



HAL
open science

**Contributions à la commande des systèmes
(max,+)-linéaires. Applications aux réseaux de
transport.**

Laurent Houssin

► **To cite this version:**

Laurent Houssin. Contributions à la commande des systèmes (max,+)-linéaires. Applications aux réseaux de transport.. Automatique / Robotique. Université d'Angers, 2006. Français. NNT : . tel-00143884

HAL Id: tel-00143884

<https://theses.hal.science/tel-00143884>

Submitted on 27 Apr 2007

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

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

**Contribution à la commande des systèmes
($max, +$)-linéaires.
Applications aux réseaux de transport.**

THÈSE DE DOCTORAT

Spécialité : Automatique et Informatique Appliquée

ÉCOLE DOCTORALE d'Angers

soutenue le

le 12 décembre 2006

à l'ISTIA - Université d'Angers

par Laurent Houssin

devant le jury ci-dessous :

Abdellah El Moudni	Président du jury	<i>Professeur, UTBM, Belfort-Montbéliard</i>
Jean-Pierre Quadrat	Rapporteur	<i>Directeur de Recherche, INRIA, Rocquencourt</i>
Jean-Jacques Loiseau	Rapporteur	<i>Directeur de Recherche, IRCCyN, Nantes</i>
Abdellah El Moudni	Examineur	<i>Professeur, UTBM, Belfort-Montbéliard</i>
Jean-Louis Ferrier	Examineur	<i>Professeur, LISA, Université d'Angers</i>
Jean-Louis Boimond	Examineur	<i>Professeur, LISA, Université d'Angers</i>
Sébastien Lahaye	Examineur	<i>Maître de Conférences, LISA, Université d'Angers</i>

Directeurs de thèse :

Jean-Louis Boimond - Sébastien Lahaye

Remerciements

Je remercie J.-L. Ferrier, directeur du Laboratoire d'ingénierie des Systèmes Automatisés, de m'avoir accueilli depuis le DEA au sein de son laboratoire. L'ambiance du LISA est une agréable incitation à s'intéresser à tous les aspects des Sciences et Techniques.

J'exprime toute ma reconnaissance à J.-J. Loiseau, Directeur de recherche CNRS à l'IRC-CyN, et J.-P. Quadrat, Directeur de recherche INRIA Rocquencourt, pour l'honneur qu'ils m'ont fait en acceptant de rapporter mon travail.

Je voudrais également remercier A. El Moudni, Professeur à l'Université de Technologie de Belfort-Montbéliard et, de nouveau, J.-L. Ferrier, Professeur à l'Institut des Sciences et Techniques de l'Ingénieur d'Angers, de l'intérêt qu'ils ont porté à mes travaux en acceptant de participer au jury.

Je tiens à exprimer particulièrement ma gratitude envers S. Lahaye, co-directeur de thèse, pour sa disponibilité, son intuition scientifique et ses nombreux conseils. La confiance qu'il m'a accordée, depuis qu'il m'a offert l'opportunité d'effectuer cette thèse de doctorat sous sa responsabilité, m'a toujours beaucoup honoré.

J'adresse également mes plus vifs remerciements à J.-L. Boimond, directeur de thèse, pour m'avoir donné l'opportunité de préparer une thèse à ses côtés. L'intérêt qu'il a porté à mes travaux et ses fréquents encouragements ont été des atouts majeurs dans l'efficacité et l'entrain de ce travail.

Mes remerciements vont également à tous les membres du LISA pour leur enthousiasme, leur bonne humeur quotidienne et leur disponibilité. Les doctorants (et docteurs) du bureau E37 : Sébastien, Nicolas (membres d'une improbable équipe de triathlon au défi des étudiants 2005), Xavier, Michel, Samir,... ont notamment su créer une émulation scientifique importante et une ambiance chaleureuse, je les remercie pour les bons moments passés durant ces années.

Durant ces trois années, j'ai rencontré des amis remarquables, doctorants de l'université d'Angers pour la plupart. Ces rencontres ont accouché d'amitiés sincères. Qu'ils trouvent ici

l'expression de ma reconnaissance.

Je remercie très chaleureusement mes parents et mes frères pour le soutien considérable qu'ils m'ont apporté et la confiance dont ils m'ont toujours témoigné.

Enfin et surtout, mille fois merci à Daphné.

Table des matières

Table des figures	vii
Introduction	1
1 Outils algébriques	5
1.1 Ensembles ordonnés	6
1.1.1 Relation d'ordre et structures ordonnées	6
1.1.2 Demi-treillis et treillis	8
1.2 Dioïdes	9
1.2.1 Dioïdes et relation d'ordre	9
1.2.2 Dioïdes et treillis	12
1.3 Applications définies sur des dioïdes	14
1.4 Propriétés spectrales des matrices définies dans un dioïde	16
1.5 Théorie de la résiduation	19
1.6 Points fixes d'applications définies sur un dioïde	23
1.6.1 Applications isotones	23
1.6.2 Applications antitones	26
2 Systèmes $(max, +)$-linéaires	33
2.1 Mise en équations de graphes d'événements temporisés	34
2.1.1 Graphes d'événements temporisés	34

2.1.2	Deux représentations analytiques des graphes d'événements temporisés	34
2.2	Représentation d'état	36
2.2.1	Forme ARMA, forme d'état	36
2.2.2	Conditions initiales canoniques	39
2.2.3	Systèmes autonomes $(max, +)$ -linéaires	41
2.3	Réponse impulsionnelle	43
2.4	Représentation en séries formelles	45
2.4.1	Le dioïde $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$	45
2.4.2	Manipulation des séries formelles	48
2.4.3	Modélisation des graphes d'événements temporisés dans le dioïde des séries formelles	55
2.5	Commande de systèmes $(max, +)$ -linéaires	57
2.5.1	Critères	57
2.5.2	Objectifs de commande	59
2.5.3	Structures de commande et synthèse	60
3	Contribution à la commande des systèmes $(max, +)$-linéaires	67
3.1	Originalités des commandes proposées	68
3.1.1	Un nouvel objectif de commande	68
3.1.2	Un nouveau critère	74
3.2	Commande en juste-à-temps de systèmes $(max, +)$ -linéaires sous contraintes	75
3.2.1	Conditions terminales	75
3.2.2	Contraintes	76
3.2.3	Critère	77
3.2.4	Synthèse	77
3.2.5	Exemple	81
3.3	Commande de systèmes $(max, +)$ -linéaires contraints en minimisant les retards	82

3.3.1	Formalisation du problème	82
3.3.2	Correction par retour d'état sur l'état	84
3.3.3	Exemple	86
3.3.4	Correction par retour d'état sur l'état : restriction à un ensemble de variables	88
3.3.5	Correction par retour d'état sur l'entrée	89
4	Applications aux réseaux de transport	93
4.1	Généralités	94
4.1.1	Infrastructure	94
4.1.2	Entités de transport	95
4.2	Pratiques d'exploitation	96
4.2.1	Tableau de marche	96
4.2.2	Régulation	97
4.3	Modélisation d'un système de transport	98
4.4	Stabilité	101
4.5	Génération de tables d'horaires	104
4.5.1	Présentation du problème	104
4.5.2	Résolution	107
4.5.3	Discussion et remarques	107
4.5.4	Exemple	108
4.6	Une ébauche pour la régulation d'un nœud ferroviaire	115
	Conclusion	117
	Annexes	118
A	Détails des exemples	119
A.1	Exemple 3.2.5, p.81	119

A.2 Exemple 3.3.3, p.86	121
Bibliographie	125

Table des figures

1.1	Diagramme de Hasse représentant les ensembles $\mathcal{Q}_f, \mathcal{P}_f$ et \mathcal{F}_f d'une application isotone.	25
1.2	Diagramme de Hasse de l'ensemble \mathcal{D}	28
1.3	Diagramme de Hasse représentant les ensembles $\mathcal{Q}_g, \mathcal{P}_g, \mathcal{F}_g$ et \mathcal{F}_{g^2} d'une application antitone g	31
2.1	Exemple de graphe d'événements temporisés.	34
2.2	Exemple de graphe d'événements temporisés possédant une représentation d'état linéaire.	37
2.3	Représentation graphique de la suite $x(n, t)$ considérée dans l'exemple 14.	46
2.4	Représentation graphique des opérations sur deux monômes.	50
2.5	Correction d'un système par un precompensateur.	61
2.6	(a) : Correction d'un système par un retour de sortie. (b) : Correction d'un système par un retour d'état.	62
2.7	(a) : Correction d'un système par un retour d'état sur l'état. (b) : Correction d'un système partiellement observable et partiellement commandable par un retour d'état sur l'état.	63
2.8	Correction d'un système par un précompensateur et un retour de sortie.	65
3.1	Schéma d'un chemin de GET.	70
3.2	Illustration des termes mis en jeu dans l'équation (3.11).	74
3.3	Exemple de série x_i contrainte dans l'intervalle $[k_{d_i}, k_{f_i}]$	77
3.4	Exemple de série x_i soumise à la contrainte.	80

3.5	Exemple de GET.	81
3.6	Exemple de GET avec une réalisation de son correcteur.	87
4.1	Réseau de transport élémentaire composé de deux lignes.	99
4.2	Exemple d'absorption de retard par un réseau stable, $\bullet : u(k)$, $\times : x(k)$	102
4.3	Trois lignes du réseau de bus de la ville d'Angers.	109

Introduction

Les *Systèmes (Dynamiques) à Événements Discrets* (SED) désignent des systèmes généralement de conception humaine. Leur évolution obéit à l'apparition d'événements qui ont lieu à des instants discrets. Les systèmes de production, les réseaux de transport et les systèmes informatiques sont, par exemple, des processus que l'on peut considérer comme des systèmes à événements discrets.

Les systèmes "naturels", c'est-à-dire ceux qui obéissent aux lois de la Physique, peuvent être modélisés par des équations différentielles qui sont dépendantes du temps. Au contraire, la diversité des systèmes à événements discrets, due à la nature des phénomènes qui entrent en jeu (notamment des phénomènes de synchronisation ou d'exclusion), a jusque-là conduit à des modèles différents mais complémentaires. La *théorie des langages et des automates* permet d'obtenir des modèles adaptés pour la description et la commande des systèmes uniquement concernés par les phénomènes de concurrence. Les systèmes qui mettent en jeu des phénomènes de synchronisation (mais pas de concurrence), peuvent quant à eux être appréhendés à travers la *théorie des systèmes linéaires dans les dioïdes*. Les *réseaux de Petri* proposent un formalisme qui permet de décrire des systèmes qui comportent à la fois des phénomènes de concurrence et de synchronisation.

La classe des SED que nous étudions ici est celle qui met en jeu des phénomènes de synchronisation. Pour un tel processus, on peut obtenir un modèle mathématique sous la forme d'équations de récurrence qui utilisent des opérateurs *max* ou *min*, et qui sont donc non linéaires. Mais traduite dans une structure algébrique particulière, appelée dioïde, cette représentation devient linéaire. La représentation est alors dite *(max, +)-linéaire*, dans le cas du dioïde $(max, +)$. L'algèbre des dioïdes est donc apparue comme la structure mathématique adéquate pour modéliser et étudier cette classe de systèmes, et au début des années 80, une théorie des systèmes linéaires dans les dioïdes a vu le jour. En particulier, des efforts ont été réalisés pour que cette théorie "colle" à la théorie conventionnelle de l'Automatique. On y retrouve notamment des similitudes dans la représentation des systèmes : forme d'état, réponse impulsionnelle, transformation des signaux comparable à la transformée en z , et dans l'analyse de ceux-ci : rôle du spectre de la matrice d'évolution.

Toujours par analogie avec la théorie conventionnelle de l'Automatique, de nombreux travaux sur la commande des systèmes $(max, +)$ -linéaires ont également été développés. Ceux-ci, essentiellement destinés à être appliqués aux systèmes de production, visent toujours à piloter les systèmes en juste-à-temps. En ce sens, ces études peuvent être vues comme des contributions aux politiques de gestion de production en flux tendus. De plus, les lois de commande proposées jusqu'ici se basent principalement sur une approche entrée-sortie des systèmes : on se focalise essentiellement sur les flux de produits en entrée et en sortie du système de production.

Le travail présenté dans ce mémoire se situe dans la continuité de ces travaux d'analyse et de commande. La motivation initiale de cette étude a été plus spécifiquement de contribuer à l'étude des réseaux de transport en les appréhendant comme des systèmes $(max, +)$ -linéaires. Plusieurs spécificités de ces systèmes ont conduit à des développements se démarquant de la littérature. En premier lieu, il s'est avéré impossible de traduire au travers de relations entrée-sortie certains aspects de la dynamique de ces systèmes. Cela nous a amené à considérer un nouvel objectif de commande sous la forme de contraintes sur les variables internes (variables d'état) du système qui traduisent chacune un aspect de la dynamique à imposer au système de transport. D'autre part, le critère de juste-à-temps n'est pas forcément le plus pertinent pour la commande des systèmes de transport. En effet, l'objectif est souvent de ralentir le moins possible les véhicules au sein d'un réseau. Pour le transport de passagers par exemple, la qualité de la commande s'évalue en grande partie au travers de la qualité de service perçue par les usagers, et cette dernière est d'autant meilleure que les véhicules sont le moins souvent et le moins longtemps immobilisés au sein du réseau. Plus généralement, afin de limiter les coûts d'exploitation et/ou l'encombrement du réseau, on cherche souvent à minimiser le nombre de véhicules en transit au sein de celui-ci. Ces considérations nous ont amené à introduire un critère original pour la commande des systèmes $(max, +)$ -linéaires. On s'est en effet intéressé à la synthèse de la loi de commande induisant le moins possible de retards au sein du système (à l'opposé du juste-à-temps qui vise à retarder au maximum l'évolution du système).

L'organisation de ce mémoire est la suivante :

- Les outils algébriques nécessaires à l'étude des SED dans les dioïdes sont décrits dans le premier chapitre. La théorie des dioïdes repose essentiellement sur celle des ensembles ordonnés, c'est pourquoi on retrouve des outils destinés, à l'origine, aux treillis. Nous mettons tout d'abord en évidence les relations entre les treillis et les dioïdes. Puis, nous analysons les propriétés des matrices à coefficients dans un dioïde. La théorie de la résiduation, qui fournit une réponse au problème d'inversion d'application sur les ensembles ordonnés, est présentée. Enfin, l'étude des points fixes d'applications définies sur des treillis a également bénéficié à l'étude de celles définies sur des dioïdes. Les résultats pour les applications isotones sont rappelés. Des développements, à notre connaissance originaux, sont proposés sur les points fixes d'applications antitones.

- Le second chapitre, essentiellement bibliographique, propose un survol de la modélisation et de la commande des systèmes décrits dans les dioïdes. On présente la mise en équations d'une sous classe de réseau de Petri adaptée à la modélisation des SED considérés ici. A partir de ces équations, une représentation d'état linéaire peut être obtenue. Le problème des conditions initiales est également abordé, ainsi que la représentation entrées-sorties. A l'image de la transformée en z des systèmes échantillonnés, les systèmes $(max, +)$ -linéaires sont dotés de la transformée en γ, δ qui permet de représenter les systèmes par leur fonction de transfert. La fin du chapitre est constituée d'un état de l'art, non exhaustif, sur la commande des systèmes $(max, +)$ -linéaires.
- Le troisième chapitre est consacré à notre contribution à la commande des systèmes $(max, +)$ -linéaires. Nous considérons un ensemble de contraintes sur l'état du système qui constitue un nouvel objectif de commande à atteindre. Dans un premier temps, nous cherchons à synthétiser la plus grande commande (*i.e.* celle qui est optimale vis-à-vis du critère du juste-à-temps) de sorte que l'état induit respecte les contraintes définies au préalable. Puis, dans un second temps, nous nous intéressons à un autre problème original dans la théorie des systèmes $(max, +)$ -linéaires qui est la synthèse d'un correcteur en boucle fermée permettant de satisfaire des spécifications exprimées sous la forme de contrainte sur l'état, tout en ralentissant le moins possible le transfert du système.
- Dans le quatrième chapitre, on aborde l'étude des réseaux de transport en tant que SED. On rappelle tout d'abord le fonctionnement de ces systèmes particuliers, puis leur mise en équations dans le dioïde $(max, +)$. Une caractéristique importante dans l'évaluation des réseaux de transport est la stabilité. Elle correspond à la capacité d'un réseau à reprendre son fonctionnement normal après l'apparition d'un retard. Nous rappelons la condition de stabilité de systèmes à correspondances totales puis nous généralisons cette condition aux réseaux à correspondances partielles. Le problème du calcul des tables d'horaires d'un réseau est abordé comme un problème de commande de système $(max, +)$ -linéaire. La commande en juste-à-temps exposée au chapitre 3 permet de résoudre ce problème. Une application au calcul des tables d'horaires d'une partie du réseau de bus de la ville d'Angers. Nous terminons le chapitre par une ébauche d'application de la commande minimisant les retards, exposée dans le chapitre 3, aux réseaux de transport. L'exemple d'un nœud ferroviaire est considéré.

Chapitre 1

Outils algébriques

Nous rassemblons ici un certain nombre de notions utiles à la compréhension de la structure algébrique que l'on utilisera par la suite, et plus particulièrement la notion d'*ordre*. Après un bref rappel sur les ensembles ordonnés, on présente cette structure algébrique particulière : les *dioïdes*. Les dioïdes, également appelés semi-anneaux idempotents, sont des ensembles naturellement ordonnés par la loi additive, notée \oplus , et plus précisément des treillis.

Les matrices à coefficients dans un dioïde présentent certaines propriétés intéressantes dans le cadre de l'étude des systèmes dynamiques à événements discrets. On formulera quelques rappels sur les propriétés de ces matrices.

Les applications définies sur un dioïde sont rarement inversibles. Cependant la *théorie de la résiduation* fournit une réponse alternative à ce problème d'inversion d'applications. En effet, la structure de treillis des dioïdes leur confère la possibilité pour une application isotone f d'établir la plus grande solution de $f(x) \preceq b$, ou la plus petite solution de $f(x) \succeq b$ pour tout b . Nous exposerons dans ce chapitre les bases de la théorie de la résiduation.

L'étude des points fixes d'applications définies sur des treillis a également fait l'objet de plusieurs recherches qui bénéficient à la théorie des dioïdes. Nous présentons des rappels classiques concernant la théorie des points fixes d'applications isotones puis nous introduisons certains résultats originaux concernant les applications antitones. En effet, s'il est difficile de décrire l'ensemble des points fixes d'une application antitone, nous mettons néanmoins à jour certaines propriétés relatives à la structure de cet ensemble.

1.1 Ensembles ordonnés

On rappelle dans cette section certaines définitions et propriétés de base relatives aux ensembles ordonnés et notamment aux treillis.

1.1.1 Relation d'ordre et structures ordonnées

Sur les entiers, on peut définir les deux relations suivantes :

$$\begin{aligned} a \leq b & : \text{ relation d'ordre habituelle} \\ a|b & : a \text{ divise } b. \end{aligned}$$

Ces deux relations présentent des propriétés communes :

réflexivité $a \leq a$ et $a|a$

antisymétrie $\left\{ \begin{array}{l} \text{Si } a \leq b \text{ et } b \leq a \text{ alors } a = b \\ \text{Si } a|b \text{ et } b|a \text{ alors } a = b \end{array} \right.$

transitivité $\left\{ \begin{array}{l} \text{Si } a \leq b \text{ et } b \leq c \text{ alors } a \leq c \\ \text{Si } a|b \text{ et } b|c \text{ alors } a|c \end{array} \right.$

On dit alors que les relations \leq et $|$ sont des relations d'*ordre*.

Définition 1 (Ensemble ordonné) *Un ensemble muni d'une relation d'ordre est dit ordonné.*

L'ensemble E muni de la relation d'ordre \preceq forme l'ensemble ordonné (E, \preceq) . Cet ensemble est dit *totalelement ordonné* si \preceq est une relation d'ordre totale, c'est-à-dire si deux éléments quelconques x et y de E sont comparables ($x \preceq y$ ou $y \preceq x$). Dans le cas contraire, l'ordre est partiel et l'ensemble est dit *partiellement ordonné* ("Partially Ordered Set" ou "Poset" en anglais). On dira que deux éléments x et y de E sont *incomparables*, noté $x||y$, s'ils vérifient $x \not\preceq y$ et $y \not\preceq x$.

Un ensemble totalement ordonné est aussi appelé une *chaîne* tandis qu'un ensemble composé d'éléments incomparables entre eux est appelé une *antichaîne*.

Exemple 1 (Ensembles ordonnés) *On présente ici quelques exemples d'ensembles munis d'une relation d'ordre.*

– L'ensemble (\mathbb{N}, \leq) est *totalelement ordonné*.

- La relation d'ordre $|$ définie dans \mathbb{N} est partielle. En considérant cette relation, on a, par exemple : $3|5$.
- L'ensemble $(\mathbb{N}^{2 \times 2}, \preceq)$ est partiellement ordonné bien que (\mathbb{N}, \leq) le soit totalement. On peut aisément le remarquer en considérant les deux éléments suivants :

$$\begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix} \parallel \begin{pmatrix} 0 & 0 \\ 1 & 1 \end{pmatrix}.$$

Définition 2 Soit E un ensemble ordonné par \preceq . On peut définir certains éléments particuliers :

Majorant : M est un majorant de $A \subset E$ si $\forall a \in A, a \preceq M$. On dit alors que A est une partie majorée de E .

Minorant : m est un minorant de $A \subset E$ si $\forall a \in A, m \preceq a$. A est une partie minorée de E .

Borne supérieure : $S = \sup A$ (également noté $S = \bigvee A$) est la borne supérieure de A si S est un majorant de A et si pour tout majorant M de A , $S \preceq M$. La borne supérieure est communément appelée "plus petit majorant".

Borne inférieure : $I = \inf A$ (également noté $I = \bigwedge A$) est la borne inférieure de A si I est un minorant de A et si pour tout minorant m de A , $m \preceq I$. La borne inférieure apparaît comme "le plus grand minorant".

Élément maximal : Un élément maximal de $A \subset E$ est un élément a de A tel que A ne contienne aucun élément plus grand que a , ce qui s'écrit :

$$\begin{cases} a \in A \\ x \in A \text{ et } a \preceq x \end{cases} \Rightarrow x = a.$$

Élément minimal : Un élément minimal b de $A \subset E$ est défini par :

$$\begin{cases} b \in A \\ x \in A \text{ et } x \preceq b \end{cases} \Rightarrow x = b.$$

Plus grand élément : On appelle plus grand élément ("Top" en anglais) d'un ensemble E , un élément noté $\top_E \in E$ tel que pour tout élément $x \in E$, on ait $x \preceq \top_E$.

Plus petit élément : On appelle plus petit élément ("Bottom" en anglais) d'un ensemble E , un élément noté $\perp_E \in E$ tel que pour tout élément $x \in E$, on ait $\perp_E \preceq x$.

Remarque 1 .

- Le plus petit élément (resp. plus grand élément), s'il existe, est nécessairement unique.

- S et I peuvent ne pas appartenir à A . Si ces bornes existent, elles sont uniques.
- Même s'il admet des majorants (resp. minorants), un ensemble n'admet pas toujours de borne supérieure (resp. borne inférieure).
- Un ensemble ordonné peut admettre plusieurs éléments minimaux (resp. maximaux).

1.1.2 Demi-treillis et treillis

Parmi les ensembles ordonnés, il est une classe particulièrement riche de propriétés : les *treillis*. En effet, les treillis établissent le lien entre les ensembles ordonnés tels qu'on les a présenté précédemment et les structures algébriques. Nous renvoyons le lecteur à [Birkhoff, 1940], [Davey and Priestley, 1990] et [Grätzer, 2003] pour une étude plus approfondie des treillis.

Définition 3 (Demi-treillis) Soit (E, \preceq) un ensemble ordonné non vide.

- (i) Si $x \vee y$ existe dans E pour tout couple $(x, y) \in E$, alors E est un demi-treillis supérieur.
- (ii) Si $x \wedge y$ existe dans E pour tout couple $(x, y) \in E$, alors E est un demi-treillis inférieur.

Définition 4 (Treillis) L'ensemble ordonné (E, \preceq) est un treillis si c'est à la fois un demi-treillis supérieur et un demi-treillis inférieur.

