimisation par le terme $\lambda(k)$ dans le pas 4 de l'algorithme 1 est inutile. Le pas 4 devient : $\mu(k) \leftarrow A\mu(k-1) \oplus Bu(k)$

Proposition 5.4. *Dans l'algorithme 2 ci-dessous, le co-état calculé dans le pas 4 est inférieur ou égal au co-état calculé dans le pas 1. On a la forme :*

$$\mu(k) \leq \lambda(k)$$

pour tout k de k_s à k_f .

Preuve : On montre par récurrence que la propriété $H_k : \mu(k) \leq \lambda(k)$ est vérifiée.

Soit $H_0 : \mu(k_s) = \lambda(k_0)$ et donc H_0 est vérifiée.

Nous supposons que H_k est vérifiée et nous cherchons à montrer que H_{k+1} l'est aussi.

Par hypothèse $\mu(k) \leq \lambda(k)$ pour $k \geq k_s$

\implies

$$\begin{aligned} \mu(k+1) &= A\mu(k) \oplus Bu(k+1) \leq A\lambda(k) \oplus Bu(k+1) \\ &= A[A\lambda(k+1) \wedge C\setminus z(k)] \oplus Bu(k+1) \\ &\leq A[A\lambda(k+1)] \oplus Bu(k+1) \\ &\leq \lambda(k+1) \oplus Bu(k+1) \end{aligned}$$

Or, au pas 2 de l'algorithme 1, $u(k) = B\lambda(k)$ et donc $Bu(k+1) = B(B\lambda(k+1)) \leq \lambda(k+1)$.

En conséquence, $\mu(k+1) \leq \lambda(k+1)$, d'où la propriété H_{k+1} est vérifiée. □

Dans la proposition 5.5, on montre que le calcul de la commande donné dans le pas 5 de l'algorithme 1, peut être supprimé.

Proposition 5.5. *Dans l'algorithme 1, le calcul de la commande dans le pas 5, donne le même résultat que dans le pas 2.*

Preuve : D'une part, dans la proposition 5.4, on a montré que la minimisation de $\mu(k)$ par $\lambda(k)$ est inutile. Cela implique que la commande calculée dans le pas 5 peut être donnée par $u_{pas5}(k) = B\setminus\mu(k) = B\setminus[A\mu(k-1) \oplus Bu_{pas2}(k)]$ (forme simplifiée du pas 4).

$$\begin{aligned} u_{pas5}(k) &\geq B\setminus(A\mu(k-1)) \oplus B\setminus(Bu_{pas2}(k)) \\ &\geq B\setminus(Bu_{pas2}(k)) \\ &\geq u_{pas2}(k) \quad \text{d'après la formule } f_2 \text{ donnée dans le théorème 1.31} \end{aligned}$$

D'autre part, comme $\mu(k) \leq \lambda(k)$ et la commande est donnée par $u_{pas2}(k) = B\setminus\lambda(k)$ et $u_{pas5}(k) = B\setminus\mu(k)$. Par conséquent $u_{pas5}(k) \leq u_{pas2}(k)$ (par isotonie de la résiduation \setminus).

On obtient donc $u_{pas5}(k) = u_{pas2}(k)$ □

En conclusion, la commande est identique au cas inégalité, c'est à dire celle calculée par la méthode "Backward".

Le calcul de la commande par la méthode "Backward" utilise le terme $A\setminus\xi(k+1) \wedge C\setminus z(k)$ résidué à gauche par la matrice B . De même, d'après la proposition 5.5, la méthode de propagation de contraintes utilise le terme "Backward" $A\setminus\lambda(k+1) \wedge C\setminus z(k)$ résidué à gauche par la matrice B , pour le calcul de la commande. Nous concluons alors que la méthode Backward et

la méthode de propagation de contraintes, réalisent la même synthèse de commande. D'après les résultats énoncés dans les propositions 5.3, 5.4 et 5.5, les améliorations portées sur l'algorithme 1 donnent la nouvelle version de cet algorithme.

Algorithme 2 :

Pas 0 : initialisation $\lambda(k_f) \leftarrow T$, $\mu(k_f) \leftarrow T$

Pas 1 : $\lambda(k) \leftarrow [A \setminus \lambda(k+1)] \wedge [C \setminus z(k)]$ pour k allant de k_f à k_s

Pas 2 : $u(k) \leftarrow B \setminus \lambda(k)$ pour k allant de k_s à k_f

Pas 3 : $\mu(k_s) \leftarrow \lambda(k_s)$

Pas 4 : $\mu(k) \leftarrow [A\mu(k-1) \oplus Bu(k)]$ k allant de k_s à k_f

Proposition 5.6. *L'algorithme 2 en aller-retour, donne la plus grande estimation possible du co-état égalité.*

Preuve : Partant de $\lambda(k_f + 1) = T$, les pas 1 et 2 donnent la plus grande solution de :

$$\begin{cases} Ax(k-1) \oplus Bu(k) & \geq x(k) \\ Cx(k) & \geq z(k) \end{cases} \quad (5.10)$$

En particulier, la solution en $\lambda(k_s)$ k_s ne peut être dépassée.

Les pas 3 et 4 donnent une solution vérifiant l'égalité de l'équation d'état $x(k) = Ax(k-1) \oplus Bu(k)$.

A chaque $x(k_s)$ est associée une unique trajectoire construite à partir de l'équation d'état. En partant de $\mu(k_s) = \lambda(k_s)$, l'équation d'état va donc donner une unique trajectoire. La proposition 5.4 indique que $\mu(k) \leq \lambda(k)$ pour tout k de k_s à k_f et opère une minimisation. Il ne peut avoir de trajectoire plus grande avec $\lambda(k) \leq \mu(k)$ avec $\lambda(k) \neq \mu(k)$.

Montrons que μ vérifie la contrainte 5.8. Par construction μ vérifie

$\mu(k) = A\mu(k-1) \oplus Bu(k) \leq \lambda(k) \leq [A \setminus \lambda(k+1)] \wedge [C \setminus z(k)]$. Elle vérifie donc également $\mu(k) \leq C \setminus z(k)$. L'inégalité suivante est aussi vérifiée : $\mu(k) \leq A \setminus \mu(k+1)$. En effet, $\mu(k+1) = A\mu(k) \oplus Bu(k+1)$ entraîne en particulier $A\mu(k) \oplus Bu(k+1) \leq \mu(k+1)$. On en déduit que $\mu(k) \leq A \setminus \mu(k+1)$. Comme $\mu(k) = A\mu(k-1) \oplus Bu(k) \leq A \setminus \mu(k+1) \wedge [C \setminus z(k)]$, μ vérifie la contrainte 5.8 :

$$\mu(k) \leq [A\mu(k-1) \oplus Bu(k)] \wedge [A \setminus \mu(k+1)] \wedge [C \setminus z(k)]$$

et c'est également la plus grande estimation car on suit la démarche de l'algorithme MD général.

□

En résumé, à partir de l'algorithme MD général, les expressions habituelles du calcul de la commande et de l'état ont été retrouvées. L'apport ici, n'est donc pas au niveau des expressions trouvées, mais dans le fait que l'on puisse les retrouver à partir d'un algorithme très général.