Remarque 2 Pour toute relation d'ordre, notée \preceq , il existe une relation d'ordre inverse, notée \succeq . Par conséquent, si (E, \preceq) est un demi-treillis supérieur (resp. demi-treillis inférieur) alors (E, \succeq) est un demi-treillis inférieur (resp. demi-treillis supérieur). C'est ce qu'on appelle le principe de dualité.

Remarque 3 Les treillis ont été précédemment définis dans un contexte "ensembliste". Il est néanmoins possible de voir ces ensembles comme des structures algébriques à part entière, munies de deux lois \vee et \wedge jouissant des propriétés classiques d'associativité, de commutativité et d'idempotence. Par la suite, on pourra noter un treillis (E, \preceq) comme la structure algébrique suivante (E, \vee, \wedge) (voir par exemple [Davey and Priestley, 1990] ou [Grätzer, 2003]).

Définition 5 (Demi-treillis complet) Soient (P, \preceq) un demi-treillis supérieur et (Q, \preceq) un demi-treillis inférieur.

- (i) Si $\bigvee S$ existe pour tout $S \subseteq P$, alors P est un demi-treillis supérieur complet.
- (ii) Si $\bigwedge R$ existe pour tout $R \subseteq Q$, alors Q est un demi-treillis inférieur complet.

Définition 6 (Treillis complet) L'ensemble ordonné (E, \preceq) est un treillis complet si c'est à la fois un demi-treillis supérieur complet et un demi-treillis inférieur complet.

Exemple 2 Le treillis $(\mathbb{N}, \vee, \wedge)$ n'est pas complet. $(\mathbb{N} \cup \{+\infty, -\infty\}, \vee, \wedge)$ est un treillis complet.

Remarque 4 Tout treillis fini est complet.

Théorème 1 [Baccelli et al., 1992, th 4.27] Un demi-treillis supérieur complet est un treillis complet, s'il contient un plus petit élément.

1.2 Dioïdes

Dans cette section, on présente la structure algébrique de dioïde, puis on met en évidence le lien entre cette structure et les ensembles ordonnés. Cette synthèse est inspirée des ouvrages [Cuninghame-Green, 1979], [Gondran and Minoux, 2001], [Baccelli et al., 1992].

1.2.1 Dioïdes et relation d'ordre

Définition 7 (Monoïde) Un ensemble \mathcal{E} , muni d'une loi de composition interne \oplus , est un monoïde si la loi est associative et possède un élément neutre. Si, de plus, la loi est commutative, le monoïde est commutatif.

Remarque 5 Dans un monoïde, on appellera parfois loi "additive", la loi \oplus .

Exemple 3 L'ensemble \mathbb{N} muni du produit usuel \times est un monoïde. L'élément neutre de (\mathbb{N}, \times) est 1. En outre, (\mathbb{N}, \times) est un monoïde commutatif.

Définition 8 (Monoïde ordonné) Un monoïde (\mathcal{E}, \oplus) est dit ordonné lorsqu'on peut définir sur \mathcal{E} une relation d'ordre \preceq compatible avec la loi \oplus , c'est-à-dire telle que :

$$\forall a, b, c \in \mathcal{E} \quad a \preceq b \implies a \oplus c \preceq b \oplus c.$$

Définition 9 (Monoïde idempotent) Un monoïde (\mathcal{E}, \oplus) est dit idempotent si la loi \oplus est commutative, associative et idempotente, c'est-à-dire qu'elle vérifie :

$$\forall a \in \mathcal{E}, \quad a \oplus a = a.$$

L'idempotence de la loi pour un monoïde idempotent induit naturellement une relation d'ordre. On montre dans le théorème suivant l'existence de cet ordre.

Théorème 2 *Si (\mathcal{E}, \oplus) est un monoïde idempotent, alors une relation d'ordre \preceq peut être définie par :*

$$a \preceq b \iff a \oplus b = b.$$

Un monoïde idempotent est donc un monoïde ordonné.

Preuve : Comme l'addition est idempotente, on a : $a = a \oplus a \succeq a$, ce qui assure la réflexivité. L'antisymétrie est prouvée par la commutativité de la loi additive : si $a \preceq b$ et $b \preceq a$ alors $b = a \oplus b = b \oplus a = a$, on en déduit directement $a = b$. La transitivité fait appel à l'associativité de la loi \oplus : si $a \preceq b$ et $b \preceq c$ alors $b = a \oplus b$ et $c = b \oplus c$ d'où $c = (a \oplus b) \oplus c = a \oplus (b \oplus c) = a \oplus c$. On vérifie maintenant que la relation d'ordre est compatible avec la loi \oplus . Soient $a, b, c \in \mathcal{E}$ tels que $a \preceq b$, on a $b \oplus c = (a \oplus b) \oplus c = a \oplus b \oplus c = a \oplus c \oplus b \oplus c = (a \oplus c) \oplus (b \oplus c)$ et donc $a \oplus c \preceq b \oplus c$.

□

On définit maintenant les notions de semi-anneau et de dioïde qui sont des structures algébriques définies à partir de monoïdes.

Définition 10 (Semi-anneau) *On appelle semi-anneau un ensemble \mathcal{D} muni de deux lois internes \oplus et \otimes telles que :*

- (i) (\mathcal{D}, \oplus) est un monoïde commutatif dont l'élément neutre est noté ε .
- (ii) (\mathcal{D}, \otimes) est un monoïde (élément neutre noté e).
- (iii) La loi multiplicative \otimes est distributive à droite et à gauche par rapport à la loi additive \oplus .
- (iv) L'élément ε est absorbant pour la loi \otimes ($\forall a \in \mathcal{D}, a \otimes \varepsilon = \varepsilon \otimes a = \varepsilon$).

Si la loi \otimes est commutative, alors $(\mathcal{D}, \oplus, \otimes)$ est un semi-anneau commutatif.

Remarque 6 *La différence fondamentale entre un anneau et un semi-anneau réside dans la propriété (i). Dans un anneau, l'addition induit une structure de groupe, tandis que dans un semi-anneau, elle confère une structure de monoïde.*

Définition 11 (Dioïde) *Soit $(\mathcal{D}, \oplus, \otimes)$ un semi-anneau. Si la loi \oplus est idempotente alors $(\mathcal{D}, \oplus, \otimes)$ est appelé dioïde (ou encore semi-anneau idempotent).*

Puisque (\mathcal{D}, \oplus) est un monoïde idempotent, c'est aussi un monoïde ordonné (théorème 2). Les mêmes propriétés s'appliquent pour un dioïde : l'idempotence de la loi \oplus induit un ordre dans le dioïde $(\mathcal{D}, \oplus, \otimes)$ qui est compatible avec les loi \oplus et \otimes .

Théorème 3 Dans un dioïde $(\mathcal{D}, \oplus, \otimes)$, la relation \preceq définie par :

$$a \preceq b \iff b = a \oplus b$$

est une relation d'ordre compatible avec les lois du dioïde, c'est-à-dire

$$\begin{aligned} a \preceq b &\implies a \oplus c \preceq b \oplus c && \forall c \in \mathcal{D} \\ a \preceq b &\implies a \otimes c \preceq b \otimes c \text{ et } c \otimes a \preceq c \otimes b && \forall c \in \mathcal{D}. \end{aligned}$$

Preuve : L'essentiel de la preuve réside dans la démonstration du théorème 2. Il reste à démontrer que la relation d'ordre est compatible avec la loi multiplicative. Soient $a, b \in \mathcal{D}$ tels que $a \preceq b$ et $c \in \mathcal{D}$, on a $b \otimes c = (a \oplus b) \otimes c = (a \otimes c) \oplus (b \otimes c)$, d'où $a \otimes c \preceq b \otimes c$ (*idem* pour la multiplication à gauche).

□

Notation 1 Comme dans le cas des structures algébriques usuelles, le signe multiplicatif sera parfois omis dans les expressions analytiques concernant les éléments d'un dioïde.

Exemple 4 (Algèbre $(\max, +)$) L'ensemble $\mathbb{R} \cup \{-\infty\}$ muni du \max , noté \oplus , et de l'addition usuelle, notée \otimes est un dioïde (on convient que $\forall x \in \mathbb{R}, x + (-\infty) = -\infty$ et donc $x \otimes \varepsilon = \varepsilon$). Les éléments neutres sont définis comme $\varepsilon = -\infty$ et $e = 0$. Ce dioïde est noté \mathbb{R}_{\max} et est traditionnellement nommé "algèbre $(\max, +)$ ".

Exemple 5 (Algèbre $(\min, +)$) L'ensemble $\mathbb{R} \cup \{+\infty\}$ muni du \min , noté \oplus , et de l'addition usuelle, notée \otimes est un dioïde pour lequel $\varepsilon = +\infty$ et $e = 0$ (on convient que $\forall x \in \mathbb{R}, x + (+\infty) = +\infty$ et donc $x \otimes \varepsilon = \varepsilon$). Ce dioïde est noté \mathbb{R}_{\min} et est traditionnellement nommé "algèbre $(\min, +)$ ".

Exemple 6 (Dioïde matriciel) Soit $(\mathcal{D}, \oplus, \otimes)$ un dioïde, on note $\mathcal{D}^{n \times n}$ l'ensemble des matrices carrées de dimension n à coefficients dans \mathcal{D} . La somme et le produit de matrices sont définies par :

$$\begin{aligned} A \oplus B &: (A \oplus B)_{ij} = A_{ij} \oplus B_{ij} \\ A \otimes B &: (A \otimes B)_{ij} = \bigoplus_{k=1}^n A_{ik} \otimes B_{kj} \end{aligned}$$

L'ensemble $\mathcal{D}^{n \times n}$ muni de ces deux opérations est un dioïde. L'élément nul d'un dioïde matriciel est la matrice composée exclusivement de ε et est noté également ε . L'élément unité

est la matrice notée Id_n qui est composée de e sur la diagonale et de ε partout ailleurs. Cette structure algébrique obéit aux axiomes d'un dioïde. Même si la relation d'ordre est totale dans le dioïde \mathcal{D} , l'ordre dans le dioïde matriciel $\mathcal{D}^{n \times n}$ ($n > 1$) est partiel.

Exemple 7 (Dioïde de séries formelles) Soit (D, \oplus, \otimes) un dioïde et f une application de \mathbb{Z} dans \mathcal{D} . On définit la série formelle $F(z)$ en l'indéterminée z à coefficients dans \mathcal{D} par :

$$F(z) = \bigoplus_{t \in \mathbb{Z}} f(t)z^t.$$

Nous désignons par $\langle F(z), z^t \rangle$ le coefficient $f(t)$ de $F(z)$ pour z^t . L'ensemble des séries formelles en l'indéterminée z et à coefficients dans \mathcal{D} muni des opérations

$$\begin{aligned} F(z) \oplus G(z) &: \langle F(z) \oplus G(z), z^t \rangle = \langle F(z), z^t \rangle \oplus \langle G(z), z^t \rangle \\ F(z) \otimes G(z) &: \langle F(z) \otimes G(z), z^t \rangle = \bigoplus_{i+j=t} \langle F(z), z^i \rangle \otimes \langle G(z), z^j \rangle \end{aligned}$$

est un dioïde noté $\mathcal{D}[[z]]$.

Le support, noté $\text{Supp}(F(z))$, d'une série $F(z)$ est défini comme l'ensemble

$$\text{Supp}(F(z)) = \{t \in \mathbb{Z} \mid f(t) \neq \varepsilon\}.$$

Lorsque le support est fini, on parle de polynôme. Le degré, noté $\text{deg}(F(z))$, (resp. la valuation, notée $\text{val}(F(z))$) est la borne supérieure (resp. la borne inférieure) du support de $F(z)$. Par exemple, la série $F(z) = 2z^1 \oplus 5z^3 \oplus 1z^4$ a pour support $\text{Supp}(F(z)) = \{1, 3, 4\}$, son degré est $\text{deg}(F(z)) = 4$ et sa valuation est $\text{val}(F(z)) = 1$.

1.2.2 Dioïdes et treillis

On a vu précédemment qu'un dioïde a une structure algébrique de semi-anneau ordonné, nous nous efforçons, dans cette section, de présenter les relations entre dioïdes et treillis.

Le théorème 3 nous permet d'affirmer qu'un dioïde a une structure de demi-treillis supérieur du fait de la relation d'ordre induite par la loi additive. Dans un dioïde $(\mathcal{D}, \oplus, \otimes)$, cette borne supérieure, traditionnellement notée \vee , correspond à la loi additive \oplus (tout couple $(a, b) \in \mathcal{D}$ admet $a \oplus b$ comme borne supérieure). De plus, on sait, d'après le théorème 1, qu'un demi-treillis supérieur muni d'un plus petit élément possède en fait une structure de treillis. Ce théorème s'applique directement au cas des dioïdes puisqu'ils possèdent en ε un plus petit élément. Un dioïde possède donc une structure de treillis. On peut donc appliquer directement aux dioïdes

les résultats sur les treillis.

Définition 12 (Dioïde complet) *Un dioïde $(\mathcal{D}, \oplus, \otimes)$ est dit complet s'il est fermé pour les sommes infinies et si la loi \otimes distribue (à gauche et à droite) sur les sommes infinies, c'est-à-dire si pour tout $c \in \mathcal{D}$ et tout sous-ensemble $\mathcal{B} \subseteq \mathcal{D}$,*

$$c \otimes \left(\bigoplus_{x \in \mathcal{B}} x \right) = \bigoplus_{x \in \mathcal{B}} (c \otimes x) \quad \text{et} \quad \left(\bigoplus_{x \in \mathcal{B}} x \right) \otimes c = \bigoplus_{x \in \mathcal{B}} (x \otimes c).$$

La structure de treillis complet d'un dioïde complet \mathcal{D} induit l'existence d'une borne inférieure. Celle-ci sera notée \wedge et définie par

$$a \wedge b = \bigoplus_{\{x \in \mathcal{D} \mid x \preceq a, x \preceq b\}} x.$$

Un dioïde complet admet donc un plus grand élément, noté \top , qui est considéré comme la somme (au sens de la loi \oplus) de tous les éléments du dioïde

$$\top = \bigoplus_{x \in \mathcal{D}} x.$$

L'élément \top est donc absorbant pour l'addition : $\top \oplus a = \top$. Rappelons néanmoins que, puisque ε est absorbant pour la loi \otimes , on a : $\top \otimes \varepsilon = \varepsilon \otimes \top = \varepsilon$.

Exemple 8 *Le dioïde défini dans l'exemple 4 n'est pas complet. Nous devons ajouter la borne supérieure $\top = +\infty$ avec la convention définie précédemment, à savoir $+\infty + (-\infty) = -\infty$. Ce nouveau dioïde complet est noté $\overline{\mathbb{R}}_{max} = (\mathbb{R} \cup \{-\infty\} \cup \{+\infty\}, max, +)$.*

Définition 13 (Dioïde distributif) *Un dioïde $(\mathcal{D}, \oplus, \otimes)$ est distributif s'il est complet et si $\forall a \in \mathcal{D}$ et pour tout sous-ensemble \mathcal{C} de \mathcal{D} , on a :*

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

et

$$\left(\bigoplus_{c \in \mathcal{C}} c \right) \wedge a = \bigoplus_{c \in \mathcal{C}} (c \wedge a).$$

1.3 Applications définies sur des dioïdes

Dans cette section, on met en évidence certaines propriétés concernant les applications définies sur des ensembles ordonnés et plus précisément sur des dioïdes.

Définition 14 (Morphisme de dioïde) Une application f d'un dioïde $(\mathcal{D}, \oplus, \otimes)$ dans un dioïde $(\mathcal{C}, \oplus, \otimes)$ est un homomorphisme si :

$$\forall a, b \in \mathcal{D} \quad f(a \oplus b) = f(a) \oplus f(b) \quad \text{et} \quad f(\varepsilon_{\mathcal{D}}) = \varepsilon_{\mathcal{C}}, \quad (1.1)$$

$$f(a \otimes b) = f(a) \otimes f(b) \quad \text{et} \quad f(e) = e. \quad (1.2)$$

De plus, si f est bijective, nous dirons que f est un isomorphisme. S'il existe un isomorphisme entre \mathcal{D} et \mathcal{C} , nous dirons que \mathcal{D} et \mathcal{C} sont isomorphes.

Une application vérifiant seulement la propriété (1.1) est appelée \oplus -morphisme et une application vérifiant seulement (1.2) est un \otimes -morphisme.

Définition 15 (Isotonie) Une application f d'un dioïde $(\mathcal{D}, \oplus, \otimes)$ dans un dioïde $(\mathcal{C}, \oplus, \otimes)$ est dite isotone si

$$\forall a, b \in \mathcal{D}, \quad a \succeq b \implies f(a) \succeq f(b),$$

autrement dit si pour tout $a, b \in \mathcal{D}$, on a

$$f(a \oplus b) \succeq f(a) \oplus f(b).$$

On dit communément qu'une fonction isotone "conserve l'ordre" (du reste, le terme anglais pour isotonie est "order preserving").

Remarque 7 La dernière équation est obtenue en considérant que $a \oplus b \succeq a$ et $a \oplus b \succeq b$. Puisque f conserve l'ordre, on obtient $f(a \oplus b) \succeq f(a)$ et $f(a \oplus b) \succeq f(b)$ d'où $f(a \oplus b) \succeq f(a) \oplus f(b)$.

Remarque 8 La composition de deux applications isotones demeure isotone.

Définition 16 (Antitonie) Une application g d'un dioïde $(\mathcal{D}, \oplus, \otimes)$ dans un dioïde $(\mathcal{C}, \oplus, \otimes)$ est dite antitone si

$$\forall a, b \in \mathcal{D}, \quad a \succeq b \implies g(a) \preceq g(b).$$

Remarque 9 Notons que le résultat de la composition de deux applications antitones est une application isotone.

Pour les applications isotones, il est possible "d'affiner" la préservation de l'ordre en s'intéressant à l'évolution des bornes supérieures et des bornes inférieures dans l'image d'une application. La notion de *continuité* caractérise les applications pour lesquelles cette conservation est effective.

Définition 17 (Continuité) Soient $(\mathcal{D}, \oplus, \otimes)$ et $(\mathcal{C}, \oplus, \otimes)$ deux dioïdes complets. Une application f de \mathcal{D} dans \mathcal{C} est dite *semi-continue inférieurement* (s.c.i. en abrégé), respectivement *semi-continue supérieurement* (s.c.s.) si, pour tout sous-ensemble $\mathcal{B} \subseteq \mathcal{D}$,

$$f\left(\bigoplus_{x \in \mathcal{B}} x\right) = \bigoplus_{x \in \mathcal{B}} f(x)$$

respectivement,

$$f\left(\bigwedge_{x \in \mathcal{B}} x\right) = \bigwedge_{x \in \mathcal{B}} f(x).$$

Remarque 10 Une application s.c.i. ou s.c.s. est nécessairement isotone puisque

$$\begin{aligned} a \succeq b &\Leftrightarrow a = a \oplus b &\Rightarrow f(a) = f(a \oplus b) = f(a) \oplus f(b) &\Leftrightarrow f(a) \succeq f(b) \\ &\Leftrightarrow b = a \wedge b &\Rightarrow f(b) = f(a \wedge b) = f(a) \wedge f(b) &\Leftrightarrow f(b) \preceq f(a). \end{aligned}$$

Définition 18 (Linéarité) Une application f d'un dioïde $(\mathcal{D}, \oplus, \otimes)$ dans un dioïde $(\mathcal{C}, \oplus, \otimes)$ est dite *linéaire* si elle satisfait les propriétés d'additivité et d'homogénéité :

$$\begin{aligned} \forall a, b \in \mathcal{D}, \forall \alpha \in \mathcal{D}, \quad f(a \oplus b) &= f(a) \oplus f(b) && \text{(additivité)} \\ f(\alpha a) &= \alpha f(a) && \text{(homogénéité)}. \end{aligned}$$

La combinaison des deux conditions mentionnées est connue sous le nom de principe de superposition, soit :

$$\forall a, b \in \mathcal{D}, \forall \alpha, \beta \in \mathcal{D}, \quad f(\alpha a \oplus \beta b) = \alpha f(a) \oplus \beta f(b).$$

Remarque 11 En toute rigueur, on devrait plutôt parler de (\oplus, \otimes) -linéarité du fait de la

structure algébrique particulière d'un dioïde. Dans l'algèbre $(\max, +)$, on parlera de $(\max, +)$ -linéarité.

Exemple 9 La multiplication à gauche par $A \in \mathcal{D}^{n \times n}$, définie par

$$\begin{aligned} L_A : \mathcal{D}^{n \times n} &\longrightarrow \mathcal{D}^{n \times n} \\ x &\longmapsto A \otimes x \end{aligned}$$

est une application linéaire.

1.4 Propriétés spectrales des matrices définies dans un dioïde

Les matrices carrées à coefficients dans un dioïde présentent certaines propriétés spectrales intéressantes qui permettent notamment d'étudier le comportement asymptotique de certains systèmes dynamiques. Cette section est consacrée à la caractérisation du spectre des matrices carrées à coefficients dans un dioïde, c'est-à-dire l'ensemble des valeurs propres de ces matrices. On entend par valeur propres et vecteurs propres d'une matrice $A \in \mathcal{D}^{n \times n}$, les scalaires $\lambda \in \mathcal{D}$ et les vecteurs $v \in \mathcal{D}^n \setminus \{\varepsilon\}$ tels que

$$A \otimes v = \lambda \otimes v.$$

On mentionne aussi la propriété de *cyclicité* de telles matrices.

Des explications suffisantes sont disponibles dans [Cohen, 1995] mais le lecteur pourra trouver l'essentiel de la théorie spectrale dans [Gondran and Minoux, 1979] et [Gaubert, 1992]. Les auteurs y mettent à jour les correspondances qui existent entre la théorie des graphes et les manipulations combinatoires de matrices.

On distinguera, par la suite, deux classes de matrices carrées : les matrices *irréductibles* et les matrices *réductibles*.

Définition 19 (Matrice irréductible) Une matrice $A \in \mathcal{D}^{n \times n}$ est dite irréductible si pour toute paire (i, j) , il existe un entier m tel que $(A^m)_{ij} \neq \varepsilon$.

Toute autre matrice carrée est considérée comme réductible.

Le résultat suivant, dû à [Gondran and Minoux, 1979], constitue l'adaptation du théorème de Perron-Frobenius (voir par exemple [Berman and Plemmons, 1979]) aux matrices irréductibles à coefficients dans un dioïde.

Théorème 4 Une matrice $A \in \mathcal{D}^{n \times n}$ irréductible admet une unique valeur propre notée $\lambda \in \mathcal{D}$ et donnée par

$$\lambda = \left(\bigoplus_{j=1}^n \left(\bigoplus_{i=1}^n (A^j)_{ii} \right) \right)^{1/j}.$$

Concernant les dioïdes, la théorie de Perron-Frobenius ne fournit pas une valeur propre "dominante" comme dans le cas classique mais une unique valeur propre pour les matrices irréductibles. Puisque le spectre se compose d'une seule valeur propre, le rayon spectral $\rho(A)$ correspond à cette valeur propre. Notons qu'à l'unique valeur propre d'une matrice irréductible peuvent être associés plusieurs vecteurs propres. Des méthodes de calcul de cette valeur propre ont été proposées dans [Karp, 1978] et plus récemment dans [Cochet-Terrasson et al., 1998].

Les matrices irréductibles possèdent, en outre, la propriété de *cyclicité*.

Théorème 5 (Cyclicité) [Cohen et al., 1983] Pour une matrice irréductible $A \in \mathcal{D}^{n \times n}$ de valeur propre λ , il existe deux entiers N et c tels que

$$\forall n \geq N, A^{n+c} = \lambda^c \otimes A^n.$$

Le plus petit entier c est appelé *cyclicité* de A .

Remarque 12 On peut constater que les éléments des puissances successives de A forment en fait des suites périodiques ou plus précisément des suites c, λ -périodiques (voir [Gaubert, 1992, chap.6]). Pour ces raisons, l'entier c est parfois appelé *période* et le terme λ est qualifié de *taux*.

Concernant les matrices réductibles, il peut y avoir plusieurs valeurs propres. Des résultats sont obtenus en décomposant la matrice en blocs irréductibles. On rappelle que toute matrice réductible B peut s'écrire sous une forme triangulaire par blocs (voir [Gaubert, 1992, chapitre 4]) :

$$A = P^{-1}BP, \text{ avec } A = \begin{pmatrix} \boxed{A_{11}} & \varepsilon & \dots & \varepsilon \\ \boxed{A_{21}} & \boxed{A_{22}} & \dots & \varepsilon \\ \vdots & & \ddots & \vdots \\ \boxed{A_{n1}} & \dots & \dots & \boxed{A_{nn}} \end{pmatrix} \quad (1.3)$$

où P est une matrice de permutation et où chaque élément $\boxed{A_{ij}}$ correspond à une matrice.