5.2.3 Commentaires sur le co-état

Soit le plus grand co-état noté $\xi(k)$ tel que les inéquations (1) et (2) du système 5.5 de la méthode "Backward" soient vérifiées. Soit également le plus grand état admissible, et au plus tard noté $\nu(k)$, correspondant au problème fondamental, c'est à dire, tel que le modèle d'état soit vérifié. Comme $\nu(k)$ doit en plus vérifier une contrainte supplémentaire par rapport au co-état qui est

$$x(k) \leq [Ax(k-1) \oplus Bu(k)] \quad (5.11)$$

nous pouvons également déduire que $\nu(k) \leq \xi(k)$. Pour tout état admissible $x(k)$ respectant la condition de la consigne, l'inégalité suivante doit être vérifiée $x(k) \leq \nu(k)$. Sous l'hypothèse de l'existence d'une commande au problème posé, nous aboutissons donc à :

$$\forall k \in [k_s, k_f], \quad x(k) \leq \nu(k) \leq \xi(k)$$

Ainsi, la quantité $\xi_i(k)$ est la dernière date où la transition x_i devrait être tirée la kème fois sans faire retarder les dates de sorties $y(k)$ au delà de $z(k)$. En résumé, le vecteur $\xi(k)$ correspond donc au fonctionnement au plus tard et n'est pas admissible. A l'opposé, le vecteur d'état $x(k)$ fonction de la commande et de l'évolution passée avant k_s ne correspond pas au fonctionnement au plus tard mais au plus tôt et est donc admissible. Entre les deux, le vecteur $\nu(k)$ est le vecteur admissible au plus tard.

Cette description permet d'introduire une nouvelle marge de temps. Il est connu que la quantité $\xi_i(k) - x_i(k)$ représente le temps ou la marge de temps disponible pour tirer la transition x_i pour la kème fois afin que le système respecte la condition $C\xi(k) \leq z(k)$. Il est nécessaire que l'ensemble de ces quantités soient positives sous peine d'absence de solution au problème classique. Dans ce cas, on modifie le problème en adaptant la consigne à l'évolution du système. Maintenant, si le système doit respecter non seulement la condition $C\xi(k) \leq z(k)$ mais aussi le modèle d'état en fonctionnement au plus tôt, la quantité $\nu_i(k_s) - x_i(k_s)$ représente le temps ou la marge de temps disponible pour tirer la transition x_i pour $k = k_s$. Les trajectoires x et ν définissent un domaine de trajectoires admissibles sur l'horizon $[k_s, k_f]$.

5.2.4 Illustration

Nous considérons l'exemple du graphe d'événements temporisé suivant :

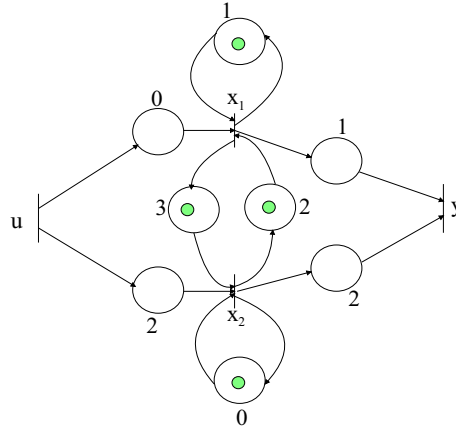


Figure 5.2 : Exemple d'un GET

Sous l'hypothèse d'un fonctionnement au plus tôt du graphe de la figure 5.2, Le modèle d'état correspondant est donné par :

$$\begin{cases} \begin{pmatrix} x_1(k) \\ x_2(k) \end{pmatrix} = \begin{pmatrix} 1 & 3 \\ 2 & e \end{pmatrix} \begin{pmatrix} x_1(k-1) \\ x_2(k-1) \end{pmatrix} \oplus \begin{pmatrix} e \\ 2 \end{pmatrix} u(k) \\ y(k) = \begin{pmatrix} 1 & 2 \end{pmatrix} \begin{pmatrix} x_1(k) \\ x_2(k) \end{pmatrix} \leq z(k) \end{cases}$$

La consigne $z(k)$ est donnée comme suit : $z(1) = 8$ $z(2) = 11$ $z(3) = 15$ $z(4) = 18$

Par la suite, nous allons appliquer les deux approches présentées précédemment afin de résoudre le problème de commande posé dans cet exemple. Une comparaison entre les résultats donnés par chacune des méthodes sera aussi exposée.

5.2.4.1 Méthode "Backward"

Une application de la méthode "Backward" nous permet d'aboutir au résultat suivant :

$$\xi_1(k) = (z(k) - 1) \wedge (\xi_1(k+1) - 1) \wedge (\xi_2(k+1) - 2)$$

$$\xi_2(k) = (z(k) - 2) \wedge (\xi_1(k+1) - 3) \wedge (\xi_2(k+1))$$

$$\hat{u}(k) = \xi_1(k) \wedge (\xi_2(k) - 2)$$

Applications numériques :

k	1	2	3	4
z	8	11	15	18
ξ_1	7	10	14	17
ξ_2	6	9	13	16
\hat{u}	4	7	11	14

Vérification de l'équation 5.11:

$$\begin{pmatrix} 17 \\ 16 \end{pmatrix} \not\leq \begin{pmatrix} 1 & 3 \\ 2 & e \end{pmatrix} \begin{pmatrix} 14 \\ 13 \end{pmatrix} \oplus \begin{pmatrix} e \\ 2 \end{pmatrix} 14 = \begin{pmatrix} 16 \\ 16 \end{pmatrix}$$

$$\begin{pmatrix} 14 \\ 13 \end{pmatrix} \not\leq \begin{pmatrix} 1 & 3 \\ 2 & e \end{pmatrix} \begin{pmatrix} 10 \\ 9 \end{pmatrix} \oplus \begin{pmatrix} e \\ 2 \end{pmatrix} 11 = \begin{pmatrix} 12 \\ 13 \end{pmatrix}$$

$$\begin{pmatrix} 10 \\ 9 \end{pmatrix} \not\leq \begin{pmatrix} 1 & 3 \\ 2 & e \end{pmatrix} \begin{pmatrix} 7 \\ 6 \end{pmatrix} \oplus \begin{pmatrix} e \\ 2 \end{pmatrix} 7 = \begin{pmatrix} 9 \\ 9 \end{pmatrix}$$

On en conclut que dans le cas de la résolution "Backward", le co-état ne vérifie pas obligatoirement l'inéquation $x(k) \leq Ax(k-1) \oplus Bu(k)$ ni donc, l'équation d'état.

5.2.4.2 Méthode utilisant l'algorithme MD

Nous obtenons un système d'inéquations après avoir appliqué la méthode proposée :

$$\left\{ \begin{array}{l} x_1(k) \leq -1 + z(k) \\ x_2(k) \leq -2 + z(k) \\ x_1(k) \leq 1x_1(k-1) \oplus 3x_2(k-1) \oplus eu(k) \\ x_2(k) \leq 2x_1(k-1) \oplus ex_2(k-1) \oplus 2u(k) \\ x_1(k-1) \leq -1 + x_1(k) \\ x_2(k-1) \leq -3 + x_1(k) \\ x_1(k-1) \leq -2 + x_2(k) \\ x_2(k-1) \leq -e + x_2(k) \\ u(k) \leq -e + x_1(k) \\ u(k) \leq -2 + x_2(k) \end{array} \right.$$

Nous appliquons ici l'algorithme MD. La résolution est effectuée en traitant les inéquations et les variables $x_i(k)$ dans un ordre arbitraire à partir d'un $k \in [k_s, k_f]$ également quelconque.

Les résultats finaux sont regroupés dans le tableau suivant :

k	1	2	3	4
z	8	11	15	18
x_1^+	7	9	12	16
x_2^+	6	9	13	16
u^+	4	7	11	14

Nous pouvons ainsi constater que les valeurs de la commande sont identiques avec celles de la méthode "Backward".

Nous testons maintenant l'inéquation 5.11 :

$$\begin{pmatrix} 16 \\ 16 \end{pmatrix} \leq \begin{pmatrix} 1 & 3 \\ 2 & e \end{pmatrix} \begin{pmatrix} 14 \\ 13 \end{pmatrix} \oplus \begin{pmatrix} e \\ 2 \end{pmatrix} 14 = \begin{pmatrix} 16 \\ 16 \end{pmatrix}$$

$$\begin{pmatrix} 12 \\ 13 \end{pmatrix} \leq \begin{pmatrix} 1 & 3 \\ 2 & e \end{pmatrix} \begin{pmatrix} 10 \\ 9 \end{pmatrix} \oplus \begin{pmatrix} e \\ 2 \end{pmatrix} 11 = \begin{pmatrix} 12 \\ 13 \end{pmatrix}$$

$$\begin{pmatrix} 9 \\ 9 \end{pmatrix} \leq \begin{pmatrix} 1 & 3 \\ 2 & e \end{pmatrix} \begin{pmatrix} 7 \\ 6 \end{pmatrix} \oplus \begin{pmatrix} e \\ 2 \end{pmatrix} 7 = \begin{pmatrix} 9 \\ 9 \end{pmatrix}$$

Sur cet exemple, on observe que l'inéquation $x(k) \leq Ax(k-1) \oplus Bu(k)$ est vérifiée ce qui montre que l'état suit une trajectoire admissible : le signe \leq peut être remplacé par un signe égal strict, dans le système ci-dessus.

5.2.4.3 Simulation

Pour l'état initial $x(0) = \begin{pmatrix} \varepsilon \\ \varepsilon \end{pmatrix}$

k	0	1	2	3	4
u		4	7	11	14
x	ϵ	4	9	12	16
	ϵ	6	9	13	16
y	8	8	11	15	18

Nous vérifions que $\forall k \in [k_s, k_f]$, $x(k) \leq \nu(k) \leq \xi(k)$. Pour cela, nous comparons les valeurs de $x(k)$, $\nu(k)$ et $\xi(k)$ qui correspondent respectivement aux vecteurs suivants.

$$\begin{aligned}
 k=1 & \begin{pmatrix} 4 \\ 6 \end{pmatrix} \leq \begin{pmatrix} 7 \\ 6 \end{pmatrix} \leq \begin{pmatrix} 7 \\ 6 \end{pmatrix} \\
 k=2 & \begin{pmatrix} 9 \\ 9 \end{pmatrix} \leq \begin{pmatrix} 9 \\ 9 \end{pmatrix} \leq \begin{pmatrix} 10 \\ 9 \end{pmatrix} \\
 k=3 & \begin{pmatrix} 12 \\ 13 \end{pmatrix} \leq \begin{pmatrix} 12 \\ 13 \end{pmatrix} \leq \begin{pmatrix} 14 \\ 13 \end{pmatrix} \\
 k=4 & \begin{pmatrix} 16 \\ 16 \end{pmatrix} \leq \begin{pmatrix} 16 \\ 16 \end{pmatrix} \leq \begin{pmatrix} 17 \\ 16 \end{pmatrix}
 \end{aligned}$$

Finalement, nous déduisons que $x(k) \leq \nu(k) \leq \xi(k)$ avec $x(k) \neq \nu(k)$ et $\nu(k) \neq \xi(k)$ pour cet exemple avec $x(0) = \begin{pmatrix} \varepsilon \\ \varepsilon \end{pmatrix}$.

5.3 Commande en juste à temps des graphes d'événements à flux temporels

5.3.1 Formulation et existence d'une loi de commande

L'objectif de la commande en juste à temps ici est de calculer le plus grand contrôle u tel que $y \leq z$ avec $y(\gamma) = C \otimes x(\gamma)$. Dans le contexte de cette étude, nous considérons le modèle d'état qui correspond aux graphes d'événements à flux temporels :

$$\begin{cases} A^- \otimes x(\gamma) \oplus B^- \otimes u(\gamma) \leq x(\gamma) \leq \bigwedge_{i=1}^{j_i} A_i^+ \otimes x(\gamma) \oplus B_i^+ \otimes u(\gamma) \\ C(\gamma) \otimes x(\gamma) = y(\gamma) \end{cases} \quad (5.12)$$

les coefficients des matrices A^- , B^- , A_i^+ , B_i^+ , et C appartiennent à $\overline{\mathbb{R}}_{max}[\gamma]$

Une première formulation du problème de commande est donnée par la proposition 5.7. Les propriétés de la résiduation énoncées dans le paragraphe 1.4.2 permettent d'arriver au résultat énoncé dans cette proposition.

Proposition 5.7. *Pour un graphe d'événements à flux temporel dont la représentation d'état est donnée par le système 5.12, le problème de commande en juste à temps peut être formulé comme suit : pour k allant de k_s à k_f , chercher le plus grand vecteur $\begin{pmatrix} x(\gamma) \\ u(\gamma) \end{pmatrix}$ tel que :*

$$\begin{pmatrix} x(\gamma) \\ u(\gamma) \end{pmatrix} \leq h \begin{pmatrix} x(\gamma) \\ u(\gamma) \end{pmatrix} \quad (5.13)$$

avec

$$h \begin{pmatrix} x(\gamma) \\ u(\gamma) \end{pmatrix} = \begin{pmatrix} (\gamma^{-1}x(\gamma)) \wedge (A^- \setminus x(\gamma)) \wedge (C \setminus z(\gamma)) \wedge \\ \bigwedge_{i=1}^{j_1} (A_i^+ \otimes x(\gamma) \oplus B_i^+ u(\gamma)) \\ B^- \setminus x(\gamma) \end{pmatrix} \quad (5.14)$$

Preuve : La démonstration est immédiate. Il suffit de considérer les inéquations du système 5.12 et d'appliquer directement le théorème 1.32 sur la résiduation matricielle. Nous obtenons alors le système suivant

$$\begin{cases} x(\gamma) \leq (\gamma^{-1}x(\gamma)) \wedge (A^- \setminus x(\gamma)) \wedge \bigwedge_{i=1}^{j_1} A_i^+ \otimes x(\gamma) \oplus B_i^+ \otimes u(\gamma) \\ x(\gamma) \leq C \setminus z(\gamma) \\ u(\gamma) \leq B^- \setminus x(\gamma) \end{cases} \quad (5.15)$$

d'où les systèmes 5.13 et 5.14. □

Le calcul de la plus grande commande revient alors à résoudre le système 5.13 dont la forme du point fixe est la suivante :

$$x \leq f(x) \quad (5.16)$$

avec f du type (min, max, +). Comme la fonction h du système 5.13 est non-homogène, elle présente la forme du système 4.16, et l'application de la théorie spectrale est impossible. Nous appliquons la procédure de relaxation ou d'homogénéisation vue dans le quatrième chapitre (voir le paragraphe 4.3.2) afin de remédier à ce problème.

La nouvelle formule relaxée, qui est une forme homogène de la fonction h permettant la résolution du problème de la commande, est donnée dans la proposition 5.8 suivante. Celle-ci fournit une condition nécessaire et suffisante d'existence d'une solution au problème de commande.

Remarque : Tous les résultats relatifs à l'opération de relaxation présentés dans le cadre de l'étude de l'estimation d'état sont aussi valables ici.

◇

Proposition 5.8. *Le système 5.13 admet une solution finie sur un horizon l si et seulement, si $\chi(\Phi_H(X_l)) \geq 0$ avec H donnée par :*

$$\begin{pmatrix} x_0 \\ x(\gamma) \\ u(\gamma) \end{pmatrix} \leq H \begin{pmatrix} x_0 \\ x(\gamma) \\ u(\gamma) \end{pmatrix} \quad (5.17)$$

avec

$$H \begin{pmatrix} x_0 \\ x(\gamma) \\ u(\gamma) \end{pmatrix} = \begin{pmatrix} x_0 \\ (\gamma^{-1}x(\gamma)) \wedge (A^- \setminus x(\gamma)) \wedge (C \setminus (z(\gamma) \otimes x_0)) \wedge \\ \bigwedge_{i=1}^{j_i} (A_i^+ \otimes x(\gamma) \oplus B_i^+ u(\gamma)) \\ B^- \setminus x(\gamma) \end{pmatrix} \quad (5.18)$$

Nous donnons seulement le squelette de la démonstration.