Chacun des blocs sur la diagonale correspond à une matrice irréductible. Par la suite, on note $\rho(\boxed{A_{ii}})$ le rayon spectral de la matrice $\boxed{A_{ii}}$.

Théorème 6 [Gaubert, 1992, chapitre 4] *L'ensemble $\text{spec}(A)$ des valeurs propres d'une matrice réductible A est caractérisé par la condition suivante :*

$$\lambda \in \text{spec}(A) \iff \exists i | \lambda = \rho(\boxed{A_{ii}}) \text{ et } \forall j \geq i, \rho(\boxed{A_{ii}}) \succeq \rho(\boxed{A_{jj}}).$$

On remarque que le bloc $\boxed{A_{ii}}$ de rayon spectral maximal (i.e. $\rho(\boxed{A_{ii}}) = \rho(A)$) vérifie toujours la condition ci-dessus, et donc $\rho(A)$ est toujours valeur propre. Les blocs finaux (on considère que $\boxed{A_{ii}}$ est un bloc final s'il n'est prédécesseur d'aucun bloc) vérifient également toujours cette condition. Leurs valeurs propres sont donc aussi valeurs propres de A . En outre, le nombre de valeurs propres ne peut excéder le nombre de blocs irréductibles qui composent A .

Pour exprimer la propriété de cyclicité des matrices réductibles, nous introduisons la notion de précédence pour les blocs diagonaux d'une telle matrice.

Définition 20 (Prédécesseurs) *L'ensemble des prédécesseurs immédiats d'un bloc $\boxed{A_{ii}}$, noté $\pi(\boxed{A_{ii}})$, est l'ensemble des blocs $\boxed{A_{jj}}$ tels que $\boxed{A_{ij}} \neq \varepsilon$. On définit alors les éléments de l'ensemble*

$$\pi^+(\boxed{A_{ii}}) = \pi(\boxed{A_{ii}}) \cup \pi^2(\boxed{A_{ii}}) \cup \dots$$

que l'on appelle les prédécesseurs de $\boxed{A_{ii}}$. On pose également

$$\pi^*(\boxed{A_{ii}}) = \boxed{A_{ii}} \cup \pi^+(\boxed{A_{ii}}).$$

Théorème 7 (Cyclicité d'une matrice réductible) [Gaubert, 1997, paragraphe 3.7, théorème 17]

Soit $A \in \mathcal{D}^{n \times n}$ une matrice réductible. Il existe deux entiers K_0 et c , et un ensemble de scalaires $\lambda_{ijl} \in \mathcal{D}$, $1 \leq i, j \leq n$, $0 \leq l \leq c - 1$ avec $c > 0$, tels que

$$\forall k \geq K_0, \quad l = [k]_c \implies A_{ij}^{k+c} = \lambda_{ijl}^c A_{ij}^k,$$

où λ_{ijl} appartient à l'ensemble des valeurs propres des blocs de A . Plus précisément, lorsque A est sous une forme bloc-triangulaire (1.3), les scalaires λ_{ijl} appartiennent à l'ensemble formé par la valeur propre du bloc associé à l'élément A_{ii} et celles des blocs prédécesseurs du bloc

auquel A_{ii} . En notant $\boxed{A_{mm}}$ le bloc auquel l'élément A_{ii} appartient, on obtient

$$\lambda_{ijl} \in \{\rho(\boxed{A_{pp}}) \mid \boxed{A_{pp}} \in \pi^*(\boxed{A_{mm}})\}.$$

De plus, si i et j appartiennent au même bloc irréductible $\boxed{A_{qq}}$, alors $\lambda_{ijl} = \rho(\boxed{A_{qq}}), \forall l$.
Soit

$$A_{ij} \in \boxed{A_{qq}} \implies A_{ij}^{k+c} = \lambda^c A_{ij}^k \text{ où } \lambda = \rho(\boxed{A_{qq}}).$$

On retrouve, en fait, la propriété de cyclicité des matrices irréductibles.

On renvoie le lecteur aux articles [Gaubert, 1994a] et [Gaubert, 1994b] pour de plus amples détails sur la cyclicité des matrices. L'auteur y introduit le lien entre les éléments de la puissance d'une matrice et les suites périodiques.

1.5 Théorie de la résiduation

A présent, appliquons aux dioïdes la théorie de la résiduation, initialement destinée aux treillis ([Blyth and Janowitz, 1972]). Des informations plus complètes sur l'application de cette théorie aux dioïdes sont disponibles dans [Baccelli et al., 1992, §4]. Cette théorie va permettre, dans certains cas, de déterminer la plus petite "sur-solution" ou la plus grande "sous-solution" de l'équation

$$f(x) = b \tag{1.4}$$

avec $f : \mathcal{D} \longrightarrow \mathcal{C}$, $(\mathcal{D}, \oplus, \otimes)$ et $(\mathcal{C}, \oplus, \otimes)$ deux dioïdes complets et $b \in \mathcal{C}$. En effet, la résolution de cette équation pose un problème d'inversion de l'application f . En particulier, si l'application f n'est pas surjective, il n'existe pas nécessairement d'antécédent dans \mathcal{D} à $b \in \mathcal{C}$ et si f n'est pas injective, l'équation (1.4) peut présenter plusieurs solutions dans \mathcal{D} .

Néanmoins, la théorie de la résiduation permet de définir des "pseudo-inverses" pour des applications définies sur des treillis, et plus particulièrement sur des dioïdes. Plus précisément, la théorie de la résiduation permet d'établir, lorsqu'elle existe, la plus grande solution de l'inéquation $f(x) \preceq b$. Cette solution est en fait la borne supérieure du sous ensemble $\{x \in \mathcal{D} \mid f(x) \preceq b\}$. Lorsqu'il existe pour tout b , cet élément sera noté $f^\#(b)$. $f^\#$ est appelée application résiduée de f .

$$f^\sharp(b) = \bigoplus_{\{x|f(x)\preceq b\}} x.$$

De la même façon, on peut considérer la borne inférieure de l'ensemble des "sur-solutions". La borne inférieure de cet ensemble, lorsqu'elle existe, sera notée $f^\flat(b)$. f^\flat est appelée application résiduée duale de f :

$$f^\flat(b) = \bigwedge_{\{x|f(x)\succeq b\}} x.$$

Le théorème suivant permet de caractériser la résiduabilité d'une application.

Théorème 8 *Soit f une application isotone d'un dioïde complet \mathcal{D} vers un dioïde complet \mathcal{C} . Sont équivalents*

- (i) *Pour tout $b \in \mathcal{C}$, il existe une plus grande sous-solution à l'équation $f(x) = b$.*
- (ii) *f est s.c.i. et $f(\varepsilon_{\mathcal{D}}) = \varepsilon_{\mathcal{C}}$.*
- (iii) *Il existe une application f^\sharp de \mathcal{C} dans \mathcal{D} qui est isotone et s.c.s., telle que*

$$f \circ f^\sharp \preceq Id_{\mathcal{C}} \tag{1.5}$$

$$f^\sharp \circ f \succeq Id_{\mathcal{D}} . \tag{1.6}$$

Quand f satisfait ces propriétés, elle est dite résiduable et f^\sharp est appelée la résiduée de f . Cette résiduée est unique.

De la même manière, un théorème peut être énoncé dans le cadre de la résiduée duale.

Théorème 9 *Soit f une application isotone d'un dioïde complet \mathcal{D} vers un dioïde complet \mathcal{C} . Sont équivalents*

- (i) *Pour tout $b \in \mathcal{C}$, il existe une plus petite sur-solution à l'équation $f(x) = b$.*
- (ii) *f est s.c.s. et $f(\top_{\mathcal{D}}) = \top_{\mathcal{C}}$.*
- (iii) *Il existe une application f^\flat de \mathcal{C} dans \mathcal{D} qui est isotone et s.c.i., telle que*

$$f \circ f^\flat \succeq Id_{\mathcal{C}} \tag{1.7}$$

$$f^{\flat} \circ f \preceq \text{Id}_{\mathcal{D}} . \quad (1.8)$$

Si f satisfait ces propriétés, on dit que f est *dualement résiduable* et f^{\flat} est appelée la *résiduée duale* de f . L'application f^{\flat} est unique.

Théorème 10 *Si f est une application résiduable de \mathcal{D} dans \mathcal{C} , on a alors les égalités suivantes :*

$$f \circ f^{\sharp} \circ f = f \quad (1.9)$$

$$f^{\sharp} \circ f \circ f^{\sharp} = f^{\sharp} . \quad (1.10)$$

Théorème 11 *Si $g : \mathcal{D} \longrightarrow \mathcal{C}$ et $f : \mathcal{C} \longrightarrow \mathcal{B}$ sont des applications résiduables, alors $f \circ g : \mathcal{D} \longrightarrow \mathcal{B}$ est également résiduable et*

$$(f \circ g)^{\sharp} = g^{\sharp} \circ f^{\sharp} . \quad (1.11)$$

Cette égalité reste vraie avec la résiduation duale.

Considérons à présent les deux applications suivantes :

- le produit à gauche par a noté $L_a, x \mapsto a \otimes x$
- la somme avec a notée $T_a, x \mapsto a \oplus x$.

On considérera que ces applications sont définies sur des diïdes \mathcal{D} et \mathcal{C} supposés complets.

Le point (ii) du théorème 8 nous fournit la preuve de la résiduabilité de L_a . En s'appuyant sur le point (ii) du théorème 9, on peut affirmer que T_a est dualement résiduable.

Nous utiliserons, par la suite, les notations suivantes.

Notation 2 *Nous noterons*

$$L_a^{\sharp}(x) = a \backslash x = \frac{x}{a}$$

l'application résiduée de L_a et

$$T_a^{\flat}(x) = x \oplus a$$

la résiduée duale de T_a .

Les applications L_a^{\sharp} et T_a^{\flat} vérifient certaines propriétés qui découlent de leur isotonie, de leur continuité (L_a^{\sharp} est s.c.s et T_a^{\flat} est s.c.i.) ainsi que des théorèmes 10 et 11. Les tableaux 1.1

et 1.2 listent certaines d'entre elles. Le lecteur pourra trouver l'intégralité des preuves de ces tableaux dans [Baccelli et al., 1992, §4.4.4].

$a \wp(x \wedge y)$	$=$	$a \wp x \wedge a \wp y$	(f.1)
$a \wp(x \oplus y)$	\succeq	$a \wp x \oplus a \wp y$	(f.2)
$a(a \wp x)$	\preceq	x	(f.3)
$a \wp(ax)$	\succeq	x	(f.4)
$a(a \wp(ax))$	$=$	ax	(f.5)
$a \wp(a(a \wp x))$	$=$	$a \wp x$	(f.6)
$a \wp x \wedge b \wp x$	$=$	$(a \oplus b) \wp x$	(f.7)

TAB. 1.1: Propriétés de L_a^\sharp .

Remarque 13 Lorsque a est inversible (c'est-à-dire qu'il existe un a^{-1} tel que $aa^{-1} = e$ et $a^{-1}a = e$), on pourra noter $L_a^\sharp(x) = a \wp x = a^{-1}x$. De plus, manipuler le scalaire a^{-1} dans $\overline{\mathbb{Z}}_{max}$, revient à manipuler l'élément $-a$ de \mathbb{Z} .

Dans le cas matriciel, le calcul de L_A^\sharp et T_C^\flat est donné par

$$(A \wp B)_{ij} = \bigwedge_{l=1}^n A_{li} \wp B_{lj},$$

et

$$(D \wp C)_{ij} = D_{ij} \wp C_{ij},$$

avec des matrices $A, C, D \in \mathcal{D}^{m \times n}$ et $B \in \mathcal{D}^{m \times p}$.

$(x \oplus y) \ominus a = (x \ominus a) \oplus (y \ominus a)$	(f.8)
$(x \wedge y) \ominus a \preceq (x \ominus a) \wedge (y \ominus a)$	(f.9)
$(x \ominus a) \oplus a = x \oplus a$	(f.10)
$(x \oplus a) \ominus a = x \ominus a$	(f.11)
$x \ominus (a \oplus b) = (x \ominus a) \ominus b$	(f.12)
$x \ominus (a \wedge b) = (x \ominus a) \oplus (x \ominus b)$	(f.13)
$ax \ominus ab \preceq a(x \ominus b)$	(f.14)

TAB. 1.2: Propriétés de T_a^b .

1.6 Points fixes d'applications définies sur un dioïde

Grâce à leur structure de treillis, il est possible d'appliquer aux dioïdes les résultats concernant les points fixes d'applications définies sur des treillis. Par la suite, nous différencions les applications isotones (définition 15) des applications antitones (définition 16).

On utilisera les notations suivantes.

Notation 3 *L'ensemble des points fixes d'une application $f : \mathcal{D} \rightarrow \mathcal{D}$ sera notée $\mathcal{F}_f : \mathcal{F}_f = \{x \in \mathcal{D} | f(x) = x\}$. Les points pré-fixes seront désignés par $\mathcal{Q}_f : \mathcal{Q}_f = \{x \in \mathcal{D} | f(x) \preceq x\}$; et on désignera l'ensemble des points post-fixes, i.e. $x \in \mathcal{D}$ tel que $f(x) \succeq x$, par \mathcal{P}_f .*

1.6.1 Applications isotones

Le théorème suivant, bien connu dans la théorie des treillis, proposé dans [Tarski, 1955] (voir autrement [Lassez et al., 1982] pour une discussion sur l'histoire et la paternité des résultats concernant les points fixes sur les ensembles ordonnés), nous informe sur la structure de l'ensemble \mathcal{F}_f quand f est isotone.

Théorème 12 *Soient $f : \mathcal{D} \rightarrow \mathcal{D}$ une application isotone et \mathcal{D} un dioïde complet. L'ensemble*

\mathcal{F}_f est un treillis complet non vide.

Théorème 13 Soient $f : \mathcal{D} \rightarrow \mathcal{D}$ une application isotone et \mathcal{D} un dioïde complet. La borne supérieure (resp. inférieure) de \mathcal{F}_f coïncide avec celle de \mathcal{P}_f (resp. \mathcal{Q}_f) :

$$\begin{aligned} \text{Sup } \mathcal{P}_f &= \text{Sup } \mathcal{F}_f, \\ \text{Inf } \mathcal{Q}_f &= \text{Inf } \mathcal{F}_f. \end{aligned} \tag{1.12}$$

Dans [Cousot and Cousot, 1979], les auteurs proposent une preuve différente du théorème précédent. Dans leur démonstration, ils mettent à jour le théorème suivant qui caractérise les ensembles \mathcal{P}_f et \mathcal{Q}_f . On retrouve également ce théorème dans [Baccelli et al., 1992, th 4.70 et 4.72] dans le cadre des dioïdes.

Théorème 14 Soient $f : \mathcal{D} \rightarrow \mathcal{D}$ une application isotone et \mathcal{D} un dioïde complet. Les ensembles \mathcal{P}_f et \mathcal{Q}_f sont des treillis complets.

Ces théorèmes nous permettent d'affirmer que la borne supérieure (resp. inférieure) de \mathcal{F}_f est aussi un point fixe et que l'on peut substituer sa recherche à celle de la borne supérieure de \mathcal{P}_f (resp. la borne inférieure de \mathcal{Q}_f). La figure 1.1 représente le diagramme de Hasse d'un treillis \mathcal{D} et illustre la structure que peuvent prendre les ensembles \mathcal{P}_f , \mathcal{Q}_f et \mathcal{F}_f pour une application f isotone.

Dans la proposition suivante, on spécifie aux dioïdes une méthode classique pour le calcul du plus grand point fixe d'une application f isotone définie sur un treillis complet (voir par exemple [Zad et al., 1999]).

Proposition 1 Si la suite définie par

$$\begin{aligned} y_0 &= \top \\ y_{k+1} &= f(y_k) \end{aligned} \tag{1.13}$$

converge en un nombre fini k_e d'itérations, alors y_{k_e} est le plus grand point fixe de f .

Preuve : Il suffit de montrer que y_{k_e} est le plus grand point post-fixe de f . L'application du théorème 13 permet alors d'affirmer que y_{k_e} est aussi le plus grand point fixe de f .

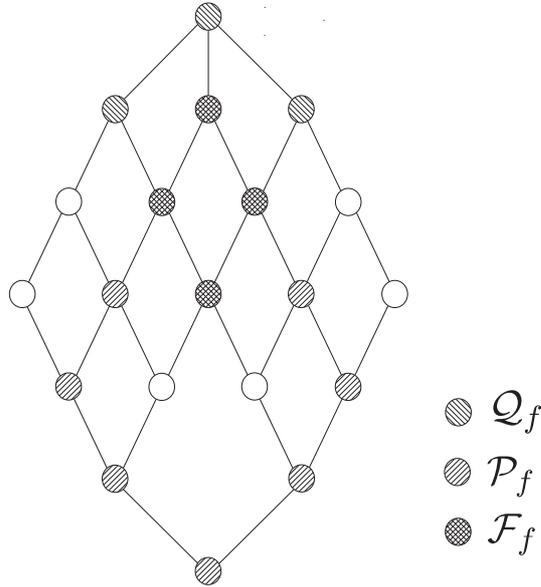


FIG. 1.1: Diagramme de Hasse représentant les ensembles Q_f , P_f et F_f d'une application isotone.

1. On vérifie que y_{k_e} appartient à P_f : $y_{k_e} = f(y_{k_e}) \Rightarrow y_{k_e} \preceq f(y_{k_e})$.
2. On montre maintenant que $y_{k_e} = \text{Sup } P_f$. Soit $x \in P_f$, il suffit de montrer que $\forall k \geq 0$, $y_k \succeq x$ (et donc *a fortiori* que $y_{k_e} \succeq x$). Nous utilisons un raisonnement par récurrence.
 - Au rang 0, on a bien $y_0 \succeq x$.
 - On prouve maintenant que si $y_k \succeq x$ alors $y_{k+1} \succeq x$: $y_{k+1} = f(y_k) \succeq f(x) \succeq x$ (par isotonie de f et sachant que $x \in P_f$).

□

Remarque 14 La suite définie en proposition 1 est décroissante. En effet, on a $u_1 = f(u_0) \preceq u_0 = \top$. De plus, si on considère $u_n = f(u_{n-1}) \preceq u_{n-1}$ alors on a aussi $u_{n+1} = f(u_n) \preceq f(u_{n-1}) = u_n$ puisque f est isotone.

La proposition 2 qui est la version duale de la proposition précédente est donnée sans preuve.

Proposition 2 Si la suite définie par

$$\begin{aligned} y_0 &= \varepsilon \\ y_{k+1} &= f(y_k) \end{aligned} \tag{1.14}$$

converge en un nombre fini k_e d'itérations, alors y_{k_e} est le plus petit point fixe de f .

Remarque 15 La suite définie en proposition 2 est croissante. En effet, on a $u_1 = f(u_0) \succeq u_0 = \varepsilon$. De plus, si on considère $u_n = f(u_{n-1}) \succeq u_{n-1}$ alors on a aussi $u_{n+1} = f(u_n) \succeq f(u_{n-1}) = u_n$ puisque f est isotone.

Corollaire 1 [Baccelli et al., 1992, §4.5.3] L'équation

$$x = Ax \oplus b \tag{1.15}$$

admet une plus petite solution ($f : x \mapsto Ax \oplus b$ est isotone). En appliquant la suite définie en proposition 2, on obtient

$$x_{k+1} = A^k b \oplus A^{k-1} b \oplus \dots \oplus b = (A^k \oplus A^{k-1} \oplus \dots \oplus Id)b.$$

Le terme $A^* = \bigoplus_{i \in \mathbb{N}} A^i$ avec $A^0 = Id$ est connu sous le nom d'étoile de Kleene (ou fermeture de Kleene). L'étoile de Kleene nous permet d'exprimer la plus petite solution de (1.15) qui est donnée par A^*b .

1.6.2 Applications antitones

Concernant les applications antitones, les résultats sont moins nombreux. On peut néanmoins mentionner [Dacić, 1983] et [Baclawski and Björner, 1979]. A notre connaissance, les résultats présentés par la suite sont originaux dans la littérature concernant les dioïdes.

Propriété 1

- Si f est une application isotone et g une application antitone alors $g \circ f$ et $f \circ g$ sont antitones.
- Si g_1 et g_2 sont des applications antitones alors $g_1 \circ g_2$ est isotone.
- Soient \mathcal{D} un dioïde complet, $x, y \in \mathcal{D}$ et g une application antitone. On a

$$\begin{cases} x \oplus y \succeq x \\ x \oplus y \succeq y \end{cases} \Rightarrow g(x \oplus y) \preceq g(x) \wedge g(y)$$

et

$$\begin{cases} x \wedge y \preceq x \\ x \wedge y \preceq y \end{cases} \Rightarrow g(x \wedge y) \succeq g(x) \oplus g(y) .$$

Dans un premier temps, on met en évidence la structure des ensembles \mathcal{P}_g et \mathcal{Q}_g .

Proposition 3 *Soient $g : \mathcal{D} \rightarrow \mathcal{D}$ une application antitone et \mathcal{D} un dioïde complet. L'ensemble \mathcal{Q}_g (resp. \mathcal{P}_g) est un demi-treillis supérieur complet (resp. demi-treillis inférieur complet) non vide selon l'ordre hérité de \mathcal{D} .*

Preuve : De toute évidence, $\top \in \mathcal{Q}_g$ et \mathcal{Q}_g est donc non vide. Considérons deux éléments $x, y \in \mathcal{Q}_g$, d'après le point trois de la remarque 1, on obtient $g(x \oplus y) \preceq g(x) \wedge g(y)$ et donc $g(x \oplus y) \preceq g(x) \oplus g(y)$. Or puisque x et y appartiennent à \mathcal{Q}_g , on en déduit $g(x) \preceq x$ et $g(y) \preceq y$ d'où $g(x) \oplus g(y) \preceq x \oplus y$, et donc $x \oplus y \in \mathcal{Q}_g$, ce qui s'applique aussi aux sommes infinies. On prouve de la même façon que \mathcal{P}_g est un demi-treillis inférieur : $\varepsilon \in \mathcal{P}_g$ (\mathcal{P}_g est non vide) et $g(x \wedge y) \succeq g(x) \oplus g(y) \succeq g(x) \wedge g(y) \succeq x \wedge y$.

□

On caractérise maintenant la structure de l'ensemble \mathcal{F}_g .

Proposition 4 *Soient $g : \mathcal{D} \rightarrow \mathcal{D}$ une application antitone et \mathcal{D} un dioïde complet. Si \mathcal{F}_g est non vide, alors \mathcal{F}_g est une antichaîne selon l'ordre hérité de \mathcal{D} .*

Preuve : Si \mathcal{F}_g est un singleton alors il n'y a rien à prouver. Supposons qu'il existe deux points fixes de g , distincts, notés x et y , et comparables tels que $x \prec y$. Par antitonie de g , on a $g(x) \succeq g(y)$ et, puisqu'ils sont points fixes, $x \succeq y$, ce qui contredit la supposition $x \prec y$.

□

Exemple 10 *On considère une application antitone $g : \mathcal{D} \rightarrow \mathcal{D}$ et \mathcal{D} un dioïde complet définis par la figure 1.2 et les relations suivantes*

$$\begin{aligned} g(\top) &= \varepsilon & g(b) &= c \\ g(a) &= \varepsilon & g(c) &= a \\ g(d) &= \varepsilon & g(\varepsilon) &= a. \end{aligned}$$

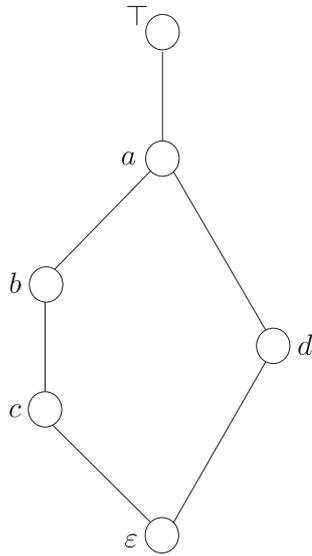


FIG. 1.2: Diagramme de Hasse de l'ensemble \mathcal{D} .

Les ensembles \mathcal{Q}_g , \mathcal{P}_g et \mathcal{F}_g obtenus sont

$$\mathcal{Q}_g = \{\top, a, b, d\}$$

$$\mathcal{P}_g = \{\varepsilon, c\}$$

$$\mathcal{F}_g = \emptyset.$$

Dans cet exemple, l'application antitone g ne possède pas de point fixe.

Proposition 5 Soient $g : \mathcal{D} \rightarrow \mathcal{D}$ une application antitone et $x \in \mathcal{D}$. On a

$$x \oplus g(x) \in \mathcal{Q}_g,$$

$$x \wedge g(x) \in \mathcal{P}_g.$$

Preuve : On a toujours

$$\begin{cases} g(x) \oplus x \succeq x \\ g(x) \oplus x \succeq g(x) \end{cases}$$

et l'antitonie de g implique

$$g(g(x) \oplus x) \preceq g(x) \preceq g(x) \oplus x.$$

De manière similaire, $g(x) \wedge x \in \mathcal{P}_g$ puisque $g(g(x) \wedge x) \succeq g(x) \succeq g(x) \wedge x$.

□

Proposition 6 Soient $g : \mathcal{D} \rightarrow \mathcal{D}$ une application antitone, $y \in \mathcal{P}_g$ et $z \in \mathcal{Q}_g$. Pour tout $x \in \mathcal{D}$ tel que $x \preceq y$ (resp. $x' \in \mathcal{D}$ tel que $x' \succeq z$), on a $x \in \mathcal{P}_g$ (resp. $x' \in \mathcal{Q}_g$).

Preuve : La preuve se déduit de l'antitonie de g :

$$\begin{aligned} x \preceq y &\Rightarrow g(x) \succeq g(y) \succeq y \succeq x \\ x' \succeq z &\Rightarrow g(x') \preceq g(z) \preceq z \preceq x'. \end{aligned}$$

□

Proposition 7 Si x est un point fixe d'une application antitone $g : \mathcal{D} \rightarrow \mathcal{D}$, alors x est un élément minimal (resp. maximal) de \mathcal{Q}_g (resp. \mathcal{P}_g).

Preuve : Soient $x \in \mathcal{F}_g$, $y \in \mathcal{P}_g$ et $z \in \mathcal{Q}_g$ tels que $y \succeq x \succeq z$. Puisque g est antitone, on obtient $g(y) \preceq g(x) \preceq g(z) \Rightarrow y \preceq g(y) \preceq x \preceq g(z) \preceq z$ d'où $y = x = z$. On en conclut donc qu'il n'y a pas d'élément de \mathcal{Q}_g (resp. \mathcal{P}_g) qui soit plus petit (resp. plus grand) que x .

□

Dans la proposition suivante, inspirée par [Dacić, 1983, th. A], on montre que pour une application antitone g , le plus grand élément des points fixes de $g^2 = g \circ g$ est l'image par g du plus petit élément de \mathcal{F}_{g^2} (et vice-versa). On rappelle que g^2 est isotone (cf. remarque 1) et donc que \mathcal{F}_{g^2} est non vide (théorème 12).

Proposition 8 Soit $g : \mathcal{D} \rightarrow \mathcal{D}$ une application antitone. En notant $u = \text{Inf } \mathcal{F}_{g^2}$ et $v = \text{Sup } \mathcal{F}_{g^2}$, on a $g(u) = v$ et $g(v) = u$.

Preuve :

L'expression de $g(u)$ conduit à

$$g(u) = g\left(\bigwedge_{x \in \mathcal{F}_{g^2}} x\right) \succeq \bigoplus_{x \in \mathcal{F}_{g^2}} g(x) \quad (1.16)$$

$$(g \text{ antitone} \Rightarrow g(a \wedge b) \succeq g(a) \oplus g(b)).$$

Cependant, les éléments de $\{g(x) | x \in \mathcal{F}_{g^2}\}$ sont aussi des points fixes de g^2 puisque $g^2(g(x)) = g(g^2(x)) = g(x)$. On peut donc en déduire que $g|_{\mathcal{F}_{g^2}}$ (où $g|_{\mathcal{F}_{g^2}}$ dénote la restriction de g à l'ensemble \mathcal{F}_{g^2}) est une permutation¹. Cette assertion se justifie quand on considère $x, y \in \mathcal{F}_{g^2}$, $x \neq y$ et $g(x) = g(y)$, on obtient alors $g^2(x) = g^2(y)$ et donc $x = y$ ce qui est en contradiction avec l'hypothèse de départ. L'inégalité (1.16) se reformule ainsi

$$g(u) \succeq \bigoplus_{y \in \mathcal{F}_{g^2}} y = v.$$

On a précédemment remarqué que $g(x)$ avec $x \in \mathcal{F}_{g^2}$ est un point fixe. L'élément $g(u)$ est donc également point fixe de g^2 . On a donc $g(u) = \bigoplus_{y \in \mathcal{F}_{g^2}} y = v$. Cette dernière égalité nous mène naturellement à $g(g(u)) = u = g(v)$.

□

Corollaire 2 *L'élément u est un point post-fixe de g et l'élément v est un point pré-fixe de g .*

Preuve : Il vient directement que $g(u) \succeq u$ et $g(v) \preceq v$ puisqu'on a $v \succeq u$ (u et v sont toujours comparables car \mathcal{F}_{g^2} a une structure de treillis complet), $g(u) = v$ et $g(v) = u$.

□

Étant donné que g^2 est isotone, les propositions 1 et 2 fournissent une méthode itérative pour le calcul de u et v .

Corollaire 3 *Si $u = v$ alors l'application g possède un seul point fixe u .*

Preuve : On remarque tout d'abord que si $u = v$ alors l'ensemble des points fixes de g^2 n'est formé que d'un seul élément u . La proposition 8 nous montre que $u \in \mathcal{P}_g$ et $v \in \mathcal{Q}_g$, on en déduit naturellement que si les deux éléments sont confondus alors ils forment un point fixe de g . En outre, les points fixes de g sont aussi point fixes de g^2 et puisque g^2 ne possède qu'un point fixe alors u est l'unique point fixe de g .

¹Une permutation est définie comme une bijection d'un ensemble sur lui même.

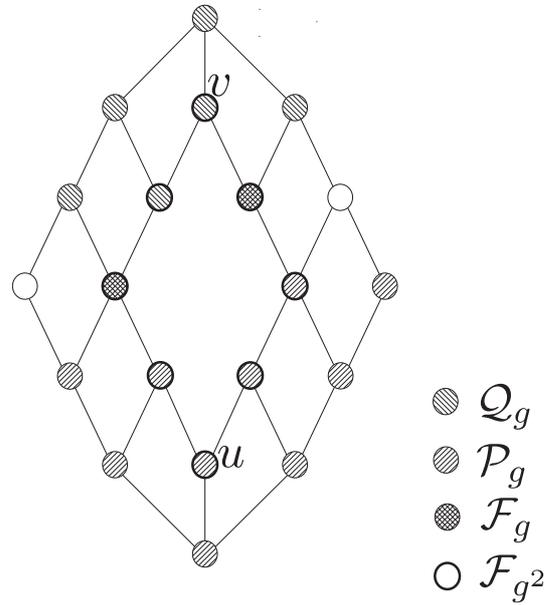


FIG. 1.3: Diagramme de Hasse représentant les ensembles Q_g , P_g , F_g et F_{g^2} d'une application antitone g .

□

Exemple 11 En reprenant l'application définie dans l'exemple 10, on obtient

$$\begin{aligned} g^2(\top) &= a & g^2(a) &= a \\ g^2(b) &= a & g^2(c) &= \varepsilon \\ g^2(d) &= a & g^2(\varepsilon) &= \varepsilon, \end{aligned}$$

d'où $F_{g^2} = \{a, \varepsilon\}$. En accord avec la proposition 8 et le corollaire 2, les éléments u et v sont définis par $v = a$ et $u = \varepsilon$.

La figure 1.3 représente un treillis \mathcal{D} et illustre les structures que peuvent prendre les ensembles Q_g , P_g , F_g et F_{g^2} lorsque g est une application antitone.

Chapitre 2

Systemes $(max, +)$ -linéaires

La classe des systèmes à événements discrets que nous étudions ici est celle qui met en jeu des phénomènes de synchronisation et de parallélisme. Ceux-ci se rencontrent régulièrement dans les systèmes de production [Cohen et al., 1983], les systèmes de transport [Braker, 1993] [Lotito et al., 2005] et les systèmes informatiques [Boudec and P., 2001].

L'objectif de ce chapitre est de proposer un survol de la modélisation et de la commande des systèmes qui peuvent être appréhendés comme des systèmes linéaires sur la structure algébrique de dioïde.

Nous montrons tout d'abord qu'une sous-classe particulière des réseaux de Petri (voir [David and Alla, 1989] ou [Murata, 1989] pour un exposé plus complet sur les réseaux de Petri), les Graphes d'Événements Temporisés (GET), fournit une représentation graphique pertinente des systèmes considérés dans notre étude. On présente deux mises en équations des GET qui conduisent toutes deux à des modèles linéaires mais dans des dioïdes différents.

Ensuite, on se ramène à un modèle mathématique couramment utilisé par l'automaticien : la représentation d'état linéaire. Les conditions initiales de ces systèmes sont également évoquées. Ces systèmes disposent également d'une représentation entrées-sorties que nous décrivons.

Puis nous introduisons le dioïde $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$ dans lequel les trajectoires associées aux événements sont manipulés comme des séries formelles. L'intérêt réside dans le fait que l'on peut alors représenter ces systèmes par leur fonction de transfert.

Ce chapitre se termine par un survol de la commande des systèmes $(max, +)$ -linéaires. Nous exposons les commandes élaborées en les distinguant suivant le critère considéré, les objectifs de commande désirés et la structure de commande adoptée.

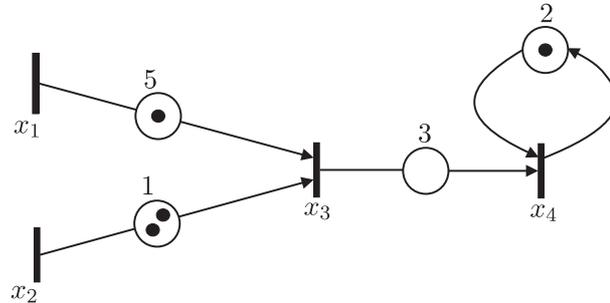


FIG. 2.1: Exemple de graphe d'événements temporisés.

2.1 Mise en équations de graphes d'événements temporisés

2.1.1 Graphes d'événements temporisés

Les GET constituent une classe de réseaux de Petri pour lesquels toute place n'a qu'une transition en amont et une transition en aval. Cette structure permet de modéliser la synchronisation et le parallélisme par contre elle exclut tout phénomène de concurrence. On peut citer [Baccelli et al., 1992, chap 2] pour un exposé plus complet sur les GET. La figure 2.1 représente un GET.

Dans le cadre de la modélisation des systèmes à événements discrets, nous considérerons qu'une transition d'un GET correspond à un événement et que le tir de celle-ci est une occurrence de cet événement. Dans la figure 2.1, il est clair que la transition x_3 exprime la synchronisation des événements x_1 et x_2 retardés respectivement de 5 et 1 unités de temps. Les places temporisées, qui induisent un temps minimum de séjour pour les jetons, ont pour rôle de traduire une tâche qui possède une durée. Les jetons qui sont initialement présents dans le système induisent un décalage. Par exemple, le tir numéro k de la transition x_3 sera engendré par le tir numéro $k - 1$ de la transition x_1 et l'activation numéro $k - 2$ de x_2 .

2.1.2 Deux représentations analytiques des graphes d'événements temporisés

Les GET sont ici utilisés comme outil de modélisation intermédiaire. Si cette modélisation graphique constitue une première étape, la seconde étape est la mise en équations du modèle graphique, c'est-à-dire la définition d'une représentation analytique du système. La modélisation des GET dans un dioïde peut être réalisée selon deux points de vue, elle conduit dans les deux

cas à des modèles linéaires, mais dans des dioïdes différents.

Le premier modèle, appelé représentation en *dateurs*, amène à manipuler des variables discrètes qui correspondent aux dates d'activation des transitions du GET. Plus précisément, on associe à l'événement x la variable discrète $x(\cdot)$ pour laquelle l'itéré $x(k)$ désigne la date de la $k + 1$ ème occurrence de l'événement (en effet, on considère que la date de la première occurrence de x est $x(0)$). Il est ensuite possible d'exprimer les variables dateurs les unes en fonction des autres. C'est le cas lorsque des événements, tels que x_3 sur la figure 2.1, sont liés par des phénomènes de synchronisation ou de retard à d'autres événements, x_1 et x_2 sur la figure.

De cette façon, on peut mettre en équations le GET de la figure 2.1.

$$\begin{cases} x_3(k) &= \max(5 + x_1(k-1), 1 + x_2(k-2)) \\ x_4(k) &= \max(3 + x_3(k), 2 + x_4(k-1)). \end{cases}$$

La représentation obtenue n'est pas linéaire. Elle met en jeu l'opérateur \max . La présence de cet opérateur se justifie par le fait qu'un dateur, par exemple $x_3(k)$, dépend de la date d'activations des transitions amont (x_1 et x_2 pour x_3) et des jetons présents dans les places amont. Le terme $x_3(k)$ est au moins aussi grand que $5 + x_1(k-1)$ qui correspond à la date du tir n° $k-1$ de x_1 (ce décalage est dû au jeton initialement présent) retardé de 5 unités de temps (à cause de la temporisation entre x_1 et x_3). Pour les mêmes raisons, $x_3(k)$ est au moins aussi grand que $1 + x_2(k-2)$. Finalement, $x_3(k)$ doit être supérieur ou égal à $5 + x_1(k-1)$ et $1 + x_2(k-2)$, l'égalité au \max de ces deux expressions résulte des hypothèses de fonctionnement *au plus tôt* (dès qu'une transition peut être tirée, elle est tirée) et des conditions initiales canoniques (les jetons initiaux sont disponibles depuis toujours, voir §2.2.2 plus loin).

Ce système d'équations peut se reformuler en utilisant l'algèbre $(\max, +)$ (définie dans l'exemple 4) :

$$\begin{cases} x_3(k) &= 5 \otimes x_1(k-1) \oplus 1 \otimes x_2(k-2) \\ x_4(k) &= 3 \otimes x_3(k) \oplus 2 \otimes x_4(k-1). \end{cases} \quad (2.1)$$

La loi additive traduisant la synchronisation, le modèle obtenu dans $\overline{\mathbb{Z}}_{\max}$ est linéaire.

La mise en équation de GET peut également se faire dans le domaine temporel, où le système est décrit par des fonctions dépendant du temps t . Dans cette optique, on ne s'intéresse plus aux dates d'activation des transitions mais au nombre d'activations de ces dernières jusqu'à un moment donné. En effet, cette représentation consiste à associer à chaque transition une variable

discrète $x(\cdot)$ (bien qu'il soit possible de la considérer continue) que l'on appellera *compteur*. Le compteur $x(t)$ désigne le nombre d'activations de la transition x survenues avant ou à la date t .

On peut mettre en équation les compteurs associés aux transitions x_3 et x_4 du GET de la figure 2.1 :

$$\begin{cases} x_3(t) &= \min(1 + x_1(t - 5), 2 + x_2(t - 1)) \\ x_4(t) &= \min(x_3(t - 3), 1 + x_4(t - 2)). \end{cases}$$

La traduction dans le dioïde $\overline{\mathbb{Z}}_{min}$ de ces équations aux compteurs conduit à un système d'équations linéaires :

$$\begin{cases} x_3(t) &= 1 \otimes x_1(t - 5) \oplus 2 \otimes x_2(t - 1) \\ x_4(t) &= x_3(t - 3) \oplus 1 \otimes x_4(t - 2). \end{cases} \quad (2.2)$$

2.2 Représentation d'état

2.2.1 Forme ARMA, forme d'état

Les SED étudiés ici, c'est-à-dire ceux qui se modélisent par des GET, bénéficient d'un modèle d'état linéaire. La représentation d'état permet une description interne d'un processus. Plus précisément, le système va être caractérisé par un ensemble de variables, internes au système, appelées variables d'état. Le modèle d'état est obtenu en établissant des équations liant les variables d'état les unes aux autres. Connaissant la valeur de l'état à un instant ainsi que la commande appliquée, nous serons en mesure de "prédire" l'évolution de l'état.

De façon classique, on appréhende un système comme stationnaire par un modèle dont les paramètres ne varient pas. En termes de réseaux de Petri, cela se traduit par l'invariance des valeurs des décalages temporels (induits par les temporisations) et événementiels (induits par les jetons initiaux) au cours du temps. Nous nous limitons ici à cette classe de systèmes. Notons que les systèmes $(max, +)$ -linéaires non stationnaires ont été étudiés dans [Lahaye, 2000].

Nous considérons donc dans notre étude des systèmes qui peuvent se mettre sous la forme d'une équation de récurrence du type :

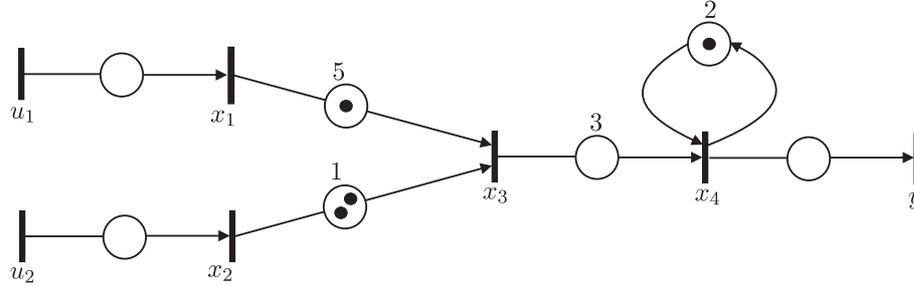


FIG. 2.2: Exemple de graphe d'événements temporisés possédant une représentation d'état linéaire.

$$\begin{cases} x(k) = \bigoplus_{i=0}^a A(i)x(k-i) \oplus \bigoplus_{j=0}^b B(j)u(k-j) \\ y(k) = \bigoplus_{l=0}^c C(l)x(k-l), \end{cases} \quad (2.3)$$

où $x(k)$ est un vecteur de dateurs. La première équation est appelée *équation d'évolution* et la seconde *équation de sortie*. La représentation (2.3) est sous la forme dite "ARMA" (de l'anglais Auto Regressive-Moving Average que l'on peut traduire par forme auto régressive à moyenne mobile), c'est-à-dire qu'elle est composée d'un terme récursif "AR" et d'un terme extérieur "MA".

Sur la figure 2.2, nous reprenons le GET de la figure 2.1 auquel nous ajoutons des variables d'entrée (dateurs associés aux transitions sources u_1 et u_2) et de sortie (dateurs associés à la transition puits y_1). Ce système possède une représentation d'état linéaire du type (2.3). On obtient le modèle suivant :

$$\left\{ \begin{array}{l} x(k) = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 3 & \varepsilon \end{pmatrix} x(k) \oplus \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ 5 & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & 2 \end{pmatrix} x(k-1) \\ \oplus \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & 1 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{pmatrix} x(k-2) \oplus \begin{pmatrix} e & \varepsilon \\ \varepsilon & e \\ \varepsilon & \varepsilon \\ \varepsilon & \varepsilon \end{pmatrix} u(k), \\ y(k) = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & e \end{pmatrix} x(k). \end{array} \right.$$

Dans l'équation (2.3), on peut faire apparaître le terme implicite comme suit :

$$x(k) = A(0)x(k) \oplus \bigoplus_{i=1}^a A(i)x(k-i) \oplus \bigoplus_{j=0}^b B(j)u(k-j).$$

Le corollaire 1 fournit la plus petite solution de cette équation implicite. Sélectionner cette solution permet de décrire comme suit le fonctionnement au plus tôt du système :

$$x(k) = \bigoplus_{i=1}^a A(0)^* A(i)x(k-i) \oplus \bigoplus_{j=0}^b A(0)^* B(j)u(k-j).$$

La manipulation combinatoire classique en Automatique conventionnelle qui consiste à étendre le vecteur d'état pour se ramener à une récurrence d'ordre 1 sur la partie AR et de 0 sur la partie MA permet de passer à une représentation d'état standard décrite par :

$$\begin{cases} x(k) = Ax(k-1) \oplus Bu(k) \\ y(k) = Cx(k). \end{cases} \quad (2.4)$$

Cette représentation, dite $(max, +)$ -linéaire, est la représentation d'état en dateurs. Son analogue en termes de compteurs existe également et est dite $(min, +)$ -linéaire.

Concernant le système de la figure 2.2, la représentation d'état est :

$$\left\{ \begin{array}{l} x(k) = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ 5 & \varepsilon & \varepsilon & \varepsilon & 4 \\ 8 & \varepsilon & \varepsilon & 2 & 4 \\ \varepsilon & e & \varepsilon & \varepsilon & \varepsilon \end{pmatrix} x(k-1) \oplus \begin{pmatrix} e & \varepsilon \\ \varepsilon & e \\ \varepsilon & \varepsilon \\ \varepsilon & \varepsilon \\ \varepsilon & \varepsilon \end{pmatrix} u(k) \\ y(k) = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & e & \varepsilon \end{pmatrix} x(k). \end{array} \right. \quad (2.5)$$

Remarque 16 Des outils informatiques existent pour manipuler ces systèmes. On peut notamment citer la boîte à outils [SWtool, 2005a], dédiée aux calculs dans $\overline{\mathbb{Z}}_{max}$, qu'on utilise avec le logiciel de calcul matriciel Scilab.

Remarque 17 Si on s'attarde sur la représentation (2.5) obtenue pour le système de la figure 2.2, on remarque que la variable x_3 n'intervient ni dans la dynamique des autres variables d'état ni dans l'équation de sortie. Elle peut être supprimée, ce qui entraîne une nouvelle représentation du système de dimension inférieure :

$$\left\{ \begin{array}{l} x(k) = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ 8 & \varepsilon & 2 & 4 \\ \varepsilon & e & \varepsilon & \varepsilon \end{pmatrix} x(k-1) \oplus \begin{pmatrix} e & \varepsilon \\ \varepsilon & e \\ \varepsilon & \varepsilon \\ \varepsilon & \varepsilon \end{pmatrix} u(k) \\ y(k) = \begin{pmatrix} \varepsilon & \varepsilon & e & \varepsilon \end{pmatrix} x(k). \end{array} \right.$$

Pour un même système, il est possible d'avoir plusieurs représentations d'état équivalentes.

Remarque 18 A la représentation d'état (2.5) on peut associer un GET différent de celui d'origine dessiné sur la figure 2.2. La transformation du GET résulte des manipulations pour aboutir à la forme (2.4) (suppression du terme implicite et extension du vecteur d'état).

2.2.2 Conditions initiales canoniques

Les conditions initiales d'un système correspondent à l'état lors de l'instant initial, c'est-à-dire l'état de départ. La connaissance de celui-ci, combiné à sa représentation d'état, permet d'étudier le comportement du système en réponse à une excitation des entrées.

L'état initial d'un système $(max, +)$ -linéaire est défini par un vecteur noté $v(k)$ qui s'ajoute à l'équation d'évolution (pour une discussion plus approfondie sur les conditions initiales voir [Baccelli et al., 1992, §2.6.3.4 et §5.4.4.1]) :

$$x(k) = Ax(k-1) \oplus Bu(k) \oplus v(k).$$

Plus précisément, $v_i(k)$ pour $0 \leq k < k_{d_i}$ renvoie les dates d'occurrence au plus tôt des événements initiaux (c'est-à-dire ceux engendrés par les jetons du marquage initial). L'indice k_{d_i} est celui de la première occurrence de x_i qui est engendrée par les entrées du système. La notion de nombre caractéristique, introduite dans [Boimond and Ferrier, 1996], permet de déterminer cet indice.

Définition 21 (Nombre caractéristique) Soit $[A]_i$ la i -ième ligne de la matrice A , le nombre caractéristique associé à la variable d'état x_i du modèle décrit par (2.4), s'il existe, est le plus petit entier, noté k_{d_i} , tel que $[A^{k_{d_i}}]_i B \neq \varepsilon$ où ε désigne ici un vecteur ligne de n éléments tous égaux à ε . Les variables n et p sont respectivement la dimension de l'état et du vecteur de commande.