Preuve : D'une part, on rappelle que l'opération de relaxation consiste à introduire une variable x_0 . A partir du système 5.14 on obtient alors : $x(\gamma) \leq (A^- \setminus x(\gamma)) \wedge (C \setminus (z(\gamma) \otimes x_0)) \wedge \bigwedge_{i=1}^{j_i} (A_i^+ \otimes x(\gamma) \oplus B_i^+ u(\gamma))$. Comme $x_0 \leq x_0$, on obtient le système 5.18.

D'autre part, le système 5.17 présente la forme :

$$x \leq F(x) \quad (5.19)$$

avec F une fonction (min, max, +) homogène. Nous appliquons ensuite les résultats de la théorie spectrale (théorème 1.56) pour vérifier l'existence d'une solution pour le système 5.17.

□

Détaillons maintenant la structure de l'application H qui va être appliquée. L'application H donnée par la relation 5.18 peut se réécrire sous la forme suivante :

$$H = H_1 \wedge H_2 \quad (5.20)$$

L'application $H_1 = \begin{pmatrix} x_0 \\ (\gamma^{-1}x(\gamma)) \wedge (A^- \setminus x(\gamma)) \wedge (C \setminus (z(\gamma) \otimes x_0)) \\ B^- \setminus x(\gamma) \end{pmatrix}$ représente la partie "backward" qui correspond aussi à la commande en "juste à temps" dans le cas d'un graphe d'événements temporisé.

La partie "forward" correspond à $H_2 = \begin{pmatrix} T \\ \bigwedge_{i=1}^{j_i} (A_i^+ \otimes x(\gamma) \oplus B_i^+ u(\gamma)) \\ T \end{pmatrix}$

Définition 5.9. Commandabilité

Un graphe d'événements est structurellement commandable si, chaque transition interne peut être atteinte par au moins un chemin venant d'une transition d'entrée.

La commandabilité structurelle donne une condition pour observer un effet dans une transition interne, dont l'origine vient d'au moins une transition d'entrée. Ceci nous permet d'introduire les deux propositions suivantes 5.10 et 5.11 dont les démonstrations sont similaires aux preuves faites respectivement pour les propositions 4.9 et 4.10 du chapitre 4. Nous considérons aussi que l'horizon de travail ici est infini.

Proposition 5.10. *Un graphe d'événements structurellement commandable à flux temporel dont les sémantiques sont le Et-Pur et le Et-Faible, vérifie $\chi(\Phi_{H_1}(X_{+\infty})) = 0$. La plus grande solution de la partie "backward" h_1 satisfait le système $h \begin{pmatrix} x \\ u \end{pmatrix} \leq \begin{pmatrix} x \\ u \end{pmatrix}$ et donne une borne supérieure finie pour l'ensemble des solutions de $\begin{pmatrix} x \\ u \end{pmatrix} \leq h \begin{pmatrix} x \\ u \end{pmatrix}$ et $\begin{pmatrix} x \\ u \end{pmatrix} = h \begin{pmatrix} x \\ u \end{pmatrix}$*

La démonstration est analogue à la proposition 4.9.

Proposition 5.11. *Dans un graphe d'événements structurellement commandable à flux temporel dont les sémantiques sont le Et-Pur et le Et-Faible, il existe au moins une solution finie satisfaisant l'égalité :*

$$\begin{pmatrix} x_0 \\ x(\gamma) \\ u(\gamma) \end{pmatrix} = H \begin{pmatrix} x_0 \\ x(\gamma) \\ u(\gamma) \end{pmatrix} \quad (5.21)$$

ou il n'existe pas de solution pour le système 5.13.

La preuve est similaire à la proposition 4.10.

5.3.2 Calcul de la commande

Comme pour le problème de l'estimation d'état vu dans le chapitre 4 (section 4.4), et de manière analogue à l'algorithme développé dans le paragraphe 4.4.1, nous proposons ici un algorithme qui permet de calculer une commande optimale dans les graphes d'événements à flux temporels. Nous gardons les mêmes notations utilisées dans l'algorithme 1 au paragraphe 4.4.1. Le vecteur $z(k)$ représente la séquence de sortie désirée.

Algorithme 3 :

Pas 0 : initialisation $\mu_i(k_f) \leftarrow T, \lambda_i(k_f) \leftarrow T$

Répéter jusqu'à : $\lambda_i(k) = \mu_i(k)$ pour $1 \leq i \leq n$, $1 \leq j \leq m$ et $k_s \leq k \leq k_f$

Pas 1 : $\lambda_i(k) \leftarrow \mu_i(k) \wedge [A^-(., i) \setminus \lambda(k+1)] \wedge [C(., i) \setminus z(k)]$ de $k = k_f$ à k_s

Pas 2 : $u_j(k) \leftarrow B^-(., j) \setminus \lambda(k) \quad \forall k \in [k_s, k_f]$

Pas 3 : $\mu_i(k_s) \leftarrow \lambda_i(k_s)$

$$\mu_i(k) \leftarrow \lambda_i(k) \wedge \bigwedge_{l=1}^{j(i)} [A_l^+(i, .) \otimes \mu(k-1) \oplus B_l^+(i, .) \otimes u(k)] \quad \text{de } k = k_{s+1} \text{ à } k_f$$

Pas 4 : $u_j(k) \leftarrow B^-(., j) \setminus \mu(k) \quad \forall k \in [k_s, k_f]$

Cet algorithme est fait de manière similaire à l'algorithme 1 du paragraphe 4.4.1 et les mêmes commentaires peuvent être affirmés. Rappelons que l'algorithme relatif à la commande proposé ici, est aussi pseudo-polynomial.

5.3.3 Illustration

Nous considérons dans l'exemple de la figure 5.3 un graphe d'événements à flux temporel (sémantiques considérées : Et-Faible et Et-Pur). Cet exemple permet d'illustrer l'approche proposée dans ce chapitre pour le calcul d'une commande optimale. La sortie désirée (consigne) est donnée par le tableau ci-dessous. On rappelle que l'objectif est de calculer $u(k)$ tel que $y(k) \leq z(k)$ pour tout k allant de 1 à 7.

k	1	2	3	4	5	6	7
z	7	11	15	18	22	25	29

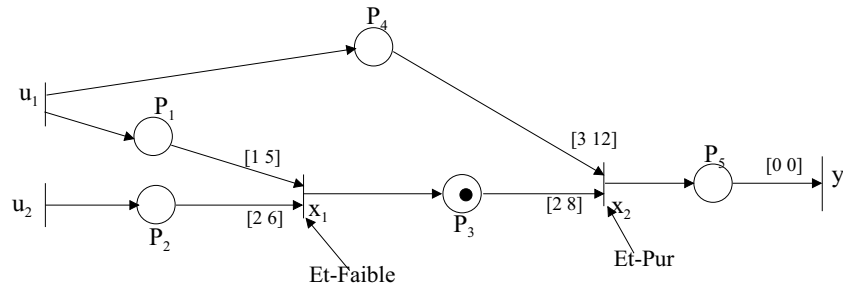


Figure 5.3 : Exemple d'un graphe d'événements à flux temporel

La première étape consiste à modéliser le graphe de la figure 5.3. Le système 5.22 présente alors le modèle correspondant à ce graphe.

$$\begin{cases} (u_1(k) + 1) \oplus (u_2(k) + 2) \leq x_1(k) \leq (u_1(k) + 5) \oplus (u_2(k) + 6) \\ (u_1(k) + 3) \oplus (x_1(k-1) + 2) \leq x_2(k) \leq (u_1(k) + 12) \wedge (x_1(k-1) + 8) \end{cases} \quad (5.22)$$

On pose $X = \{u_1, u_2, x_1, x_2\}$.