Pour pouvoir être manipulée comme un dateur, chaque variable v_i est prolongée comme suit : $v_i(k) = \varepsilon$ pour $k < 0$ et $v_i(k) = v_i(k_{d_i} - 1)$ pour $k \geq k_{d_i}$. On dit que les conditions initiales sont *canoniques* si $\forall k \in \mathbb{Z}$, $v(k) = \varepsilon$ et on se ramène alors à l'équation d'état (2.4), c'est-à-dire sans le vecteur $v(k)$. En termes de réseaux de Petri, cela revient à admettre que les jetons du marquage initial sont disponibles depuis l'instant $t = -\infty$. Si des jetons sont initialement présents dans le système, la notion de conditions initiales canoniques peut être vue comme une "énergie interne" du système, à savoir qu'en l'absence de tir des transitions d'entrée, le tir de certaines variables d'état, voire de sortie, reste possible. Pour chaque variable d'état, le nombre caractéristique équivaut au nombre d'occurrences possibles sans occurrence d'événements d'entrée (c'est-à-dire sans tir des transitions d'entrée).

Remarque 19 Dans le cas des compteurs, puisque la notion de conditions initiales canoniques est synonyme de jetons du marquage initial disponibles à la date $-\infty$, les activations des transitions dues au marquage initial ont lieu à la date $-\infty$ et donc à $t = 0$, un compteur $x(t)$ peut être supérieur à 0. Le GET de la figure 2.2 permet d'illustrer la notion de conditions initiales canoniques. En effet, la première occurrence de x_3 a lieu à la date $-\infty$ et donc le compteur $x_3(t)$ vaut 1 dès cette date.

Exemple 12 Pour le GET considéré sur la figure 2.2 et en tenant compte de la représentation d'état exposée dans la remarque 17, le calcul des nombres caractéristiques donne

$$\begin{aligned} k_{d_1} &= 0 & k_{d_2} &= 0 \\ k_{d_3} &= 1 & k_{d_4} &= 1. \end{aligned}$$

Les événements x_3 et x_4 ne seront conditionnés par les variables d'entrée qu'à partir de leur deuxième occurrence, i.e. l'occurrence numéro 1.

Le nombre caractéristique k_{d_i} correspond au décalage événementiel entre la variable d'état x_i et les entrées du système. Il est également possible de calculer le décalage événementiel entre un état x_i et une sortie y_j .

Définition 22 Le décalage événementiel entre un état x_i et une sortie y_j , s'il existe, est défini comme le plus petit entier $k_{f_{ji}}$ tel que $C_j[A^{k_{f_{ji}}}]^i \neq \varepsilon$ (la notation $[A]^i$ désigne la i -ème colonne de A).

Exemple 13 Concernant le GET de la figure 2.2, le modèle de la remarque 17 fournit les décalages événementiels suivants

$$\begin{aligned} k_{f_{11}} &= 1 & k_{f_{12}} &= 2 \\ k_{f_{13}} &= 0 & k_{f_{14}} &= 1. \end{aligned}$$

2.2.3 Systèmes autonomes $(max, +)$ -linéaires

Certains systèmes $(max, +)$ -linéaires possèdent la particularité d'être autonomes, c'est-à-dire qu'ils ne comportent pas d'entrée¹. Leur évolution est donc uniquement décrite à partir des conditions initiales et de la matrice d'évolution. La représentation d'état de ces systèmes prend la forme suivante :

$$x(k) = Ax(k-1). \quad (2.6)$$

En effectuant les itérations successives de la relation (2.6), on obtient

¹ou encore que l'influence des entrées sur l'évolution de l'état peut être négligée (typiquement, $u(k) = \varepsilon, \forall k$).

$$\begin{aligned}
x(k) &= Ax(k-1) \\
&= A^2x(k-2) \\
&= \vdots \\
&= A^kx(0).
\end{aligned}$$

Soit

$$x(k+c) = A^{k+c}x(0).$$

On peut remarquer que l'évolution de l'état peut être caractérisée simplement à partir des puissances de la matrice d'évolution. On distingue deux cas suivant la structure de la matrice d'évolution. Si celle-ci est irréductible, après un régime transitoire le vecteur d'état tout entier entre dans un régime périodique.

La périodicité de ces systèmes est directement déduite de la propriété de cyclicité des matrices $(max, +)$ irréductibles (énoncée dans le théorème 5). Pour tout k suffisamment grand, on a en effet

$$\begin{aligned}
x(k+c) &= A^{k+c}x(0) \\
&= \lambda^c A^k x(0) \\
&= \lambda^c x(k).
\end{aligned}$$

Autrement dit, c occurrences d'événements ont lieu en $c \times \lambda$ unités de temps, où λ est la valeur propre de A et c sa cyclicité (*cf.* théorème 5). Le scalaire λ correspond par conséquent au temps de cycle moyen (soit l'inverse du taux de production).

Pour les matrices d'évolution réductibles, il est possible d'obtenir une inégalité bornant supérieurement l'évolution de chacune des composantes du vecteur d'état quand le régime stationnaire est atteint.

C'est la propriété de cyclicité de la matrice d'évolution réductible (énoncée dans le théorème 6) qui permet de caractériser l'évolution d'un dateur $x_i(k)$ donnée par l'équation (2.6). On suppose sans perte de généralités que, la matrice A est sous la forme bloc-triangulaire inférieure (1.3). Pour tout $k \geq K_0$, on a alors

$$\begin{aligned}
x_i(k+c) &= \bigoplus_{j=1}^i A_{ij}^{k+c} x_j(0) \\
&= \bigoplus_{j=1}^i \lambda_{ijl}^c A_{ij}^k x_j(0).
\end{aligned}$$

On rappelle que λ_{ijl} appartient à l'ensemble formé par la valeur propre du bloc associé à l'élément A_{ii} et celles des blocs prédécesseurs du bloc auquel A_{ii} appartient. Soit

$$\lambda_{ijl} \in \{\rho(\boxed{A_{pp}}) \mid \boxed{A_{pp}} \in \pi^*(\boxed{A_{mm}})\},$$

en notant $\boxed{A_{mm}}$ le bloc auquel l'élément A_{ii} appartient (*cf.* théorème 6).

On pose

$$\forall i \in \boxed{A_{mm}}, \Lambda_i = \max\{\rho(\boxed{A_{pp}}) \mid \boxed{A_{pp}} \in \pi^*(\boxed{A_{mm}})\}.$$

ce qui nous permet de borner supérieurement l'évolution du dateur x_i de la façon suivante :

$$\begin{aligned} x_i(k+c) &\leq \Lambda_i^c \left(\bigoplus_{j=1}^i A_{ij}^k x_j(0) \right) \\ \Leftrightarrow x_i(k+c) &\leq \Lambda_i^c x_i(k). \end{aligned}$$

2.3 Réponse impulsionnelle

On met ici en évidence la relation entrée-sortie d'un système décrit sous sa forme d'état (2.4).

Le développement de la récurrence (2.4) donne

$$\begin{aligned} y(k) &= Cx(k) \\ &= CAx(k-1) \oplus CBu(k) \\ &= CA^2x(k-2) \oplus CABu(k-1) \oplus CBu(k) \\ &= \vdots \\ &= CA^p x(k-p) \oplus \bigoplus_{i=0}^{p-1} CA^i Bu(k-i). \end{aligned}$$

On pose maintenant

$$h(i) = \begin{cases} \varepsilon & \text{si } i < 0 \\ CA^i B & \text{sinon.} \end{cases}$$

Par ailleurs, en adoptant des conditions initiales canoniques, on a $x(j) = \varepsilon$ et $u(j) = \varepsilon$ pour $j < 0$. Aussi, pour un k donné, et en considérant un p assez grand, c'est-à-dire $p \geq k$, on peut écrire

$$y(k) = \bigoplus_{i \in \mathbb{Z}} h(i)u(k-i) . \quad (2.7)$$

Dans l'équation (2.7), la sortie n'est plus dépendante de l'état du système mais uniquement de son entrée. Cela s'explique par le fait que l'on a considéré $p \geq k$. A savoir que dans l'équation (2.7), nous considérons le système depuis le début de l'établissement de la commande, pour les événements antérieurs, on a $x(k) = \varepsilon$ et $y(k) = \varepsilon$. Cette équation peut être perçue comme la somme des effets de la commande. De ce point de vue, $y(k)$ est le résultat de la k -ième occurrence de la commande, *i.e.* $h(0)u(k)$, de la $k-1$ -ième occurrence de la commande, *i.e.* $h(1)u(k-1)$, de la $k-2$ -ième occurrence de la commande... et ce, jusqu'au début de l'établissement de la commande.

La variable $h(k)$ est également appelée *réponse impulsionnelle* dans le sens où elle correspond à la sortie du système lorsqu'on lui applique une entrée du type :

$$e(i) = \begin{cases} \varepsilon & \text{si } i < 0 \\ e & \text{sinon.} \end{cases}$$

Ce signal est considérée comme une impulsion (à l'événement 0) dans la théorie des systèmes $(max, +)$ -linéaires. Dans le cadre des GET, cette entrée correspond à tirer une infinité de fois la transition correspondante à l'instant 0.

Remarque 20 .

- On se permet de remarquer la forte analogie de la réponse impulsionnelle donnée en (2.7) avec la relation entrée-sortie des systèmes continus échantillonnés.
- Dans le cas d'un système multi-entrées ou multi-sorties, la réponse impulsionnelle $h(k)$ est une matrice. L'élément $h_{ij}(k)$ représente la réponse du k -ième événement de la sortie y_i à une impulsion (à l'événement 0) sur l'entrée u_j .
- La réponse d'un système $(max, +)$ -linéaire invariant à une entrée quelconque $u(k)$ est la convolution linéaire de $u(k)$ avec la réponse impulsionnelle $h(k)$ du système. On parle de *sup-convolution* pour désigner l'opération (2.7), notamment dans [Baccelli et al., 1992], à cause de la structure algébrique dans laquelle on décrit les systèmes (dans le dioïde $(min, +)$, on parle d'*inf-convolution*).

2.4 Représentation en séries formelles

Dans la théorie conventionnelle des systèmes, face à un produit de convolution, il est classique d'introduire la transformée de Fourier ou de Laplace des fonctions considérées. Les modèles discrets ont aussi une représentation fréquentielle : la transformée en z . La classe de systèmes considérés ici disposent également d'une transformée. Cette transformation, introduite par [Cohen et al., 1989], permet de prendre en compte les deux aspects des représentations en dateurs ou en compteurs.

2.4.1 Le dioïde $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$

Dans l'exemple 7, on a présenté un dioïde de séries formelles ainsi que la manipulation de ses éléments au travers des lois \oplus et \otimes . Le dioïde $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$ est un dioïde de séries formelles particulièrement bien adapté à la modélisation des GET. En effet, il permet de manipuler uniquement des séries croissantes et comme on va le voir par la suite, ce sont ces séries qui permettent de coder les tirs des transitions d'un GET. Dans ce dioïde, on manipule des séries formelles en deux indéterminées commutatives γ et δ à exposants dans \mathbb{Z} et à coefficients booléens.

On définit en premier lieu la suite $x(n, t)$ associée à l'événement x . Cette suite prend pour valeur des éléments de $\mathbb{B} = \{\varepsilon, e\}$. On pose

$$x(n_i, t) = \begin{cases} e & \text{si l'occurrence } n^\circ n_i \text{ de l'événement } x \text{ a lieu au plus tôt après la date } t \\ \varepsilon & \text{sinon.} \end{cases}$$

Exemple 14 *Considérons que l'occurrence numéro 2 (c'est-à-dire la troisième occurrence) de l'événement x a lieu à la date 3, on obtient alors la suite suivante :*

$$x(2, t) = \begin{cases} e & \text{si } t \leq 3 \\ \varepsilon & \text{si } t > 3. \end{cases}$$

Nous devons également tenir compte de la non décroissance de ces suites. En effet, si une occurrence numérotée n a lieu à la date t alors l'occurrence numérotée $n + 1$ intervient au plus tôt à la date t . Pour la suite $x(n, t)$, cela se caractérise par

$$x(n, t_i) = \begin{cases} e & \text{si à la date } t_i, \text{ une occurrence de } x \text{ a déjà eu lieu} \\ \varepsilon & \text{sinon.} \end{cases}$$

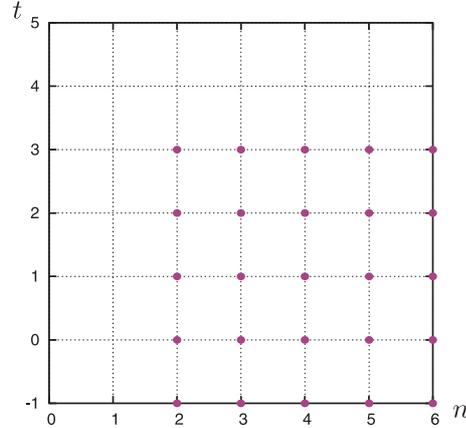


FIG. 2.3: Représentation graphique de la suite $x(n, t)$ considérée dans l'exemple 14.

Exemple 15 En reprenant l'exemple 14, il paraît logique que les occurrences de x dont le numéro est supérieur à 2 ne peuvent avoir lieu avant la date 3. On traduit cette propriété par

$$x(n, 3) = \begin{cases} e & \text{si } n \geq 2 \\ \varepsilon & \text{si } n < 2. \end{cases}$$

De façon formelle, on obtient la propriété suivante pour la suite $x(n, t)$:

Propriété 2 Soient x un événement et $x(n, t)$ la suite à coefficient dans \mathbb{B} qui lui est associée. Si $x(n_1, t_1) = e$ alors pour tout couple (n', t') tel que $n' \geq n_1$ et $t' \leq t_1$, on a $x(n', t') = e$.

Pour une occurrence d'événement $x(n_1, t_1)$ donnée, on ne doit donc pas considérer un élément de la suite $x(n, t)$ mais tout un ensemble de points (on parle aussi de "collection" de points dans [Cohen et al., 1989]) qui prend la forme d'un cône sud-est engendré par cette occurrence. On dit que le couple (n_1, t_1) domine les couples qui se situent dans le cône sud-est de sommet (n_1, t_1) . L'occurrence $x(n_1, t_1)$ peut s'interpréter de deux façons :

- l'occurrence numéro n_1 de l'événement x a lieu au plus tôt à l'instant t_1 ,
- à la date t_1 , il y a eu au moins n_1 occurrences de l'événement x .

Exemple 16 La suite considérée dans l'exemple 14 prend pour valeur e pour tout couple (n', t') tel que $n' \geq 2$ et $t' \leq 3$. On obtient la représentation graphique de la figure 2.3.

On s'intéresse maintenant à la transformée en γ, δ qui associe à la suite $x(n, t)$ la série formelle en deux indéterminées commutatives γ et δ à exposants dans \mathbb{Z} :

$$x = \bigoplus_{(n,t) \in \mathbb{Z}^2} x(n, t) \gamma^n \delta^t .$$

L'ensemble de ces séries formelles est un dioïde distributif noté $\mathbb{B}[\gamma, \delta]$ doté des règles d'addition et de produit définies dans l'exemple 7. Cependant, toutes les séries de $\mathbb{B}[\gamma, \delta]$ ne permettent pas de modéliser les transitions d'un GET. Étudions maintenant la forme d'une série qui code une transition de GET.

La représentation des séquences de tirs d'un GET induit les règles de simplification suivantes :

1.

$$\gamma^n \delta^t \oplus \gamma^{n'} \delta^t = \gamma^{\min(n, n')} \delta^t \quad (2.8)$$

2.

$$\gamma^n \delta^t \oplus \gamma^n \delta^{t'} = \gamma^n \delta^{\max(t, t')} \quad (2.9)$$

puisque d'une part $\gamma^{\min(n, n')} \delta^t$ domine $\gamma^n \delta^t$ et $\gamma^{n'} \delta^t$, et d'autre part $\gamma^n \delta^{\max(t, t')}$ domine $\gamma^n \delta^t$ et $\gamma^n \delta^{t'}$.

Exemple 17 Comme on peut le constater sur la figure 2.3, on a

$$\gamma^2 \delta^3 = \gamma^4 \delta^3 \oplus \gamma^2 \delta^3$$

et

$$\gamma^2 \delta^3 = \gamma^2 \delta^1 \oplus \gamma^2 \delta^3 .$$

En outre, puisqu'on considère chaque occurrence comme un cône, on a la relation suivante

$$\begin{aligned}
x &= \gamma^n \delta^t \\
&= \gamma^n \delta^t \oplus \gamma^{n+1} \delta^t \oplus \gamma^{n+2} \delta^t \oplus \dots \\
&\quad \oplus \gamma^n \delta^t \oplus \gamma^n \delta^{t-1} \oplus \gamma^n \delta^{t-2} \oplus \dots \\
&= \gamma^n \delta^t \left(\bigoplus_{i \in \mathbb{N}} \gamma^i \right) \oplus \gamma^n \delta^t \left(\bigoplus_{i \in \mathbb{N}} \delta^{-i} \right) \\
&= \gamma^n \delta^t \gamma^* \oplus \gamma^n \delta^t (\delta^{-1})^* \\
&= \gamma^n \delta^t (\gamma^* \oplus (\delta^{-1})^*).
\end{aligned}$$

Les éléments de $\mathbb{B}[[\gamma, \delta]]$ qui sont en mesure de coder des trajectoires de tir de GET sont ceux qui restent inchangés lorsqu'on les multiplie par $(\gamma^* \oplus (\delta^{-1})^*)$. Cet ensemble est un dioïde, noté $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$, dans lequel l'élément nul est noté $\varepsilon = \bigoplus_{(n,t) \in \mathbb{Z}^2} \varepsilon \gamma^n \delta^t$, l'élément unité est défini comme $e = (\gamma^* \oplus (\delta^{-1})^*)$. Le dioïde $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$ est complet et sa borne supérieure est notée $\top = (\gamma^{-1})^* \oplus \delta^*$.

Remarque 21 .

- L'élément neutre e peut présenter plusieurs notations. Tout d'abord $e = (\gamma^* \oplus (\delta^{-1})^*) = \gamma^* (\delta^{-1})^*$ puisque lorsque a et b sont des variables commutatives, on a $(a \oplus b)^* = a^* b^*$ (une preuve est donnée dans [Baccelli et al., 1992, lemme 4.107]). Ensuite, si l'on tient compte des propriétés (2.8) et (2.9), on peut également noter l'élément unité $e = \gamma^0 \delta^0 = \gamma^0 = \delta^0$.
- Puisqu'un monôme² $\gamma^n \delta^t$ de $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$ est invariant par rapport au produit par $\gamma^* (\delta^{-1})^*$, il existe une multitude de notations qui représentent ce cône. Nous considérerons généralement son représentant minimal, c'est-à-dire sa forme après avoir appliqué les règles de simplification (2.8) et (2.9).

2.4.2 Manipulation des séries formelles

La transformée en γ, δ d'un dateur associé à une transition d'un GET permet de coder l'ensemble de ses tirs (on parle alors de trajectoire). On peut s'interroger sur l'influence de la multiplication par γ ou δ sur une trajectoire. Débutons avec le produit par γ sur une série $x \in \mathcal{M}_{in}^{ax}[[\gamma, \delta]]$:

$$\begin{aligned}
\gamma x &= \gamma \bigoplus_{(n,t) \in \mathbb{Z}^2} x(n,t) \gamma^n \delta^t \\
&= \bigoplus_{(n,t) \in \mathbb{Z}^2} x(n,t) \gamma^{n+1} \delta^t \\
&= \bigoplus_{(n,t) \in \mathbb{Z}^2} x(n-1, t) \gamma^n \delta^t.
\end{aligned}$$

²On appelle monôme une série composée d'un seul élément du type $\gamma^n \delta^t$. Une série n'est rien d'autre qu'une somme de monômes.

Le résultat est un décalage d'une unité des numéros d'événements. Ce décalage est appelé "retard événementiel".

Exemple 18 En reprenant l'exemple 14, la trajectoire est donnée par $x = \gamma^2 \delta^3$ et le produit par γ conduit à $\gamma x = \gamma^3 \delta^3$. La trajectoire obtenue est telle que l'occurrence n°2 a lieu à l'instant $-\infty$ et l'occurrence n°3 à la date 3. Graphiquement, cela correspond à décaler d'une unité vers la droite le cône de la figure 2.3.

On examine maintenant l'effet du produit par δ sur une série x de $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$.

$$\begin{aligned} \delta x &= \delta \bigoplus_{(n,t) \in \mathbb{Z}^2} x(n,t) \gamma^n \delta^t \\ &= \bigoplus_{(n,t) \in \mathbb{Z}^2} x(n,t) \gamma^n \delta^{t+1} \\ &= \bigoplus_{(n,t) \in \mathbb{Z}^2} x(n,t-1) \gamma^n \delta^t. \end{aligned}$$

Le produit par δ induit un décalage temporel d'une unité de temps qui retarde toutes les occurrences d'une trajectoire. Ce décalage est aussi appelé "retard temporel".

Exemple 19 Après multiplication par δ , la trajectoire $x = \gamma^2 \delta^3$ de l'exemple 14, devient $\delta x = \gamma^2 \delta^4$. Dans la série obtenue, la troisième occurrence n'a plus lieu à la date 3 mais à la date 4. Cette manipulation décale la représentation graphique de la série x d'une unité vers le haut.

L'addition et le produit dans $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$ sont définis dans l'exemple 7. Puisque le dioïde $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$ est complet, tout couple $x = \bigoplus_{i \in I} \gamma^{n_i} \delta^{t_i}$ et $x' = \bigoplus_{j \in J} \gamma^{n'_j} \delta^{t'_j}$, où $\gamma^{n_i} \delta^{t_i}$ (resp. $\gamma^{n'_j} \delta^{t'_j}$) désignent les monômes du représentant minimal de x (resp. x'), admet une borne inférieure notée $x \wedge x'$ et définie par

$$\begin{aligned} x \wedge x' &= \left(\bigoplus_{i \in I} \gamma^{n_i} \delta^{t_i} \right) \wedge \left(\bigoplus_{j \in J} \gamma^{n'_j} \delta^{t'_j} \right) \\ &= \bigoplus_{i \in I} \left(\gamma^{n_i} \delta^{t_i} \wedge \left(\bigoplus_{j \in J} \gamma^{n'_j} \delta^{t'_j} \right) \right) \quad (\text{seconde égalité de la définition 13}) \\ &= \bigoplus_{i \in I} \left(\bigoplus_{j \in J} \left(\gamma^{n_i} \delta^{t_i} \wedge \gamma^{n'_j} \delta^{t'_j} \right) \right) \quad (\text{seconde égalité de la définition 13}). \end{aligned}$$

Pour deux monômes $\gamma^n \delta^t$ et $\gamma^{n'} \delta^{t'}$, on peut vérifier la règle de simplification suivante

$$\gamma^n \delta^t \wedge \gamma^{n'} \delta^{t'} = \gamma^{\max(n,n')} \delta^{\min(t,t')}.$$

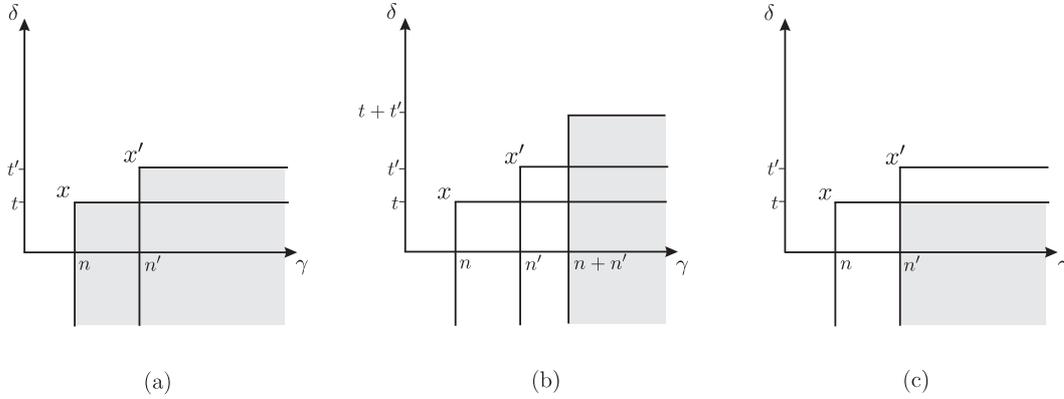


FIG. 2.4: Représentation graphique des opérations sur deux monômes.

Nous représentons graphiquement l'effet des lois \oplus , \otimes et \wedge sur deux monômes $x = \gamma^n \delta^t$ et $x' = \gamma^{n'} \delta^{t'}$ sur la figure 2.4 (le résultat de l'opération est représenté en gris) :

- (a) La somme de x et de x' est l'union de leur cône respectif.
- (b) Le produit des deux monômes est égal à $\gamma^{n+n'} \delta^{t+t'}$. La représentation graphique de ce produit se traduit par une somme vectorielle des points (n, t) et (n', t') .
- (c) La borne inférieure de x et de x' est représentée par l'intersection de leurs cônes respectifs.

Exemple 20 Nous considérons deux séries $x_1 = \gamma^1 \delta^2 \oplus \gamma^4 \delta^6$ et $x_2 = \delta^1 \oplus \gamma^2 \delta^2 \oplus \gamma^5 \delta^3$.

$$x_1 \oplus x_2 = \delta^1 \oplus \gamma^1 \delta^2 \oplus \gamma^4 \delta^6,$$

on remarque que $\gamma^4 \delta^6$ domine $\gamma^5 \delta^3$ ($\gamma^4 \delta^6 \succeq \gamma^5 \delta^3$).

$$x_1 \otimes x_2 = \gamma^1 \delta^3 \oplus \gamma^3 \delta^4 \oplus \gamma^4 \delta^7 \oplus \gamma^6 \delta^8 \oplus \gamma^9 \delta^9.$$

$$x_1 \wedge x_2 = \gamma^1 \delta^1 \oplus \gamma^2 \delta^2 \oplus \gamma^5 \delta^3.$$

La résiduation s'applique aussi dans le dioïde $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$. Les monômes de $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$ sont inversibles : $(\gamma^n \delta^t)^{-1} = \gamma^{-n} \delta^{-t}$. Le résultat de la résiduation de deux séries $x = \bigoplus_{i \in I} \gamma^{n_i} \delta^{t_i}$ et $x' = \bigoplus_{j \in J} \gamma^{n'_j} \delta^{t'_j}$ peut être évalué grâce à l'inversibilité des monômes et à la formule (f7) du tableau 1.1 :

$$\begin{aligned}
x \bowtie x' &= \left(\bigoplus_{i \in I} \gamma^{n_i} \delta^{t_i} \right) \bowtie x' \\
&= \bigwedge_{i \in I} \gamma^{n_i} \delta^{t_i} \bowtie x' && \text{(formule (f7) du tableau 1.1)} \\
&= \bigwedge_{i \in I} \gamma^{-n_i} \delta^{-t_i} x' && \text{(les monômes sont inversibles).}
\end{aligned}$$

Exemple 21 En reprenant les séries x_1 et x_2 définies dans l'exemple 20, on obtient

$$x_1 \bowtie x_2 = \gamma^{-1} \delta^{-4} \oplus \gamma^1 \delta^{-3}$$

et

$$x_2 \bowtie x_1 = \gamma^1 \oplus \gamma^2 \delta^1 \oplus \gamma^4 \delta^3.$$

La valuation et le degré d'une série, introduits dans l'exemple 7, se définissent dans $\mathcal{M}_{in}^{ax}[\gamma, \delta]$ respectivement par le plus petit exposant en γ du représentant minimal de la série et son plus grand exposant en δ (la valuation en δ et le degré en γ n'ont pas de sens, cf. [Baccelli et al., 1992, §5.4.2.3]). L'exemple qui suit illustre ces calculs.

Exemple 22 On considère les séries x_1 et x_2 de l'exemple 20. Les calculs de la valuation et du degré donnent :

$$\begin{aligned}
val(x_1) &= 1 & deg(x_1) &= 6 \\
val(x_2) &= 0 & deg(x_2) &= 3.
\end{aligned}$$

Plus précisément, la valuation est définie comme l'application suivante

$$\begin{aligned}
val &: \mathcal{M}_{in}^{ax}[\gamma, \delta] &\longrightarrow & \bar{\mathbb{Z}}_{min} \\
x &&\longmapsto & val(x).
\end{aligned}$$

On vérifie facilement que l'application val est s.c.i. (voir [Baccelli et al., 1992, Lemme 4.93]) puisque

$$val\left(\bigoplus_{x \in \mathcal{C}} x\right) = \bigoplus_{x \in \mathcal{C}} val(x).$$

Il faut néanmoins garder à l'esprit que le signe \oplus du terme de gauche n'a pas la même signification que celui de droite. En effet, dans $val(x_1 \oplus x_2)$, le signe \oplus représente la somme au

sens de $\mathcal{M}_{in}^{ax}[\gamma, \delta]$ tandis que celui de $val(x_1) \oplus val(x_2)$ correspond à celui de $\overline{\mathbb{Z}}_{min}$, à savoir l'opérateur min .

Exemple 23 Avec les séries $x_1 = \gamma^1 \delta^2 \oplus \gamma^4 \delta^6$ et $x_2 = \delta^1 \oplus \gamma^2 \delta^2 \oplus \gamma^5 \delta^3$, on peut vérifier que $val(x_1 \oplus x_2) = val(x_1) \oplus val(x_2)$:

$$\begin{aligned} val(x_1 \oplus x_2) &= val(\delta^1 \oplus \gamma^1 \delta^2 \oplus \gamma^4 \delta^6) = 0 \\ val(x_1) \oplus val(x_2) &= 1 \oplus 0 = 0. \end{aligned}$$

En outre, on a $val(\varepsilon) = \varepsilon$ (attention, là encore, les deux ε n'ont pas le même sens). En accord avec le point (ii) du théorème 8, on peut affirmer que la valuation est une application résiduable.

Proposition 9 L'application val est résiduable et sa résiduée est donnée par

$$val^\sharp(x) = \gamma^x \delta^*$$

Preuve : On a prouvé précédemment que val était résiduable, il reste à montrer que $val^\sharp(x) = \gamma^x \delta^*$. Il suffit pour cela de vérifier que l'application satisfait le troisième point du théorème 8, soit :

$$\begin{aligned} val \circ val^\sharp(x) &= val(\gamma^x \delta^*) = x \\ val^\sharp \circ val(x) &= \gamma^{val(x)} \delta^* \succeq x. \end{aligned}$$

□

De façon générale, on pourra toujours mettre une série codant les tirs d'une transition d'un GET sous la forme :

$$x = p \oplus qr^*,$$

avec

$$p = \bigoplus_{i=0}^{\alpha} \gamma^{n_i} \delta^{t_i} \quad q = \bigoplus_{j=0}^{\beta} \gamma^{N_j} \delta^{T_j}$$

et r un monôme $\gamma^\nu \delta^\tau$.

Cette forme est appelée forme *périodique* en raison du terme r^* qui, quand il est différent de e , conduit à un comportement périodique de x à partir du monôme $\gamma^{n\alpha}\delta^{t\alpha}$. On définit maintenant une autre propriété des séries : la causalité.

Définition 23 (Causalité) Une série x de $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$ est dite causale si $x = \varepsilon$ ou si son représentant minimal possède tous ses exposants dans \mathbb{N} . Une matrice est dite causale si toutes ses composantes sont causales.

Remarque 22 Il existe une autre définition de la causalité ([Baccelli et al., 1992, §5.7.2]) équivalente à celle exposée ci-dessus. Elle se résume à faire remarquer que la série x est causale si $x = \varepsilon$ ou si $val(x) \geq 0$ et $x \geq \gamma^{val(x)}$.

L'ensemble des séries causales est un sous-dioïde noté $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]^+$. Pour une série x de $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$, il est possible d'obtenir le plus grand élément de $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]^+$ qui soit plus petit que x . Cette opération appelée *projection causale* (voir [Cottenceau, 1999, §2.6]) est notée Pr_+ .

Définition 24 On définit la projection causale d'un élément $x \in \mathcal{M}_{in}^{ax}[[\gamma, \delta]]$ comme suit :

$$\begin{aligned} Pr_+(x) &= Pr_+(\bigoplus_{i \in I} x(n_i, t_i) \gamma^{n_i} \delta^{t_i}) \\ &= \bigoplus_{i \in I} x_+(n_i, t_i) \gamma^{n_i} \delta^{t_i}, \end{aligned}$$

$$\text{où } x_+(n_i, t_i) = \begin{cases} x(n_i, t_i), & \text{si } (n_i, t_i) \geq (0, 0); \\ \varepsilon, & \text{sinon.} \end{cases}$$

Graphiquement, la projection causale consiste à ne conserver que les monômes situés dans le cadran nord-est.

Remarque 23 Il faudra, par la suite, garder à l'esprit que si une série x est causale alors elle est invariante par Pr_+ , i.e. $Pr_+(x) = x$.

Exemple 24 Pour la série $x = \gamma^{-1}\delta^{-2} \oplus \delta^1 \oplus \gamma^3\delta^2 \oplus \gamma^4\delta^4$ on a $Pr_+(x) = \delta^1 \oplus \gamma^3\delta^2 \oplus \gamma^4\delta^4$.

On introduit également la projection Pr_a .

Définition 25 Soit $Pr_a : \mathcal{M}_{in}^{ax}[\gamma, \delta] \rightarrow \mathcal{M}_{in}^{ax}[\gamma, \delta]$ l'application définie par

$$\begin{aligned} Pr_a(x) &= Pr_a\left(\bigoplus_{i \in I} x(n_i, t_i) \gamma^{n_i} \delta^{t_i}\right) \\ &= \bigoplus_{i \in I} x_a(n_i, t_i) \gamma^{n_i} \delta^{t_i}, \end{aligned}$$

$$\text{où } x_a(n_i, t_i) = \begin{cases} x(n_i, t_i), & \text{si } t_i \geq a; \\ \varepsilon, & \text{sinon.} \end{cases}$$

Pour une série $x \in \mathcal{M}_{in}^{ax}[\gamma, \delta]$, l'application Pr_a consiste à ne conserver que les monômes de x dont l'exposant en δ est supérieur ou égal à a .

Exemple 25 Pour la série $x = \gamma^{-1}\delta^{-2} \oplus \delta^1 \oplus \gamma^3\delta^2 \oplus \gamma^4\delta^4$ on a $Pr_2(x) = \gamma^3\delta^2 \oplus \gamma^4\delta^4$.

Dans la proposition suivante, on montre que l'application Pr_a est résiduable.

Proposition 10 L'application $Pr_a : \mathcal{M}_{in}^{ax}[\gamma, \delta] \rightarrow \mathcal{M}_{in}^{ax}[\gamma, \delta]$ est résiduable et sa résiduée est définie par $Pr_a^\sharp(x) = x \oplus (\gamma^{-1})^* \delta^{a-1}$.

Preuve : L'application Pr_a est résiduable puisque, en accord avec le point (ii) du théorème 8, on a $Pr_a(\varepsilon) = \varepsilon$ et Pr_a est semi-continue inférieurement. On peut ensuite vérifier que le point (iii) du théorème 8 est respecté pour $Pr_a^\sharp(x) = x \oplus (\gamma^{-1})^* \delta^{a-1}$. En effet, on a

$$\begin{aligned} Pr_a \circ Pr_a^\sharp(x) &= Pr_a(x \oplus (\gamma^{-1})^* \delta^{a-1}) \\ &= Pr_a(x) \preceq x \\ Pr_a^\sharp \circ Pr_a(x) &= Pr_a^\sharp(Pr_a(x)) \\ &= Pr_a(x) \oplus (\gamma^{-1})^* \delta^{a-1} \succeq x \end{aligned}$$

□

Certains éléments non causaux de $\mathcal{M}_{in}^{ax}[\gamma, \delta]$ possèdent une sur-approximation dans $\mathcal{M}_{in}^{ax+}[\gamma, \delta]$. Nous exposons maintenant la notion de *sur-approximation causale* introduite dans [Houssin et al., 2006a] et mettons à jour les conditions d'existence de cette approximation.

Proposition 11 Soit $x \in \mathcal{M}_{in}^{ax}[\gamma, \delta]$. Sont équivalents

- (i) le représentant minimal de x ne possède pas d'exposants en γ négatifs,
(ii) il existe un plus petit $x' \in \mathcal{M}_{in}^{ax+}[[\gamma, \delta]]$ tel que $x' \succeq x$. L'élément x' est alors appelé sur-approximation causale de x .

Preuve : Si x est causal, la démonstration est évidente et on a $x' = x$. Considérons maintenant le cas où x n'est pas causal. On décompose chaque série x en une partie causale x_c et une partie non causale $x_{\bar{c}}$: $x = x_c \oplus x_{\bar{c}}$.

(i) \Rightarrow (ii) : Si x n'a pas d'exposants en γ négatifs, on peut écrire $x_{\bar{c}} \preceq \gamma^{val(x)}\delta^0$. De plus, $\gamma^{val(x)}\delta^0$ est le plus petit élément de $\mathcal{M}_{in}^{ax+}[[\gamma, \delta]]$ qui soit plus grand que $x_{\bar{c}}$. On a donc logiquement $x' = \gamma^{val(x)}\delta^0 \oplus x_c$.

(ii) \Rightarrow (i) : S'il existe un plus petit $x' \in \mathcal{M}_{in}^{ax+}[[\gamma, \delta]]$ tel que $x' \succeq x$ alors $0 \leq val(x')$ et $val(x') \leq val(x)$. La série x ne comporte donc pas d'exposant négatif en γ .

□

Remarque 24 La librairie [SWtool, 2005b] (en C++), développée au LISA, constitue un outil adapté pour la manipulation des séries de $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$. Diverses fonctions sont implémentées pour calculer, entre autres, la somme, le produit, la résiduation du produit, de séries ou de matrices de séries rationnelles.

2.4.3 Modélisation des graphes d'événements temporisés dans le dioïde des séries formelles

L'objet de cette section est de rappeler que tout transfert de GET peut se représenter par une matrice constituée de séries de $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$.

L'équation d'état (2.3) devient dans $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$:

$$\begin{cases} x &= Ax \oplus Bu \\ y &= Cx, \end{cases} \quad (2.10)$$

où A , B et C sont des matrices à éléments dans $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$.

Exemple 26 Le modèle d'état de la remarque 17 conduit à la représentation suivante dans $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$:

$$\left\{ \begin{array}{l} x = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \gamma\delta^8 & \varepsilon & \gamma\delta^2 & \gamma\delta^4 \\ \varepsilon & \gamma & \varepsilon & \varepsilon \end{pmatrix} x \oplus \begin{pmatrix} e & \varepsilon \\ \varepsilon & e \\ \varepsilon & \varepsilon \\ \varepsilon & \varepsilon \end{pmatrix} u \\ y = \begin{pmatrix} \varepsilon & \varepsilon & e & \varepsilon \end{pmatrix} x. \end{array} \right.$$

Notons que, pour obtenir la représentation d'un GET dans $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$, il n'est pas nécessaire de d'abord traduire le fonctionnement du système dans $\overline{\mathbb{Z}}_{max}$ ou $\overline{\mathbb{Z}}_{min}$. On peut directement établir des relations entre les variables d'état, d'entrée et de sortie dans $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$.

Si on s'intéresse au fonctionnement au plus tôt du système, on utilise le corollaire 1 pour trouver la plus petite solution de (2.10) :

$$x = A^*Bu.$$

On parvient alors au fonctionnement entrée-sortie suivant :

$$y = CA^*Bu = Hu, \quad (2.11)$$

où H est appelé la *matrice de transfert* du système. On voit ici une propriété importante de la transformée en γ, δ : les produits de convolution dans $\overline{\mathbb{Z}}_{max}$ deviennent des multiplications dans $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$. En effet, le produit de convolution de l'équation (2.7) a évolué en un simple produit dans (2.10). On note que la matrice de transfert H n'est rien d'autre que la transformée en γ, δ de la réponse impulsionnelle du système.

Exemple 27 *Le calcul de la matrice de transfert à partir de la représentation d'état de l'exemple 26 donne :*

$$H = \begin{pmatrix} \gamma\delta^8(\gamma\delta^2)^* \\ \gamma^2\delta^4(\gamma\delta^2)^* \end{pmatrix}.$$

Dans la suite de ce chapitre, nous nous intéressons à la synthèse de commande de systèmes $(max, +)$ -linéaires. Dans certains cas, des correcteurs sont utilisés. Nous présentons maintenant la notion de *réalisabilité*. La validité des correcteurs repose sur cette définition.

Définition 26 Un élément x de $\mathcal{M}_{in}^{ax}[\gamma, \delta]$ est dit réalisable s'il existe un GET mono-entrée/mono-sortie dont cet élément est la fonction de transfert, ou plus précisément, s'il existe trois matrices A , B et C à éléments dans $\mathcal{M}_{in}^{ax}[\gamma, \delta]$ telles que x puisse s'écrire $x = CA^*B$.

Le théorème suivant établit la relation entre réalisabilité et causalité.

Théorème 15 [Cohen et al., 1989] Soit $x \in \mathcal{M}_{in}^{ax}[\gamma, \delta]$. Sont équivalents

- (i) x est réalisable.
- (ii) x est périodique et causale.

2.5 Commande de systèmes $(max, +)$ -linéaires

De façon classique, on entend par *commande* d'un système $(max, +)$ -linéaire le pilotage de ses entrées en vue d'obtenir un comportement spécifié pour le système. Plus concrètement, il s'agit ici de fixer les dates d'occurrences des événements d'entrée pour les SED considérés, ou encore, de fixer les dates de tir des transitions d'entrée (transition sources) pour les GET correspondant. Depuis une quinzaine d'années, plusieurs approches complémentaires ont été proposées.

Avant de présenter dans le chapitre suivant notre contribution sur le sujet, nous proposons dans cette section un bref état de l'art sur la commande des systèmes $(max, +)$ -linéaires. Celui-ci ne se veut en aucun cas exhaustif et le choix des approches présentées est partial (il vise notamment à faciliter la présentation de notre contribution). Nous avons pris le parti de caractériser chaque commande selon trois points :

- la définition d'un critère à optimiser,
- la spécification d'objectifs de commande,
- la structure adoptée pour la réalisation de la commande.

2.5.1 Critères

Un critère est généralement défini par une fonction, appelée *fonction coût*, que l'on cherche à optimiser.

- Les premiers travaux concernant le pilotage des systèmes $(max, +)$ -linéaires ont considéré le critère de *juste-à-temps*³. La problématique du juste-à-temps est d'activer les événements d'entrée aux dates les plus tardives tout en satisfaisant certains objectifs de

³Juste-à-temps est la traduction littérale du terme anglais "just-in-time".

contrôle. Plus précisément, dans les études sur la commande des GET, le principe du juste-à-temps vise à

- effectuer le nombre minimum de tirs des transitions d'entrée,
 - retarder au maximum le tir de ces transitions,
- tout en satisfaisant des objectifs de commande.

La fonction coût sous-jacente que l'on cherche à maximiser quantifie donc les dates d'occurrences des événements d'entrée, ce qui a pour conséquence, si on s'intéresse à un système de production, de minimiser les en-cours ou stocks internes. Dans le milieu industriel, on parle de production en flux tendus (par opposition à la production en flux poussés), qui consiste à produire la quantité juste nécessaire au moment où l'on en a besoin.

- Plus récemment, un critère différent a été considéré notamment dans [De Schutter and van den Boom, 2000]. La commande utilisée correspond à l'adaptation de la commande prédictive aux systèmes $(max, +)$ -linéaires. Le critère usuel (à minimiser) de la commande prédictive se décompose en une somme pondérée de deux sous-critères qui correspondent, d'une part, à l'erreur quadratique entre une trajectoire de référence connue et la sortie prédite du système, et d'autre part, à l'amplitude des variations de la commande (en vue de "ménager" les actionneurs). Ce dernier sous critère peut éventuellement être remplacé par l'énergie de la commande. Pour les systèmes $(max, +)$ -linéaires, si le critère d'erreur de sortie vis-à-vis d'une trajectoire de référence reste d'actualité, minimiser l'énergie de la commande ou les variations de celle-ci ne présente pas d'intérêt. Ce sous-critère a donc été substitué par une fonction coût qui correspond à l'inverse de la somme des dates de tirs de toutes les entrées, soit

$$J_{in} = - \sum_{j=1}^{N_p} \sum_{i=1}^m u_i(k + j - 1),$$

pour un système à m entrées et où N_p désigne à l'horizon de prédiction. Le but de ce sous critère est de maximiser la somme des dates d'activations des événements d'entrées (en ce sens, ce sous critère se rapproche de celui du juste-à-temps exposé plus tôt). Un critère global considéré par [De Schutter and van den Boom, 2000] est donné par :

$$J = J_{out} + \lambda J_{in},$$

où J_{out} représente le critère de poursuite de trajectoire et λ est une variable scalaire utilisée pour pondérer le critère de commande.

2.5.2 Objectifs de commande

Les objectifs de commande qui ont été considérés pour les systèmes $(max, +)$ -linéaires sont proches de ceux de la théorie classique de la commande des systèmes. On en recense ici trois :

- *La poursuite de trajectoire.* Cette commande vise à synthétiser une trajectoire d'entrée de sorte que la réponse en sortie soit la plus proche possible d'une trajectoire de référence, également appelée consigne. Celle-ci correspond à un comportement désiré en sortie du système. Lorsque le critère du juste-à-temps est considéré, l'objectif de commande revient à calculer une trajectoire d'entrée de sorte que la sortie obtenue soit plus petite ou égale à une trajectoire de référence. Autrement dit, dans le cas du juste-à-temps, le problème de poursuite de trajectoire considéré consiste à rechercher la plus grande commande u telle que $Hu \preceq z$, où z est la trajectoire de consigne et H la matrice de transfert du système. Cet objectif particulier induit que la sortie obtenue ne dépasse jamais la trajectoire de référence.
- *La poursuite de modèle.* L'objectif est ici d'établir un correcteur modifiant le comportement dynamique du système en vue d'obtenir pour le système corrigé des performances spécifiées sous la forme d'un modèle dit de référence (généralement exprimé dans la littérature sous la forme d'une matrice de transfert dans le dioïde $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$, cf. équation (2.10)).
- *Le rejet de perturbation.* Dans un processus, les perturbations peuvent avoir des origines diverses : bruit du système, dynamique non gouvernable (dans ce cas, on parle aussi d'entrées non contrôlables). Dans la théorie classique des systèmes, le problème du rejet de perturbation se résume à synthétiser une commande telle que la perturbation n'ait pas d'influence sur la sortie du système. Pour les SED considérés ici, il n'est pas possible de trouver une commande qui puisse annuler l'effet de la perturbation sur la sortie du système. En effet, une perturbation agit comme un retard sur la sortie et l'action de commander un système $(max, +)$ -linéaire se réduit à provoquer des retards sur les occurrences des événements internes au système (alors que pour annuler l'effet de la perturbation, il faudrait "accélérer" les dates de tirs). Le rejet de perturbation se révèle donc être un problème différent, à savoir : trouver une commande qui maintient la sortie inchangée au regard de celle générée par la perturbation. Le but de cette commande est de ne pas produire de retard supplémentaire de la sortie par rapport à ceux introduits par la perturbation.

Considérons la représentation $(max, +)$ -linéaire suivante :

$$\begin{cases} x &= Ax \oplus Bu \oplus Sq \\ y &= Cx, \end{cases} \quad (2.12)$$

où le vecteur q regroupe des entrées non maîtrisables. Les éléments de q ont pour effet de ralentir les occurrences de certaines variables d'état. Le fonctionnement au plus tôt du système perturbé nous dicte le transfert :

$$y = CA^*Bu \oplus CA^*Sq.$$

Formellement, l'objectif du rejet de perturbation est de trouver une commande u telle que

$$CA^*Sq = CA^*Bu \oplus CA^*Sq.$$

2.5.3 Structures de commande et synthèse

Commandes en boucle ouverte

En boucle ouverte, on applique une valeur de commande à l'entrée du système considéré, et on suppose que sa sortie réagira de façon connue en fonction de cette valeur, sans contrôler le bon déroulement de cette action. Ce mode de commande ne nécessite aucune mesure des variables du système.

- Concernant les systèmes $(max, +)$ -linéaires, une structure de commande en boucle ouverte a été proposée en premier lieu dans [Cohen et al., 1989]. Dans cette étude, le critère considéré est le juste-à-temps, et l'objectif de commande se traduit par une poursuite de trajectoire de référence. Nous rappelons ici la méthode de synthèse utilisée pour l'expliquer.