La démarche de la construction du graphe de calcul ainsi que le calcul du vecteur spectral est présentée dans des exemples au chapitre 3. On s'est limité ici à donner le résultat de calcul du vecteur spectral. Nous avons obtenu $\chi(\Phi_{g^+}(X_7)) \geq 0$. Par conséquent le système 5.3 admet une solution finie, d'où on peut calculer une commande optimale ainsi que le plus grand état.

Le système peut se réécrire sous la forme matricielle suivante :

$$\begin{aligned} & A^- \otimes \begin{pmatrix} x_1(k-1) \\ x_2(k-1) \end{pmatrix} \oplus B^- \otimes \begin{pmatrix} u_1(k) \\ u_2(k) \end{pmatrix} \leq \begin{pmatrix} x_1(k) \\ x_2(k) \end{pmatrix} \\ & \leq \left[A_1^+ \otimes \begin{pmatrix} x_1(k-1) \\ x_2(k-1) \end{pmatrix} \oplus B_1^+ \otimes \begin{pmatrix} u_1(k) \\ u_2(k) \end{pmatrix} \right] \wedge \left[A_2^+ \otimes \begin{pmatrix} x_1(k-1) \\ x_2(k-1) \end{pmatrix} \oplus B_2^+ \otimes \begin{pmatrix} u_1(k) \\ u_2(k) \end{pmatrix} \right] \end{aligned} \quad (5.23)$$

avec $A^- = \begin{pmatrix} \varepsilon & \varepsilon \\ 2 & \varepsilon \end{pmatrix}$, $B^- = \begin{pmatrix} 1 & 2 \\ 3 & \varepsilon \end{pmatrix}$, $A_1^+ = \begin{pmatrix} \varepsilon & \varepsilon \\ 8 & \varepsilon \end{pmatrix}$, $A_2^+ = \begin{pmatrix} T & T \\ \varepsilon & \varepsilon \end{pmatrix}$, $B_1^+ = \begin{pmatrix} 5 & 6 \\ \varepsilon & \varepsilon \end{pmatrix}$,
 $B_2^+ = \begin{pmatrix} T & T \\ 12 & \varepsilon \end{pmatrix}$ et $C = \begin{pmatrix} \varepsilon & 0 \end{pmatrix}$

Une mise en oeuvre de l'algorithme 3 sur cet exemple a permis de calculer la commande optimale ainsi que le plus grand état. Les résultats de simulation sont regroupés dans le tableau suivant :

k	1	2	3	4	5	6	7
x_1	9	13	16	20	23	27	31
x_2	7	11	15	18	22	25	29
u_1	4	8	12	15	19	22	26
u_2	7	11	14	18	21	25	29

A partir du tableau précédent regroupant les valeurs de la commande $u(k)$ et $x(k)$, on peut vérifier que la relation $y(k) = Cx(k) \leq z(k)$ pour tout k allant de 1 à 7.

5.4 Conclusion

Dans ce chapitre, une nouvelle approche a été proposée afin de résoudre le problème classique de la commande, ceci pour de nouveaux systèmes. L'objectif est donc que la sortie du système suive une évolution désirée selon le critère du "au plus tard".

Dans une première partie, une reformulation du problème de commande relatif aux GET, a donné une expression qui utilise les opérations min, max, l'addition ainsi que la résiduation. La variable d'état admissible et au plus tard, est la minimisation d'un terme "backward" comme dans l'approche classique, et d'un terme "forward" correspondant à l'équation d'état. Un algorithme spécifique aux GET permettant le calcul de la commande et du co-état les plus grands, a été développé. L'approche a été ensuite comparée avec la méthode classique "Backward". Nous avons montré que les deux méthodes donnent le même résultat pour le calcul de la séquence de commande. En revanche, dans la démarche proposée, l'équation d'état est vérifiée par le co-état, contrairement à la méthode "Backward" dont le co-état ne vérifie qu'une inégalité.

Dans une deuxième partie, nous avons abordé le problème de commande en juste à temps, dans les graphes d'événements à flux temporels, en donnant les grandes lignes de l'approche. Les sémantiques de tir choisies sont le Et-Pur et le Et-Faible. Ceci nous a permis d'obtenir une modélisation mathématique que l'on peut traiter et qui englobe une large classe de modèles. Au cours de cette partie, nous avons étudié l'existence d'une commande optimale respectant la sortie désirée, et garantissant le fonctionnement admissible du graphe. Afin d'appliquer la théorie spectrale, le passage entre deux formulations non-homogène et homogène, a été réalisé grâce à l'opération de relaxation ou d'homogénéisation que nous avons introduite dans le chapitre 4. Nous avons abordé la commandabilité structurelle dans ces graphes, afin de trouver une borne supérieure à l'ensemble des solutions ainsi qu'une proposition analysant l'espace des solutions. Enfin, nous avons élaboré un algorithme afin de calculer cette commande optimale. Afin de donner une illustration de cette approche, nous avons pris un exemple de graphe d'événements à flux temporel. Sur ce dernier toutes les étapes, de la modélisation à la mise en oeuvre de l'algorithme de calcul, en passant par la vérification d'une solution finie, ont été suivies. Rappelons pour terminer que les outils développés ici sont également valables pour les différents types de graphes d'événements temporels, comme le montre la figure 2.17 du chapitre 2 sur la modélisation.

Chapitre 6

Conclusion et perspectives

Contributions

Dans ce mémoire, nous avons abordé les problèmes de vivacité, d'estimation et de commande optimale dans les graphes d'événements (GE) temporisés et temporels. Ces différents thèmes s'appuient naturellement sur une étape préalable de modélisation de ces graphes dans les algèbres dites "exotiques". C'est l'objet du second chapitre. Celui-ci introduit la mise en équations de graphes d'événements temporisés mais aussi p-temporels, t-temporels, arcs temporels et à flux temporels. A notre connaissance, c'est la première fois que cela a été réalisé. L'intérêt de ces modèles est qu'ils introduisent un certain non-déterminisme dans la durée des temporisations. Les sémantiques de synchronisation peuvent, d'autre part, être complexes. De ce fait, une classification des différents graphes d'événements permet de clarifier du moins algébriquement ces phénomènes. Le modèle algébrique des graphes d'événements à flux temporels à sémantiques Et-fort et Et-Faible, représentent ainsi un modèle dit "d'intervalle" dont la forme généralise celui du temporisé mais aussi des autres GE considérés. Une seule forme du modèle d'intervalle suffit ainsi pour décrire les différents types de graphes d'événements. C'est ce modèle qui sera le support principal des trois chapitres suivants, c'est à dire la vivacité, l'estimation et la commande.

Dans ce chapitre de modélisation, nous considérons en particulier les GE p-temporels dont la particularité est la possibilité de mort de jetons, en raison de la présence de contraintes sur la durée de séjour maximale des jetons dans les places. Nous analysons les fonctionnements au plus tôt mais aussi au plus tard. Nous avons montré que le fonctionnement au plus tôt (et également au plus tard), pour les graphes d'événements p-temporels, ne peut être pris localement comme dans les GET mais globalement. Ainsi, la notion du fonctionnement au plus tôt des graphes d'événements p-temporels ne peut être définie de manière analogue aux GET et, une conséquence, est que la notion de fonction de transfert ne peut être exprimée simplement en raison de la complexité des phénomènes.