Nous avons vu au paragraphe 2.4.3 que la représentation entrée-sortie d'un GET peut s'écrire dans $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$ sous la forme

$$y = H \otimes u,$$

où $H = CA^*B$. De façon équivalente, on peut écrire

$$y = L_H(u)$$

avec

$$L_H : x \longmapsto H \otimes x .$$

Le problème de poursuite de trajectoire considéré consiste à rechercher la plus grande commande u telle que $L_H(u) \preceq z$, où z est la trajectoire de référence. La plus grande solution, notée u_{opt} , s'écrit formellement :

$$u_{opt} = \bigoplus_{\{u \in \mathcal{D} \mid L_H(u) \preceq z\}} u.$$

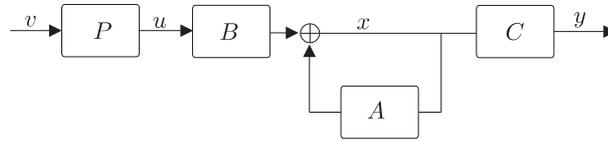


FIG. 2.5: Correction d'un système par un precompensateur.

On se ramène donc à un problème de pseudo-inversion de l'application L_H . La théorie de la résiduation permet de le résoudre de façon optimale (voir §1.5).

Théorème 16 *La commande u_{opt} existe et est donnée par $u_{opt} = H \setminus z$.*

Différents travaux ont permis d'étendre la synthèse de cette commande aux cas où :

- des entrées sont non maîtrisables [Menguy et al., 2000b]
- la consigne peut être réactualisée [Menguy et al., 2000b]
- le système est non stationnaire [Lahaye, 2000]
- le système est modélisé par un graphe d'événements valué hybride [Hamaci et al., 2006].

- Dans [Lhommeau, 2003, §5.3], la synthèse d'une commande en boucle ouverte est proposée pour le rejet de perturbation en juste-à-temps.
- La commande en boucle ouverte peut aussi se présenter sous la forme d'un précompensateur adjoint au système nominal (*cf.* figure 2.5). Avec pour critère le juste-à-temps, et pour objectif de commande la poursuite de modèle, les travaux [Libeaut and Loiseau, 1996] [Cottenceau, 1999][Hardouin et al., 1997] élaborent un plus grand précompensateur de transfert P de sorte que le transfert $H \otimes P$ du système contrôlé (où H est le transfert nominal du système) soit inférieur ou égal au transfert G_{ref} du modèle de référence. Le problème se pose alors de la façon suivante : trouver le plus grand précompensateur P de sorte que $H \otimes P \preceq G_{ref}$. La théorie de la résiduation permet de calculer le plus grand précompensateur $H \setminus G_{ref}$, cependant il n'est pas toujours causale. La projection causale de $H \setminus G_{ref}$ est également solution de $H \otimes P \preceq G_{ref}$ et elle constitue le plus grand précompensateur causal.
- Dans [De Schutter and van den Boom, 2000], une adaptation aux systèmes $(max, +)$ -linéaires de la commande prédictive est proposée. Le critère considéré a été décrit dans la partie 2.5.1 et la synthèse de la commande fait appel à des techniques d'optimisation convexe. De plus, dans [van den Boom and De Schutter, 2001], les auteurs discutent de l'influence des paramètres utilisés (horizon de prédiction, horizon de commande, pondération λ du critère) sur la commande réalisée.

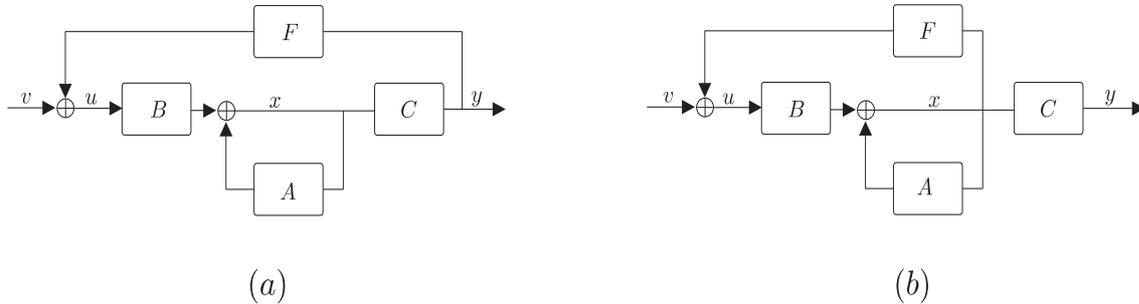


FIG. 2.6: (a) : Correction d'un système par un retour de sortie. (b) : Correction d'un système par un retour d'état.

Commandes en boucle fermée

Une autre structure, empruntée à l'automatique classique et utilisée pour la commande des systèmes $(max, +)$ -linéaires, est la boucle de retour ou feedback. Cette structure fait apparaître dans le calcul de la loi de commande la sortie ou l'état du système.

Nous rappelons ici différentes structures de commande en boucle fermée qui ont été utilisées pour les SED $(max, +)$ -linéaires.

- Le retour de sortie nécessite de connaître à tout instant la valeur de la sortie du système. Un système corrigé par un retour de sortie $u = v \oplus Fy$ (voir figure 2.6(a)) vérifie

$$\begin{cases} x &= Ax \oplus B(v \oplus Fy) \\ y &= Cx. \end{cases}$$

En considérant le fonctionnement au plus tôt, le système corrigé possède la dynamique suivante

$$x = A^*Bv \oplus A^*BFy.$$

Par conséquent, l'équation de sortie devient alors

$$y = CA^*Bv \oplus CA^*BFy.$$

Finalement, en sélectionnant la plus petite solution (celle qui correspond au fonctionnement au plus tôt) de l'équation implicite en y , on obtient

$$y = (CA^*BF)^*CA^*Bv = (HF)^*Hv.$$

La dernière égalité s'obtient en notant $H = CA^*B$ la matrice de transfert du système non corrigé. Cette structure de commande a été initialement considérée dans

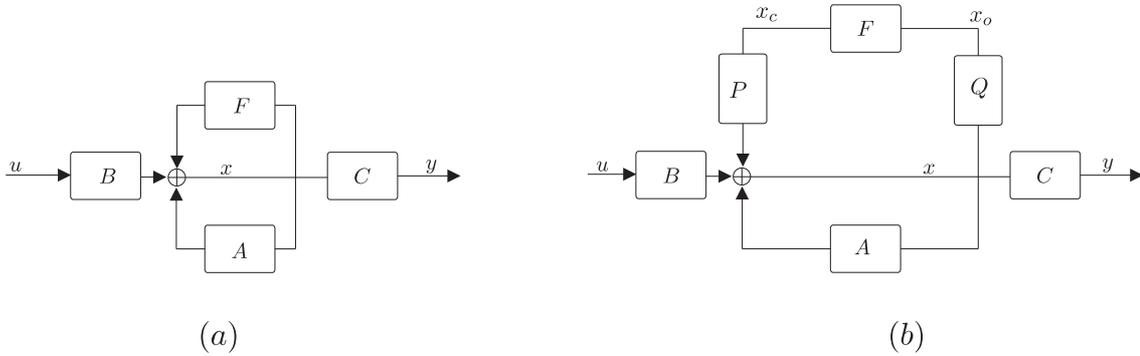


FIG. 2.7: (a) : Correction d'un système par un retour d'état sur l'état. (b) : Correction d'un système partiellement observable et partiellement commandable par un retour d'état sur l'état.

[Cottenceau et al., 1999] puis [Cottenceau et al., 2001] à des fins de poursuites de modèle en juste-à-temps. Plus précisément, le problème traité est de trouver le plus grand feedback F tel que $(HF)^*H$ (le transfert du système corrigé) soit inférieur ou égal à G_{ref} le transfert du modèle de référence. A partir d'outils issus de la théorie de la résiduation, un feedback a été mis à jour sous certaines conditions sur le modèle de référence. Cette structure a également été employée dans [Lhommeau et al., 2003] pour traiter le rejet de perturbation en juste-à-temps.

- Concernant la correction par retour d'état (figure 2.6(b)), on procède de la même façon pour dégager le transfert. Le modèle d'état du système corrigé est donné par :

$$\begin{cases} x &= Ax \oplus B(v \oplus Fx) \\ y &= Cx. \end{cases}$$

La plus petite solution de système permet de faire apparaître le transfert du système bouclé

$$y = C(A \oplus BF)^* Bv.$$

Ce type de correction nécessite que l'état du système soit mesurable.

Cette structure a aussi été étudié dans [Cottenceau et al., 2001] et [Lhommeau, 2003] pour résoudre respectivement les problèmes de poursuite de modèle (avec des conditions sur le modèle de référence) et de rejet de perturbation en juste-à-temps.

- Une structure en boucle fermée adoptée par [Lüders and Santos-Mendes, 2002] est le retour d'état sur l'état (figure 2.7(a)). Pour cette structure de commande, il est pris pour hypothèse que les événements internes du systèmes sont mesurables et que le correcteur

a la possibilité de retarder ces événements. Nous présentons ci-dessous la représentation d'état du système corrigé

$$\begin{cases} x &= Ax \oplus Bu \oplus Fx \\ y &= Cx. \end{cases}$$

Le transfert du système corrigé est explicité dans la relation entrée-sortie

$$y = C(A \oplus F)^* Bu.$$

En outre, dans [Lüders and Santos-Mendes, 2002], les auteurs considèrent le cas où des variables d'état ne sont pas commandables ou observables. Dans ce contexte, deux matrices P et Q sont introduites pour représenter l'accès partiel au vecteur d'état. La matrice P définit l'action de la commande sur le vecteur d'état, et la matrice Q représente comment le vecteur d'état est observé. La figure 2.7(b) illustre cette forme de correction. Le système est alors décrit par $x = Ax \oplus Bu \oplus Px_c$ où $x_c = Fx_o$ est le retour d'état fourni par le correcteur F et x_o est la version observée du vecteur d'état. La représentation d'état du système contrôlé est alors

$$\begin{cases} x &= Ax \oplus Bu \oplus PFQx \\ y &= Cx. \end{cases}$$

Dans [Lüders and Santos-Mendes, 2002], l'objectif de commande est la poursuite de modèle vis-à-vis du critère de juste-à-temps. Les résultats reposent essentiellement sur la théorie de la résiduation.

- Une structure plus générale a été considérée par [Maia et al., 2003] où la correction est assurée par l'utilisation simultanée d'un précompensateur et d'un retour de sortie. La figure 2.8 illustre cette approche.

Cette étude répond aussi au problème de poursuite de modèle en juste-à-temps. Le calcul des correcteurs découle de résultats sur la théorie de la résiduation. Par rapport aux travaux sur le retour de sortie dans [Cottenceau et al., 1999], aucune hypothèse sur le modèle de référence n'est requise pour l'obtention des correcteurs optimaux.

- Mentionnons également l'étude [Boimond and Ferrier, 1996] qui peut être considérée comme de la commande en boucle fermée de systèmes $(max, +)$ -linéaires. S'inspirant de la structure de commande à modèle interne⁴, les auteurs résolvent en utilisant la théorie de la résiduation un problème de poursuite de trajectoire en juste-à-temps.

⁴La structure de commande à modèle interne est parfois qualifiée de commande en boucle ouverte généralisée.

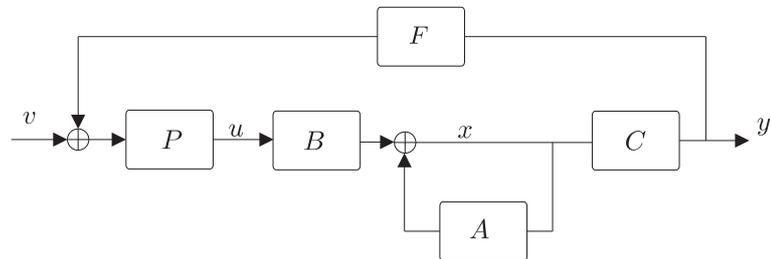


FIG. 2.8: Correction d'un système par un précompensateur et un retour de sortie.

- Une transposition de la commande adaptative indirecte a été proposée dans [Menguy et al., 2000a]. L'objectif de commande considéré est la poursuite de trajectoire vis-à-vis du critère de juste-à-temps. Les auteurs font appel à des techniques de résiduation pour résoudre le problème posé.

Le lecteur pourra trouver dans [Hardouin, 2004] une synthèse plus détaillée sur le calcul de certains des correcteurs exposés ci-dessus. On signale également que les travaux de [Lhommeau, 2003] concernent la robustesse de certaines commandes exposées précédemment et l'auteur met à jour une commande robuste pour les systèmes $(max, +)$ -linéaires incertains.

Chapitre 3

Contribution à la commande des systèmes $(max, +)$ -linéaires

Dans la partie 2.5, nous avons proposé un état de l'art sur la commande des systèmes $(max, +)$ -linéaires. Ce chapitre est consacré à nos apports sur le sujet. Plus précisément, sont introduites, ici, deux nouvelles approches de commande.

- La première contribution consiste en une commande en boucle ouverte en juste-à-temps. L'originalité réside dans la prise en compte d'un nouvel objectif de commande. En effet, cette commande aura pour rôle de satisfaire un ensemble de contraintes définies sur les variables internes du système. A l'opposé des commandes en juste-à-temps proposées jusque-là, celle-ci se démarque en ne considérant plus un objectif de commande sous la forme d'un transfert entrée-sortie de référence ou une sortie de référence, mais par un ensemble de contraintes sur l'état du système. Ces contraintes permettent de définir une spécification plus précise du comportement désiré du système. Les résultats sur les points fixes d'applications isotones définies sur des dioïdes, décrits au §1.6.1, sont utilisés pour la synthèse de cette commande.
- L'autre apport correspond à une commande en boucle fermée dont l'objectif est également spécifié sous la forme de contraintes sur l'état du système. Cependant, à l'inverse du critère en juste-à-temps qui consiste à ralentir au maximum le système, on prend en compte un critère original qui vise à minimiser les retards induits par la commande. En d'autres termes, on cherche le plus petit transfert du système corrigé tel que l'état satisfasse un ensemble de contraintes. Le problème de synthèse du correcteur se formalise comme un problème de recherche d'un point préfixe minimal d'une application antitone. Les développements, à notre connaissance originaux, du §1.6.2 trouvent leur utilité dans la synthèse de cette correction.

Le chapitre débute par une section dans laquelle on s'attache à mettre en relief et expliciter les originalités mises en oeuvre dans les approches de commande proposées.

3.1 Originalités des commandes proposées

Dans cette section, nous mettons en relief les originalités mises en oeuvre dans les approches de commande proposées dans ce chapitre. Nous spécifions tout d'abord un nouvel objectif de commande. Nous présentons ensuite un critère original pour le contrôle des systèmes $(max, +)$ -linéaires.

3.1.1 Un nouvel objectif de commande

Dans la section 2.5, nous avons recensé les objectifs de commande qui, jusqu'à présent, ont été employés pour la commande des systèmes $(max, +)$ -linéaires, à savoir la poursuite de trajectoire, la poursuite de modèle ainsi que le rejet de perturbation. Nous introduisons ici une nouvelle spécification de commande pour ces systèmes. Celle-ci consiste à imposer des contraintes qui s'exercent sur l'état du système. Formellement, si le système $(max, +)$ -linéaire considéré est représenté dans le dioïde des séries formelles $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$, cet objectif de commande revient à imposer que l'état satisfasse l'inéquation

$$x \preceq f(x), \quad (3.1)$$

ou l'inéquation

$$x \succeq f(x), \quad (3.2)$$

où $f : \mathcal{M}_{in}^{ax}[[\gamma, \delta]]^n \rightarrow \mathcal{M}_{in}^{ax}[[\gamma, \delta]]^n$ est une application isotone.

En considérant ce type d'inégalités sur les variables internes du systèmes, on est en mesure de spécifier finement les aspects de la dynamique souhaitée pour le système corrigé. En effet, ces contraintes sur l'état permettent d'exprimer des éléments de dynamique qui ne se formulent pas (ou très difficilement) en termes de transfert du système. A notre connaissance, il n'est notamment pas possible de les considérer au travers d'un modèle entrée-sortie de référence ou d'une trajectoire à poursuivre en sortie du système.

Afin de motiver, plus encore, ce formalisme et illustrer l'intérêt vis-à-vis des SED, nous explicitons dans la suite de cette section un certain nombre de contraintes qui peuvent s'exprimer sous la forme (3.1) et/ou (3.2). Nous détaillons là le "jeu" de contraintes que nous serons amenés à utiliser pour les applications au chapitre suivant. Il ne se veut en aucun cas être une

liste exhaustive des contraintes pouvant se mettre sous les formes (3.1) ou (3.2).

- Dans certains systèmes, on peut souhaiter imposer un laps de temps minimum entre deux occurrences d'un même événement. Pour un système appréhendé sous la forme d'un GET, cela revient à assurer une fréquence maximale pour les tirs d'une même transition. A partir d'une représentation en dateurs, cette contrainte s'écrit dans $\overline{\mathbb{Z}}_{max}$

$$x_i(k+1) \succeq \Delta_i^{min} x_i(k),$$

où x_i est l'événement sujet à la contrainte et Δ_i^{min} est le temps minimum désiré entre deux occurrences successives de x_i .

Dans le dioïde des séries formelles $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$, cette inéquation devient

$$\begin{aligned} \gamma^{-1} x_i &\succeq \delta \Delta_i^{min} x_i \\ \Leftrightarrow \gamma \delta \Delta_i^{min} x_i &\preceq x_i. \end{aligned}$$

Il est possible d'établir une forme globale de la contrainte (avec éventuellement plusieurs variables d'état assujetties) en utilisant la représentation matricielle suivante

$$x \succeq \gamma \Delta^{min} x, \quad (3.3)$$

dans laquelle Δ_{min} est une matrice diagonale définie par

$$\Delta_i^{min} = \begin{cases} \delta \Delta_i^{min} & \text{si l'écart } \Delta_i^{min} \text{ est souhaité entre deux occurrences de } x_i, \\ \varepsilon & \text{sinon.} \end{cases}$$

De cette manière, on obtient une contrainte de la forme (3.2).

Cependant, cette contrainte peut aussi se décrire par une inéquation du type (3.1) puisque le terme $\gamma \Delta^{min}$ est inversible (Δ^{min} est diagonale et γ est un scalaire, cf. remarque 13, p.22), à savoir :

$$x \preceq \gamma^{-1} (\Delta^{min})^{-1} x.$$

- Le pendant de la précédente contrainte est de chercher à limiter le temps entre deux occurrences successives d'un même événement, ce qui est décrit dans le dioïde $\overline{\mathbb{Z}}_{max}$ par l'équation

$$x_i(k) \preceq \Delta_i^{max} x_i(k-1), \quad (3.4)$$

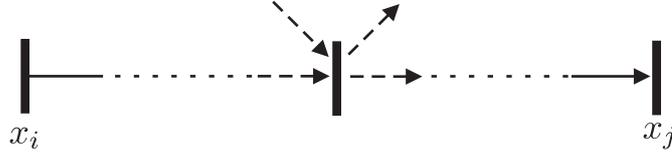


FIG. 3.1: Schéma d'un chemin de GET.

où $\Delta_i^{max} \in \mathbb{N}$ est le temps maximum entre deux occurrences successives de l'événement x_i . Cependant, on doit prendre en compte que les conditions initiales considérées sont canoniques, soit $x_i(k) = \varepsilon$ pour $k < k_{d_i}$. Si on applique l'inégalité précédente, on obtient, en fait, $x_i(k) \preceq \varepsilon, \forall k$. Aussi, on propose une meilleure traduction de cette contrainte :

$$\begin{cases} x_i(k) \preceq \Delta_i^{max} \otimes x_i(k-1) & \text{pour } k_{d_i} < k, \\ x_i(k) = x_i(k_{d_i}) & \text{pour } k = k_{d_i}. \end{cases}$$

$$\Leftrightarrow x_i(k) \preceq \Delta_i^{max} \otimes x_i(k-1) \oplus x_i(k_{d_i}) \quad \text{pour } k_{d_i} \leq k.$$

Dans le dioïde $\mathcal{M}_{in}^{ax}[\gamma, \delta]$, cette inégalité s'écrit pour tout le vecteur d'état

$$x \preceq \gamma \Delta^{max} x \oplus x_d. \quad (3.5)$$

dans laquelle Δ_{max} est une matrice diagonale définie par

$$\Delta_{ii}^{max} = \begin{cases} \delta \Delta_i^{max} & \text{si } x_i \text{ est soumis à la contrainte (3.4),} \\ \varepsilon & \text{sinon.} \end{cases}$$

et $x_{d_i} = \gamma^{k_{d_i}} \delta^{x_i(k_{d_i})}$.

- On peut chercher à borner la durée de certaines activités, ou tâches, au sein d'un SED. Un exemple classique pour illustrer cette contrainte est celui du four : si l'on considère un système de production dans lequel les pièces passent dans un four, la durée de séjour des pièces ne doit pas y excéder une certaine valeur sous peine de les endommager. Pour un système $(max, +)$ -linéaire modélisé par un GET, cela revient à souhaiter limiter le temps de présence des jetons dans certains chemins du graphe. La commande de ces systèmes avec une telle contrainte a été considérée dans [Lahaye et al., 2004] où les auteurs mettent en oeuvre un correcteur sous la forme d'un précompensateur dans un objectif de poursuite de modèle en juste-à-temps et dans [Amari et al., 2005] où une commande en boucle fermée est étudiée.

Examinons la mise en équation de cette contrainte pour le chemin d'un GET schématisé

sur la figure 3.1. On note α_{ji} le nombre de jetons initialement présents dans le chemin allant de la transition x_i à la transition x_j et τ_{ji} le temps maximum souhaité de présence pour les jetons dans ce même chemin. Cette contrainte s'exprime par

$$\begin{aligned} & \overbrace{x_j(k + \alpha_{ji})}^{\substack{\text{date de sortie du jeton introduit} \\ \text{dans le chemin par le tir n}^{\circ}k \text{ de } x_i}} - \overbrace{x_i(k)}^{\substack{\text{date du tir numéro } k \text{ de } x_i}} \preceq \tau_{ji}, \\ \Leftrightarrow & x_j(k + \alpha_{ji}) \preceq \tau_{ji} \otimes x_i(k) \quad (\text{dans } \overline{\mathbb{Z}}_{max}). \end{aligned}$$

Dans $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$, la dernière inéquation se traduit

$$x_j \preceq \gamma^{\alpha_{ji}} \delta^{\tau_{ji}} x_i. \quad (3.6)$$

On remarque que les temps de séjour des places du chemin n'interviennent pas dans la modélisation de la contrainte. Traduite sous sa forme matricielle, cette contrainte s'exprime par

$$x \preceq \Phi x, \quad (3.7)$$

où Φ est une matrice carrée de dimension $n \times n$ telle que

$$\begin{cases} \Phi_{ll} = e & \text{pour tout } l \neq j \\ \Phi_{ji} = \gamma^{\alpha_{ji}} \delta^{\tau_{ji}} \\ \Phi_{pq} = \varepsilon & \text{sinon.} \end{cases} \quad (3.8)$$

Si on souhaite appliquer plusieurs contraintes de ce type à différents chemins du GET, on définit une matrice Φ (décrite par (3.8)) pour chacune de ces contraintes, et on aboutit à la formulation du type (3.1) suivante :

$$x \preceq \Phi_1 x \wedge \Phi_2 x \wedge \dots \wedge \Phi_q x,$$

où q est le nombre de chemins sur lesquels on souhaite limiter le temps de présence des jetons.