Un deuxième objectif est la validation des modèles du type intervalle et l'analyse de la vivacité des graphes d'événements p-temporels sous l'hypothèse que la structure du GE soit vivante. Nous avons d'abord établi des conditions sur la compatibilité des systèmes de type intervalle (f, g) . La première méthode analyse le graphe de calcul par le vecteur spectral. Le graphe de calcul présente cependant une structure complexe, faisant introduire des rétroactions et sa taille est, d'autre part, infinie. Un repérage purement manuel des circuits dans le graphe de calcul est extrêmement difficile, sinon impossible et exige une informatisation, ce qui est l'objet d'un travail actuellement, dans la communauté scientifique. Récemment, des outils et des approches ont été développées pour le calcul du vecteur spectral des fonctions $(\min, \max, +)$ [CZ02] et [CZ04], ainsi que pour la résolution des systèmes d'inéquations de type $(\min, \max, +)$ [CZ05].

Nous avons ensuite appliqué les résultats trouvés dans le cas où f et g sont respectivement de type $(\max, +)$ et $(\min, +)$ afin de développer une approche pour étudier le problème de la vivacité des GE p-temporels. La possibilité de re-formuler le graphe d'événements p-temporel sous la forme matricielle $\mathcal{X} \geq \gamma^1 \mathbf{A}^- \mathcal{X} \oplus \gamma^{-1} \mathbf{A}^+ \mathcal{X}$ nécessite l'introduction d'une condition de vivacité "statique". L'introduction d'une série particulière de matrices aboutit à une seconde approche analysant la vivacité temporelle, en permettant une interprétation condensée du graphe de calcul. Les matrices sont de la dimension du nombre de transitions du GE et sont ainsi de tailles plus réduites. Le calcul est d'autre part itératif et exploite les résultats des calculs précédents. En résumé, la deuxième approche peut être qualifiée de dynamique contrairement à la première. Celle-ci peut cependant traiter des modèles plus complexes comme les GE à flux temporels.

L'estimation d'état et la détection de défaillances est un autre centre d'intérêt. Nous considérons en particulier, les graphes d'événements temporisés et à flux temporels pour les sémantiques Et-Pur et Et-Faible. Le problème d'estimation pour ces derniers a été re-formulé sous forme d'un problème du type point fixe $x \leq f(x)$ avec f une fonction $(\min, \max, +)$. La résolution nous a conduit à introduire un estimateur optimal permettant d'estimer le plus grand état. La particularisation aux GE temporisés nous amène à un second algorithme. D'autre part, la synthèse de cet estimateur nous a conduit à développer une approche pour la détection de défaillances. Cette approche se base sur la vérification de la cohérence entre le modèle et les données qui décrivent ce dernier. L'image de cette cohérence est donnée par deux contraintes relatives aux données, émanant de l'entrée et de la sortie, ce qui constitue un moyen de test en ligne. En effet, la vérification de ces deux contraintes signifie que le modèle suit son fonctionnement nominal. Dans le cas contraire, l'occurrence d'une défaillance s'est traduite par une incohérence entre le modèle et la réalité du processus.

La dernière partie traite le problème de la commande optimale dans les graphes d'événements

temporisés et à flux temporels. Ainsi, la classique approche “Backward” a été analysée. Pour cela, un algorithme spécifique aux GET permettant le calcul de la commande et du co-état les plus grands, a été déduit à partir de l’algorithme général de point fixe MD. Après reformulation, la méthode “Backward” a été comparée ensuite avec l’approche développée. Les deux méthodes donnent le même résultat pour le calcul de la séquence de commande. Ceci montre que la méthode “Backward” ne constitue qu’un cas particulier de résolution du type point fixe [DDA03b]. Dans la démarche proposée, l’équation d’état est vérifiée par le co-état contrairement à la méthode “Backward” de base, dont le co-état ne vérifie qu’une inégalité. Guidée par cette première étude, nous avons ensuite abordé le problème de commande en juste à temps, dans les graphes d’événements à flux temporels. Nous avons d’abord appliqué la théorie spectrale afin de vérifier l’existence d’une commande optimale. Le passage entre les deux formulations non-homogène et homogène a été réalisé grâce à l’opération de relaxation ou d’homogénéisation, que nous avons introduite dans le chapitre 4. La borne supérieure de l’ensemble des solutions ainsi qu’une proposition analysant l’espace des solutions, a été introduite grâce à la commandabilité structurelle dans ces graphes. Enfin, un algorithme de calcul de commande optimale a été élaboré. La démarche proposée peut rapidement être appliquée sur d’autres problèmes, comme la commande optimale de graphes d’événements avec spécifications définies, par exemple avec un graphe d’événements p-temporel.

Perspectives

Une perspective est le développement d’une commande en ligne s’inspirant d’approches de l’automatique classique en algèbre usuelle. Dans ce domaine, le comparateur est probablement l’un des composants les plus importants car il touche au fonctionnement profond de la boucle fermée qui est basée sur l’idée de comparaison et qui est relativement simple : par exemple, dans une régulation de vitesse d’une voiture, lorsque cette vitesse est trop faible par rapport à la consigne (une montée), le comparateur indiquera qu’une action est nécessaire (ici une accélération). Inversement, si la vitesse est trop forte (une descente), le comparateur demandera le ralentissement du véhicule. Enfin, lorsque la vitesse correspond à la consigne, le véhicule aura un comportement optimal.

Maintenant, si nous transposons dans l’algèbre des fonctions topicales, une commande peut de manière similaire être définie où les dates des sorties sont les plus grandes possibles en fonction des contraintes et en particulier d’une consigne qui correspond aux dates désirées. Cette commande peut être optimale dans le cas où le système est connu avec précision. Cependant, le système peut subir une perturbation comme la variation d’une temporisation, qui peut être soit une augmentation, soit une diminution de durée. Dans ce cas, afin que l’objectif de “juste à temps” soit respecté et ainsi converger vers l’optimalité, le système de commande, comme en

automatique classique, devra accélérer ou ralentir selon les perturbations. Lorsque celles-ci disparaîtront, le système de commande gèrera le processus de nouveau, de manière optimale. Ainsi, par exemple, un graphe d'événements temporisé sera de nouveau commandé par l'approche "Backward".

En ce sens, la démarche proposée par B. De Schutter et T. van den Boom qui s'inspire de la commande à prédiction basée sur le modèle (Model Predictive Control en anglais) en automatique classique, est une réponse pertinente, car elle s'appuie sur un critère d'optimalité à minimiser qui pénalise les retards, et est maximale, lorsque le fonctionnement est en "Juste à temps". Le critère scalaire est composé d'un critère relatif à l'entrée et d'un critère relatif à la sortie. Ce dernier représente une erreur de poursuite où la somme des retards par rapport aux dates de sortie désirées est nulle lorsqu'il n'y a pas de retard. Le critère relatif aux entrées maximise celles-ci. La résolution du problème les conduit à une commande prédictive à fenêtre glissante qui gère aussi bien le cas "trop tôt" que "trop tard" ce qui en fait un outil performant. En effet, l'étude du critère montre qu'elle généralise l'approche dite "Backward" basée sur la résiduation.

Comme les systèmes considérés sont du type graphe d'événements temporisé, une perspective émerge naturellement, c'est celle de la généralisation de l'approche prédictive à des modèles plus complexes, p-temporel, arcs temporels, ... tout en gardant un critère d'optimalité. L'objectif est alors d'intégrer les différents éléments nécessaires à une commande prédictive, c'est à dire l'estimation optimale de l'état courant, la commande optimale, ... Considérant ces différents éléments, ce mémoire participe ainsi au développement de cet objectif.

Bibliographie

- [BCOQ92] F. Baccelli, G. Cohen, G.J. Olsder, and J.P. Quadrat. *Synchronization and linearity. An Algebra for Discrete Event Systems*. Wiley, New York, 1992.
- [BD91] B. Berthomieu and M. Diaz. Modeling and verification of time dependent systems using time Petri nets. *IEEE Trans. on Software Eng*, 17(N°3):259–273, 1991.
- [Bir40] G. Birkhoff. *Lattice theory*. Number XXV. Providence, Rhode Island, 1940.
- [BJ72] T.-S. Blyth and M.-F. Janowitz. *Residuation Theory*. Pergamon press, 1972.
- [BM03] Thierry Bousch and Jean Mairesse. *Fonctions topicales à portée finie et fonctions uniformément topicales*. Cours dispensé à l'Université de Paris sud, 2003.
- [Bon05] P. Bonhomme. Control of time critical systems using partial orders. In *proceedings of IMACS*, Paris, France, 2005.
- [Boy01] M. Boyer. *Contribution à la modélisation des systèmes à temps contraint et application au multimédia*. Thèse, Université Paul Sabatier, Juillet 2001.
- [Bra83] G.-W. Brams. *Réseaux de Petri : théorie et pratique*. Masson, Paris, France, 1983.
- [CDY02] S. Chakraborty, D.L. Dill, and K.Y. Yun. Efficient algorithms for approximate time separation of events. *Sadhana*, 27, Part 2:129–162, April 2002.
- [CG79] R.-A. Cuninghame-Green. *Minimax Algebra*. Number 166 in Lecture notes in Economics and Mathematical Systems. Springer, 1979.
- [CGB03] R.A. Cuninghame-Green and P. Butkovic. The equation $ax=by$ over $(\max, +)$. *Theoretical Computer Science*, 293:3–12, 2003.
- [CGQ93] G. Cohen, S. Gaubert, and J. P. Quadrat. From first to second-order theory of linear discrete event systems. In *12th IFAC*, Sydney, Jul. 1993.
- [CGQ98] G. Cohen, S. Gaubert, and J.-P. Quadrat. Max-plus algebra and system theory : Where we are and where to go now. In *IFAC Conference on System Structure and Control*, Nantes, France, 1998.

- [CHBF01] B. Cottenceau, L. Hardouin, J.-L. Boimond, and J.-L. Ferrier. Model Reference Control for Timed Event Graphs in Dioid. *Automatica*, 37:1451–1458, August 2001.
- [Chr83] P. Chrétienne. *Les Réseaux de Petri Temporisés*. Thèse Université Pierre et Marie Curie (Paris VI), Paris, 1983.
- [CMS99] A. Cerone and A. Maggiolo-Schettini. Time-base expressivity of time petri nets for system specification. *Theoretical Computer Science*, 216(1-2):1–53, March 1999.
- [Cot99] B. Cottenceau. *Contribution à la commande de systèmes à événements discrets : synthèse de correcteurs pour les graphes d'événements temporisés dans les dioïdes*. Thèse, LISA - Université d'Angers, 1999.
- [CT80] M.G. Crandall and L. Tartar. Some relations between non expansive and order preserving maps. *In Proceedings of the AMS*, 78(3):385–390, 1980.
- [CT01] J. Cochet-Terrasson. *Algorithme d'itération sur les politiques pour les applications monotones contractantes*. Thèse, Ecole Nationale Supérieure des Mines de Paris, Décembre 2001.
- [CTCG+98] J. Cochet-Terrasson, G. Cohen, S. Gaubert, M. McGettrick, and J.P. Quadrat. Numerical computation of spectral elements in max-plus algebra. *In IFAC Conference on System Structure and Control*, Nantes, France, 1998.
- [CTGG99] J. Cochet-Terrasson, S. Gaubert, and J. Gunawardena. A constructive fixed point theorem for min-max functions. *Dynamics and Stability of systems*, 14(4):407–433, 1999.
- [CZ02] Y. Cheng and D-Z. Zheng. On the cycle time of non-autonomous min-max systems. *In Proceedings Workshop On Discrete Event Systems WODES*, Zaragoza, Spain, October 2002.
- [CZ04] Y. Cheng and D-Z. Zheng. A cycle time computing algorithm and its application in the structural analysis of min-max systems. *Discrete Event Dynamic Systems: Theory and Applications*, 14:5–30, 2004.
- [CZ05] Y. Cheng and D-Z. Zheng. Min-max inequalities and the timing verification problem with max and linear constraints. *Discrete Event Dynamic Systems : Theory and Applications*, 15:119–143, 2005.
- [DA97] R. David and H. Alla. *Du Grafcet aux réseaux de Petri*. Hermès, Paris, 1997.
- [DA04] R. David and H. Alla. *Discrete, Continuous, and Hybrid Petri Nets*. Springer-Verlag, Berlin, 2004.

- [DAD03] M.K. Didi-Alaoui and P. Declerck. Control synthesis in $(\max, +)$ systems. In *Proceedings of the 9th IEEE International Conference ETFA*, Lisbon, Portugal, 2003.
- [DAD04] M.K. Didi-Alaoui and P. Declerck. From dioid algebra to p-time event graphs. In *Proceedings of the ICINCO*, Setubal, Portugal, 2004.
- [DAJ92] A. Desrochers and R. Y. Al-Jaar. *Applications of Petri nets in manufacturing systems*. IEEE Press, 1992.
- [DDA02] P. Declerck and M.K. Didi-Alaoui. Estimation d'état par l'étoile de kleene. In *Proceedings of CIFA'2002*, Nantes, France, 2002.
- [DDA03a] P. Declerck and M.K. Didi-Alaoui. Constraint propagation for max-plus-linear discrete event systems: application to the state estimation. In *Proceedings of the IMACS/IEEE Multiconference CESA*, Lille, France, 2003.
- [DDA03b] P. Declerck and M.K. Didi-Alaoui. Synthèse de commande des systèmes dynamiques à événements discrets. In *Proceedings of the MSR*, Metz, France, 2003.
- [DDA04] P. Declerck and M.K. Didi-Alaoui. Modelling and liveness analysis of p-time event graphs in the $(\min, \max, +)$ algebra. In *Proceedings of the International Conference on SMC*, Hague, Netherlands, 2004.
- [DDA05a] P. Declerck and M.K. Didi-Alaoui. Extremal trajectories in p-time event graphs. application to control synthesis with specifications. In *Proceedings of The CDC-ECC*, Seville, Spain, 2005.
- [DDA05b] P. Declerck and M.K. Didi-Alaoui. Optimal control synthesis in interval descriptor systems application to time stream event graphs. In *Proceedings of the 16th IFAC World Congress*, Prague, Czech Republic, 2005.
- [DDA05c] P. Declerck and M.K. Didi-Alaoui. State estimation in time event graphs. application to fault detection. In *Proceedings of the 17th IMACS World Congress*, Paris, France, 2005.
- [DDC05] J.P. Denat, S.C. Dutilleul, and F. Chetouane. Cycle time calculation for electroplating lines steady state operation using p-time strongly connected event graphs. In *Proceedings of IMACS*, Paris, France, 2005.
- [Dia01] M. Diaz. *Les réseaux de Petri : Modèles fondamentaux*. Hermès, Paris, France, 2001.
- [DP02] B.A. Davey and H.A. Priestley. *Introduction to lattices and order*. Cambridge University Press, Cambridge, UK, 2002.

- [DS93] M. Diaz and P. S enac. Time stream Petri nets: A model for timed multimedia streams synchronisation. In *Proceedings of the MultiMedia Modeling'93*, Singapore, November 1993.
- [Gau92] S. Gaubert. *Th eorie des Syst emes Lin eaires dans les Dio ides*. Th ese,  cole des Mines de Paris, Juillet 1992.
- [Gav02] Martin Gavalec. Monotone eigenspace structure in max-min algebra. *Linear Algebra and its Applications*, 345:149–167, 2002.
- [Gaz97] M. J. Gazarik. *Monitoring and control of manufacturing systems based on the Max-plus formulation*. Ph.D. thesis, Georgia Tech, Septembre 1997.
- [Ger88] J.J. Gertler. Survey on model-based failure detection and isolation in complex plants. *IEEE Control System Magazine*, pages 3–11, 1988.
- [GG98a] S. Gaubert and J. Gunawardena. The duality theorem for min-max functions. *IEEE Trans. on Automatic Control, C. R Acad mie des Sciences/ Elsevier, Paris*, t. 326 S erie I:43–48, 1998.
- [GG98b] S. Gaubert and J. Gunawardena. A non-linear hierarchy for discrete event dynamical systems. In *Proceedings of WODES'98*, Cagliari, Italy, August 1998. IEEE.
- [GG99] S. Gaubert and J. Gunawardena. Existence of eigenvectors for monotone homogeneous functions. In *Technical Report HPL-BRIMS-98-99*, Hewlett-Packard Labs, August 1999.
- [GK95] J. Gunawardena and M. Keane. On the existence of cycle time for some nonexpansive maps. In *Technical Report HPL-BRIMS-95-003*. Hewlett-Packard Labs, 1995.
- [GM01] M. Gondran and M. Minoux. *Graphes, dio ides et semi-anneaux*. Tec & Doc, Paris, France, 2001.
- [Gun94a] J. Gunawardena. Cycle time and fixed points of min-max functions. In *G. Cohen and J.P. Quadrat, editors, 11th International Conference on analysis and optimization of systems, Springer LNCIS 199*, pages 266–272, 1994.
- [Gun94b] J. Gunawardena. Min-max function. In *Discret Event Dynamic Systems*,, volume 4, pages 377–406, 1994.
- [Gun03] J. Gunawardena. From max-plus algebra to nonexpansive mappings: a nonlinear theory for discrete event systems. Number 293, pages 141–167. *Theoretical Computer Science*, 2003.

- [Har04] L. Hardouin. *Sur la commande linéaire de systèmes à événements discrets dans l'algèbre (max, +)*. Habilitation à diriger des recherches, ISTIA, Université d'Angers, Juin 2004.
- [KDCD96] W. Khansa, J. P. Denat, and S. Collart-Dutilleul. p-time petri nets for manufacturing systems. In *proceedings of WODES'96*, pages 94–102, Edimburgh, UK, 1996.
- [KG94] R. Kumar and V.K. Garg. Extremal solutions of inequations over lattices with applications to supervisory control. In *Proceedings of the 33rd Conference on Decision and Control*, pages 3636–3641, Lake Buena Vista, Florida, USA, December 1994.
- [KG95] R. Kumar and V.K. Garg. *Modeling and Control of Logical Discrete Event System*. Kluwer Academic Publishers, USA, 1995.
- [Kha97] W. Khansa. *Réseaux de Petri p-temporels: contribution à l'étude des systèmes à événements discrets*. Thèse, Université de Savoie, Annecy, Mars 1997.
- [Lho03] M. Lhommeau. *Étude de systèmes à événements discrets dans l'algèbre (max, +) : - Synthèse de correcteurs robustes dans un dioïde d'intervalles. - Synthèse de correcteurs en présence de perturbations*. Thèse, LISA - Université d'Angers, 2003.
- [Lib96] L. Libeaut. *Sur l'utilisation des dioïdes pour la commande des systèmes à événements discrets*. Thèse, Ecole centrale de Nantes, 1996.
- [MD92] K. Mc Millan and D. Dill. Algorithms for interface timing verification. In *Proceedings of the IEEE, International Conference on Computer Design: VLSI in Computers and Processors*, 1992.
- [Men97] E. Menguy. *Contribution à la commande des systèmes linéaires dans les dioïdes*. Thèse, LISA - Université d'Angers, 1997.
- [Mer74] P. Merlin. *A study of the Recoverability of Computer System*. Ph.D. thesis, Dep. Comput. Sci., Univ. California, Irvine, 1974.
- [MF76] P. Merlin and D. J. Faber. Recoverability of communication protocols. *IEEE Transactions on Communications*, COM-24(9), Septembre 1976.
- [MS69] M. Menon and H. Schneider. The spectrum of an operator associated with a matrix. *Linear Algebra and its applications*, (2):321–334, 1969.
- [Mur89] T. Murata. Petri nets : Properties, analysis and applications. *IEEE Proceedings : Special issue on Discrete Event Systems*, 77(4):541–581, 1989.

- [Ols91a] G.J. Olsder. Descriptor systems in the max-min algebra. In *Proceedings of the European Control Conference*, pages 1825–1830, Grenoble, France, 1991.
- [Ols91b] G.J. Olsder. Eigenvalues of dynamic min-max systems. *Journal of Discrete Event Dynamic Systems*, (1):177–207, 1991.
- [Ols93] G.J. Olsder. Analyse de systèmes min-max. Rapport de recherche. Institut National de Recherche en Informatique et en Automatique 1904, 1993.
- [Pet62] C.-A. Petri. *Kommunikation mit Automaten*. Thèse, Institut für Instrumentelle Mathematik, Bonn, Germany, 1962.
- [Pop91] J. Popova. On time Petri nets. *J. Inform. Process. Cybern*, pages 227–244, 1991.
- [PW97] J. M. Prou and E. Wagneur. Controllability in the max algebra. 1997.
- [PX95] J. M. Proth and X. Xie. *Les réseaux de Petri pour la conception et la gestion des systèmes de production*. Masson, Paris, 1995.
- [Ram74] C. Ramchandani. *Analysis of Asynchronous Concurrent Systems by Timed Petri Nets*. Ph.D. thesis, MIT, Cambridge, 1974.
- [Sif77] J. Sifakis. Use of Petri nets for performances evaluation. In *Measuring Modeling and Evaluation Computer Systems*. E. Beilner and E. Gelenbe, 1977.
- [Sif78] J. Sifakis. Structural properties of Petri nets. In *Mathematical foundation of computer science*, volume 64. J. Windowski (ed), Springer Verlag, 1978.
- [Sif79] J. Sifakis. Performance evaluation of systems using nets. In *Net Theory and Application, Proceeding of the Advanced Course on General net Theory of Processes and Systems*, volume 84 of LNCS, pages 307–319, Hamburg, 1979. G. Goos and J. Harmanis.
- [Sén96] P. Sénac. *Contribution à la modélisation des systèmes multimédias et hypermédias*. thèse, Université Paul Sabatier, Juin 1996.
- [Tha04] K. Thamani. Modélisation et résolution des systèmes min-max-plus. Rapport de stage de DEA, DEA Automatique et Informatique Appliquée, Ecole Centrale de Nantes, LISA, ISTIA, France, 2004.
- [Wal83] B. Walter. Timed net for modeling and analysing protocols with time. In *Proceedings of the IFIP Conference on Protocol Specification Testing and Verification*, North-Holland, 1983.
- [Wal95] E. Walkup. *Optimization of linear max-plus systems with application to timing analysis*. Ph.D. thesis, university of Washington, 1995.

- [WB98] E. Walkup and G. Borriello. *A general linear max-plus solution technique*. Idempotency, Jeremy Gunawardena, 1998.
- [Wil76] A.S. Wilsky. A survey of several failure detection method. *Automatica*, pages 601–611, 1976.
- [Wou01] J. Van Der Woude. A characterisation of the eigenvalue of a general (min, max, +)-system. *Discrete Event Dynamic Systems : Theory and Applications*, 11:203–210, 2001.
- [WS03] J. Van Der Woude and Subiono. Conditions for the structural existence of an eigenvalue of a bipartite (min, max, +)-system. *Theoretical Computer Science*, 293:13–24, 2003.