Pour exprimer cette contrainte sous la forme (3.2), il suffit de remarquer que l'équation (3.6) peut s'écrire de façon équivalente

$$\gamma^{-\alpha_{ji}} \delta^{-\tau_{ji}} x_j \preceq x_i.$$

Sur un vecteur d'état de dimension n , la contrainte s'écrit sous la forme matricielle

$$x \succeq \Phi' x,$$

où Φ' est une matrice carrée de dimension $n \times n$ telle que

$$\Phi'_{ij} = \begin{cases} \gamma^{-\alpha_{ji}} \delta^{-\tau_{ji}} & \text{s'il existe une contrainte sur chemin reliant } x_i \text{ à } x_j, \\ \varepsilon & \text{sinon.} \end{cases}$$

- Il peut également être souhaitable de borner le nombre de jetons dans certains chemins d'un GET. Si ce graphe modélise un système de production, ce type de contrainte a pour but de limiter les en-cours dans certaines parties du réseau. On considère, à nouveau, le chemin d'un GET schématisé sur la figure 3.1. Le nombre de jetons contenus dans ce chemin est donné par $x_i(t) - x_j(t) + \alpha_{ji}$, où $x_i(t)$ et $x_j(t)$ sont les compteurs associés aux transitions x_i et x_j . Borner le nombre de jetons dans le chemin revient alors à imposer la relation suivante

$$\kappa_{ji} \succeq x_i(t) - x_j(t) + \alpha_{ji}, \quad \forall t$$

où la variable κ_{ji} désigne le nombre maximum de jetons désiré dans le chemin. Dans $\overline{\mathbb{Z}}_{min}$, cette relation devient $\kappa_{ji} x_j(t) \preceq \alpha_{ji} x_i(t)$ (on rappelle que la relation d'ordre dans $\overline{\mathbb{Z}}_{min}$ est l'inverse de la relation d'ordre naturel). Finalement, dans le dioïde des séries formelles, la contrainte devient

$$\gamma^{\kappa_{ji} - \alpha_{ji}} x_j \preceq x_i. \quad (3.9)$$

La forme matricielle obtenue est alors telle que $x \succeq \Gamma x$ (qui est conforme à l'inéquation (3.2)) avec Γ une matrice $n \times n$ définie de la façon suivante

$$\Gamma_{ij} = \begin{cases} \gamma^{\kappa_{ji} - \alpha_{ji}} & \text{s'il existe une contrainte sur chemin reliant } x_i \text{ à } x_j, \\ \varepsilon & \text{sinon.} \end{cases} \quad (3.10)$$

Si on souhaite appliquer plusieurs contraintes de ce type à différents chemins du GET, on définit une matrice Γ (décrite par (3.10)) pour chacune de ces contraintes, et on aboutit à la formulation du type (3.2) suivante :

$$x \succeq \Gamma_1 x \oplus \Gamma_2 x \oplus \dots \oplus \Gamma_q x,$$

où q est le nombre de chemins sur lesquels on souhaite limiter le temps de présence des jetons.

De même que pour la contrainte précédente, l'expression de la contrainte peut se faire sous la forme (3.1). Plus précisément, l'équation (3.9) peut s'écrire, de façon équivalente

$$x_j \preceq \gamma^{\alpha_{ji} - \kappa_{ji}} x_i.$$

Cela conduit à la forme matricielle $x \preceq \Gamma' x$ où Γ' est une matrice carrée de dimension $n \times n$ telle que

$$\begin{cases} \Gamma'_{ll} = e & \text{pour tout } l \neq j \\ \Gamma'_{ji} = \gamma^{\alpha_{ji} - \kappa_{ji}} \\ \Gamma'_{pq} = \varepsilon & \text{sinon.} \end{cases}$$

- Nous présentons maintenant une contrainte introduite dans [Houssin et al., 2005]. Celle-ci vise à imposer un nombre voulu d'occurrences d'un événement entre deux dates. Partant de la représentation en dateurs d'un système, cette contrainte s'exprime pour un événement x_i par

$$\exists k \in \mathbb{Z} \text{ tel que } x_i(k) \succeq t_i \text{ et } x_i(k+m) \preceq t_i + r,$$

où m est le nombre minimum d'occurrences de x_i souhaitées pendant l'intervalle de temps $[t_i, t_i + r]$. Pour exprimer cette contrainte sur la série $x_i \in \mathcal{M}_{in}^{ax}[[\gamma, \delta]]$, on utilise les applications val et Pr_a définies au §2.4.2. Nous détaillons d'abord comment procéder pour spécifier qu'au moins une occurrence est souhaitée sur l'intervalle $[t_i, t_i + r]$. Le numéro de la première occurrence dans la série x_i ayant lieu après $t_i + r$ (c'est-à-dire pour $t \geq t_i + r + 1$) doit être strictement supérieur au numéro de la première occurrence qui a lieu après $t_i - 1$ (c'est-à-dire pour $t \geq t_i$), ce qui dans $\overline{\mathbb{Z}}_{min}$, s'écrit :

$$\begin{aligned} val(Pr_{t_i+r+1}(x_i)) &< val(Pr_{t_i}(x_i)) \\ \iff val(Pr_{t_i+r+1}(x_i)) &\preceq 1 \otimes val(Pr_{t_i}(x_i)). \end{aligned}$$

Pour spécifier m occurrences de l'événements x_i , on utilise alors l'inéquation

$$val(Pr_{t_i+r+1}(x_i)) \preceq m \otimes val(Pr_{t_i}(x_i)). \quad (3.11)$$

Une illustration des termes mis en jeu dans cette équation est proposée sur la figure 3.2. Les applications val et Pr_a étant résiduables (voir les propositions 9, p.52 et 10, p.54), en utilisant le théorème 11, l'inéquation précédente peut être reformulée comme suit :

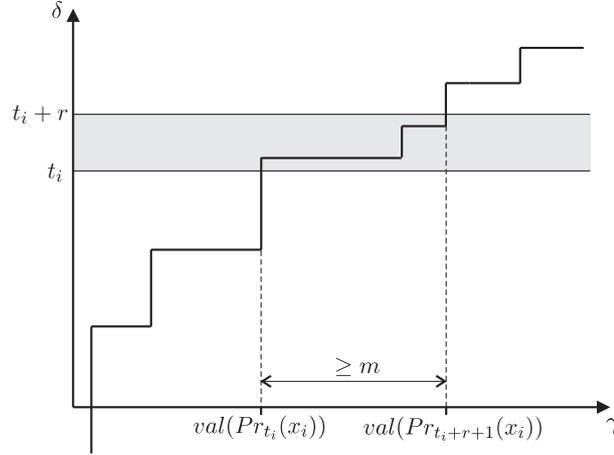


FIG. 3.2: Illustration des termes mis en jeu dans l'équation (3.11).

$$\begin{aligned}
 x_i &\preceq Pr_{t_i+r+1}^\#(val^\#(m \otimes val(Pr_{t_i}(x_i)))) \\
 \iff x_i &\preceq \gamma^{m \otimes val(Pr_{t_i}(x_i))} \delta^* \oplus (\gamma^{-1})^* \delta^{t_i+r}.
 \end{aligned}$$

La contrainte obtenue se présente donc sous la forme $x \preceq f(x)$ dans $\mathcal{M}_{in}^{ax}[\gamma, \delta]$, où f est isotone puisque les application val et Pr_a , ainsi que leur résiduées, sont isotones et la composition d'applications isotones reste isotone (remarque 8).

- La poursuite d'une trajectoire de référence constitue, en elle même, un objectif de commande dans [Cohen et al., 1989]. Cependant, nous la considérons, par la suite, comme une contrainte qui se formule dans le dioïde $\mathcal{M}_{in}^{ax}[\gamma, \delta]$ par

$$x \preceq C \bowtie z, \quad (3.12)$$

où C représente la matrice de sortie du modèle d'état (2.10) et z représente la trajectoire de référence. En d'autres termes, on ramène le problème de poursuite de trajectoire en une contrainte de la forme (3.1).

3.1.2 Un nouveau critère

Quelle que soit la structure choisie (voir notamment les figures 2.5, 2.6, 2.7 et 2.8), l'action d'un correcteur adjoint à un système $(max, +)$ -linéaire se borne à retarder les occurrences des événements au sein de ce dernier. Autrement dit, à l'aide d'un correcteur, on peut seulement se contenter de "ralentir" la dynamique du système. Comme il apparaît au §2.5.1, les critères

employés jusque-là visent tous à retarder au maximum le système sans enfreindre l'objectif de commande choisi. Concrètement, lorsque l'on se place dans le dioïde des séries formelles $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$, on cherche à synthétiser un correcteur dont le transfert est le plus grand possible. Au §3.3, on propose la synthèse de correcteurs qui, au contraire, "freinent" le moins possible l'évolution du système tout en satisfaisant un objectif de commande spécifié sous la forme de contrainte (*cf.* §3.1.1). À l'inverse des contrôleurs présentés lors de l'état de l'art au §2.5.3, ceux recherchés ici doivent avoir un transfert le plus petit possible. Ce nouveau critère induit également l'utilisation d'outils de synthèse différents de ceux employés jusque là.

3.2 Commande en juste-à-temps de systèmes $(max, +)$ -linéaires sous contraintes

Cette partie est consacrée à la détermination d'une commande en boucle ouverte pour les SED représentés comme des systèmes linéaires dans les dioïdes proposée dans [Houssin et al., 2005] et [Houssin et al., 2007]. Cette commande est optimale vis-à-vis du critère de juste-à-temps. L'originalité de cette commande par rapport à celles recensées au §2.5 se situe au niveau de l'objectif de commande. Outre la spécification d'une trajectoire de référence à poursuivre, on considère ici toute contrainte sur l'état du type $x \preceq f(x)$ (spécialement celles explicitées au §3.1.1). La commande proposée peut donc être considérée comme une généralisation du problème de commande décrit dans [Cohen et al., 1989]. Les outils de synthèse mêlent la théorie de la résiduation et la théorie des points fixes d'applications isotones.

On s'inspire ici du formalisme adopté dans [Borne et al., 1990] pour la commande optimale des systèmes linéaires conventionnels. Le principe de cette commande se résume en trois points :

- Le processus vérifie des conditions initiales et finales données.
- Les variables d'état sont soumises à certaines contraintes.
- La commande est dite "optimale" dans le sens où elle optimise un critère.

3.2.1 Conditions terminales

Les conditions terminales caractérisent à la fois l'état initial, c'est-à-dire l'état à l'instant où l'on commence à agir sur le processus (*via* les entrées), et l'état final, c'est-à-dire l'état à l'issue de l'application d'une commande en entrée.

Comme on a pu le détailler au §2.2.2, l'état initial d'un SED modélisé dans $\overline{\mathbb{Z}}_{max}$ est donné par les dates d'occurrence au plus tôt des événements initiaux. On considère ici des conditions initiales canoniques. Pour chaque variable d'état x_i , on a défini le nombre caractéristique (définition 21, p.40), c'est-à-dire l'indice k_{d_i} de la première occurrence engendrée par les entrées du

système.

Dans le cadre de notre commande, l'état final, que l'on se donne sous la forme d'un vecteur k_{f_y} , correspond aux indices des dernières occurrences des événements de sortie y que l'on désire commander. De cet objectif, on peut déduire quelles sont les dernières occurrences des variables d'état qu'il est nécessaire de piloter. On note $k_{f_{y_j}}$ (resp. k_{f_i}) la dernière occurrence de l'événement y_j (resp. x_i) que l'on désire commander. Le calcul de l'indice k_{f_i} fait intervenir le décalage événementiel $k_{f_{j_i}}$ entre l'état x_i et chaque sortie y_j du système (voir définition 22). La dernière occurrence de x_i à commander pour générer la dernière occurrence de la sortie y_j est donnée par $k'_{f_{j_i}} = k_{f_{y_j}} - k_{f_{j_i}}$. On en déduit ensuite que la dernière occurrence de x_i à commander est celle qui permet d'assurer les derniers événements de toutes les sorties, il en résulte $k_{f_i} = \max(k'_{f_{1_i}}, k'_{f_{2_i}}, \dots, k'_{f_{p_i}})$ pour un système comportant p sorties.

3.2.2 Contraintes

Le système que l'on désire commander peut être soumis à des contraintes. Elles spécifient l'objectif de commande défini dans la partie 3.1.1. On considère ici des contraintes qui peuvent s'exprimer sous la forme d'une inéquation du type $x \preceq f(x)$. Ces contraintes s'appliquent sur un intervalle d'occurrences pour chacune des variables d'état concernées. En effet, pour une variable d'état x_i elles ne peuvent être appliquées avant l'occurrence k_{d_i} , car avant celle-ci les occurrences de x_i ne sont pas contrôlables *via* les entrées du système. La dernière occurrence de x_i qui nous intéresse est k_{f_i} , au delà, il est inutile de contraindre x_i puisque cela concernerait des occurrences non désirées pour les événements de sortie. Les contraintes ne s'appliquent donc que pour les indices d'occurrences appartenant à l'intervalle $[k_{d_i}, k_{f_i}]$ pour chaque variable d'état x_i . La figure 3.3 montre l'intervalle d'occurrences sur lequel une variable d'état x_i peut être contrainte, les zones grisées correspondent aux occurrences que l'on ne peut pas contrôler, ou que l'on ne souhaite pas contrôler. Afin de les coder dans le dioïde $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$, dans lequel on manipule l'ensemble de la trajectoire d'un dateur $\{x_i(k)\}_{k \in \mathbb{Z}}$ sous la forme de la série x_i , on a recours à des vecteurs ω et ν définis comme ci-dessous :

$$\begin{aligned}
 x &\preceq (g_1(x) \wedge \omega) \oplus \nu \\
 x &\preceq (g_2(x) \wedge \omega) \oplus \nu \\
 \vdots &\quad \quad \quad \vdots \\
 x &\preceq (g_q(x) \wedge \omega) \oplus \nu,
 \end{aligned} \tag{3.13}$$

avec ω (respectivement ν) un vecteur tel que $\omega_i = \gamma^{k_{d_i}} \delta^*$ (respectivement $\nu_i = \gamma^{k_{f_i} + 1} \delta^*$) et chaque g_l , $l = 1, 2, \dots, q$, est une application de $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]^n$ dans $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]^n$ (n est la

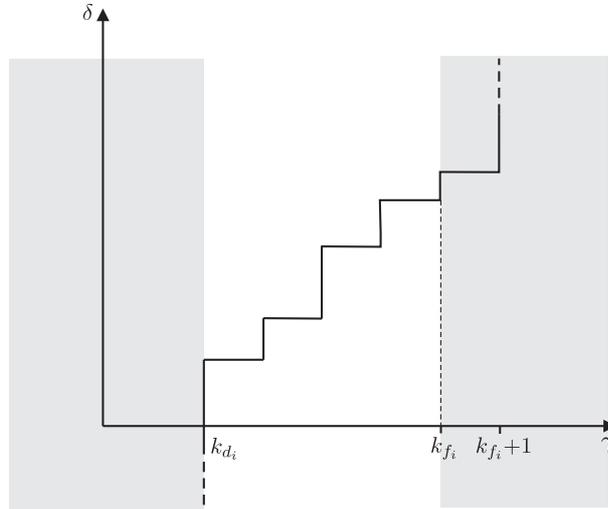


FIG. 3.3: Exemple de série x_i contrainte dans l'intervalle $[k_{d_i}, k_{f_i}]$.

dimension du vecteur d'état) modélise une contrainte. Les vecteurs ω et ν permettent de relaxer les contraintes pour les indices d'occurrences des événements x_i , $i = 1, 2, \dots, n$, en dehors de l'intervalle $[k_{d_i}, k_{f_i}]$. Le nombre d'occurrences commandées pour chaque variable d'état x_i est donné par $k_{f_i} - k_{d_i} + 1$.

Les contraintes agissent le plus souvent sur le vecteur d'état, mais dans le cas où les contraintes portent sur le vecteur d'entrée, ou celui de sortie, une solution simple est d'étendre le vecteur d'état. Cette manipulation permet de toujours modéliser les contraintes sous la forme (3.13).

3.2.3 Critère

Le critère utilisé dans la synthèse de cette commande est le critère du juste-à-temps. On va chercher à retarder au maximum les occurrences des événements d'entrée (c'est-à-dire, chercher le plus grand vecteur de commande u) tout en respectant les performances imposées par la spécification de commande (ici les conditions terminales et les contraintes).

3.2.4 Synthèse

On montre que la synthèse de la commande proposée se ramène à la recherche d'un plus grand point fixe d'une application isotone.

Afin de pouvoir être mise en oeuvre, la commande u recherchée doit être causale (voir la remarque 23, p.53), c'est-à-dire qu'elle doit vérifier l'égalité $u = Pr_+(u)$. On peut aisément vérifier que quel que soit u , on a $Pr_+(u) \preceq u$, ce qui nous conduit à prendre en compte l'inégalité suivante dans le calcul de la commande :

$$u \preceq Pr_+(u). \quad (3.14)$$

De plus, l'état généré par la commande u doit satisfaire l'ensemble des contraintes (3.13). En considérant le fonctionnement au plus tôt du système, on a $x = A^*Bu$ et en utilisant la notation

$$g'_i(u) = A^*B \backslash ((g_i(A^*Bu) \wedge \omega) \oplus \nu),$$

la recherche de la plus grande commande u revient à déterminer la plus grande solution du système d'inéquations :

$$\left\{ \begin{array}{l} u \preceq g'_1(u) \\ u \preceq g'_2(u) \\ \vdots \\ u \preceq g'_q(u) \\ u \preceq Pr_+(u), \end{array} \right.$$

ce qui est équivalent à trouver le plus grand u respectant l'inéquation

$$u \preceq g'_1(u) \wedge g'_2(u) \wedge \dots \wedge g'_q(u) \wedge Pr_+(u) \triangleq f(u). \quad (3.15)$$

Proposition 12 *Si la procédure itérative suivante converge en un nombre fini, noté k_e , d'itérations*

$$\begin{aligned} u_0 &= \top \\ u_{k+1} &= f(u_k), \end{aligned}$$

alors u_{k_e} est la commande qui respecte les conditions terminales ainsi que les contraintes traduites par f (équation (3.15)) pour laquelle le critère est optimal.

Preuve : L'ensemble des commandes qui respectent les contraintes coïncide avec l'ensemble \mathcal{P}_f

des points post-fixes de f . La suite définie dans la proposition 1, si elle converge, nous fournit le plus grand élément $Sup \mathcal{P}_f$ de cet ensemble.

□

Proposition 13 *Si parmi les contraintes, une trajectoire de référence z à poursuivre est considérée (avec $i \in \{1, \dots, p\}$, $z_i(k) \prec \top$ pour $k \leq k_{f_{y_i}}$), alors la suite définie dans la proposition 12 converge.*

Preuve : Nous montrons que l'ensemble des commandes qui respectent l'équation (3.15) quand une trajectoire de référence z est considérée est un ensemble fini. Puisque la suite est décroissante (remarque 14, p.25), l'algorithme converge.

Comme on l'a vu au §3.1.1 et plus particulièrement dans l'équation (3.12), une contrainte de trajectoire de référence peut être mise en équation par l'inégalité $x \preceq C \backslash z$. En tenant compte des conditions terminales du système, l'inéquation devient

$$x \preceq (C \backslash z \wedge \omega) \oplus \nu.$$

Par ailleurs, l'hypothèse du fonctionnement au plus tôt conduit à la contrainte suivante sur le vecteur de commande

$$u \preceq A^* B \backslash ((C \backslash z \wedge \omega) \oplus \nu),$$

ce qui permet de borner l'évolution de la suite $f(u_k)$ par

$$f(u_k) \preceq A^* B \backslash ((C \backslash z \wedge \omega) \oplus \nu) \quad \text{pour tout } k \geq 0. \quad (3.16)$$

On observe d'autre part, que l'on a $g'_l(u_k) = A^* B \backslash ((g_l(A^* B u_k) \wedge \omega) \oplus \nu) \succeq A^* B \backslash ((\varepsilon \wedge \omega) \oplus \nu) = A^* B \backslash \nu$ pour $l \in \{1, \dots, q\}$ et $Pr_+(u_k) \succeq Pr_+(A^* B \backslash \nu) = A^* B \backslash \nu$. On en déduit alors :

$$f(u_k) \succeq A^* B \backslash \nu \quad \text{pour tout } k \geq 0. \quad (3.17)$$

On s'intéresse maintenant à la structure des vecteurs qui encadrent $f(u_k)$ dans (3.16) et (3.17).

L'élément i du vecteur $A^* B \backslash ((C \backslash z \wedge \omega) \oplus \nu)$ est de la forme suivante : $(A^* B \backslash ((C \backslash z \wedge \omega) \wedge \nu))_i = \gamma^{n_d} \delta^{t_d} \oplus \dots \oplus \gamma^{n_f} \delta^{t_f} \oplus \gamma^{n_f+1} \delta^*$. Le premier exposant n_d en γ correspond au numéro

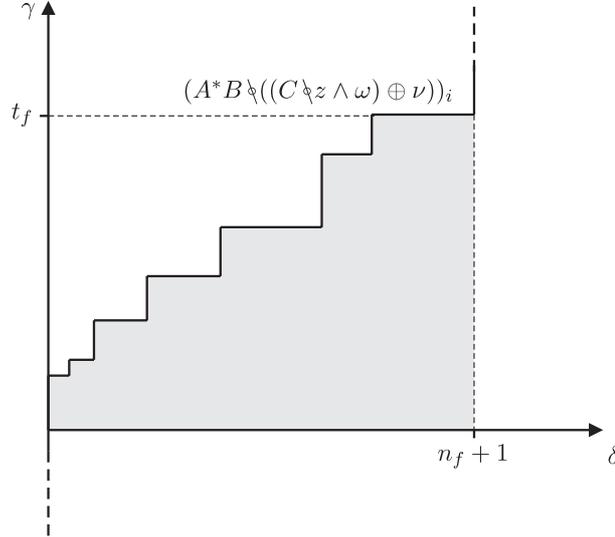


FIG. 3.4: Exemple de série x_i soumise à la contrainte.

de la première occurrence de x_i contrôlable par les entrées (donné par ω_i), soit k_{d_i} , moins le décalage événementiel entre la variable d'état x_i et les entrées du système, c'est-à-dire k_{d_i} . On a donc $n_d = k_{d_i} - k_{d_i} = 0$. De la même façon, n_f correspond à la dernière occurrence de x_i à commander, soit k_{f_i} , moins le décalage événementiel entre la variable d'état x_i et les entrées du système, c'est-à-dire k_{d_i} . Il en découle $n_f = k_{f_i} - k_{d_i}$. Les exposants $\{t_d, \dots, t_f\}$ appartiennent à $\mathbb{N} \cup \{\varepsilon\}$, puisque $z_j(k) \in \mathbb{N} \cup \{\varepsilon\}$ pour $k \leq k_{f_{y_j}}$.

La structure de ν implique que les éléments de $A^*B \setminus \nu$ ont la forme $(A^*B \setminus \nu)_i = \gamma^{n_f+1} \delta^*$.

On obtient donc pour chaque élément $(f(u_k))_i$:

$$\gamma^{n_f+1} \delta^* \preceq (f(u_k))_i \preceq \gamma^{n_d} \delta^{t_d} \oplus \dots \oplus \gamma^{n_f} \delta^{t_f} \oplus \gamma^{n_f+1} \delta^*. \quad (3.18)$$

La figure 3.4 schématise un exemple de l'encadrement (3.18).

L'encadrement (3.18) permet d'affirmer que la série $(A^*B \setminus ((C \setminus z \wedge \omega) \oplus \nu))_i$ est telle que

- sa valuation est égale à 0 (ce qui paraît logique puisque la première occurrence de l'entrée est numérotée 0),
- les exposants en δ associées aux exposants en γ compris entre n_d et n_f sont finis (*i.e.* t_d, \dots, t_f sont finis),

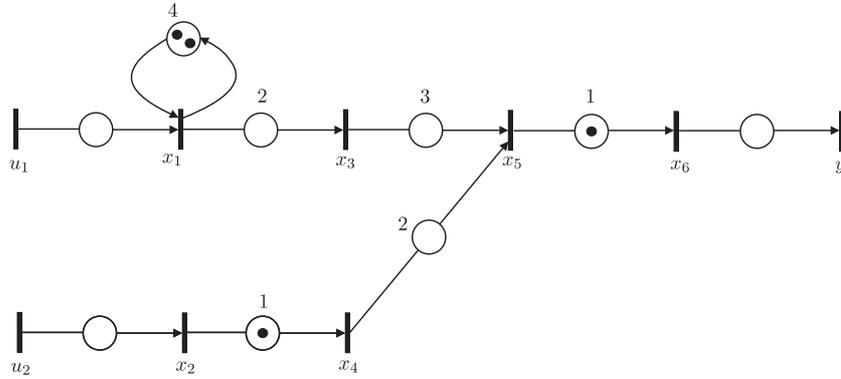


FIG. 3.5: Exemple de GET.

- pour les exposants en γ supérieurs à n_f , les exposants en δ sont infinis.

De plus, seuls les monômes causaux de u_k sont préservés dans $f(u_k)$ (puisque la contrainte (3.14) interdit toute forme non causale).

A partir de ces observations, on déduit qu'il existe un nombre fini de séries causales satisfaisant (3.18). En d'autres termes, chaque élément $(f(u_k))_i$ appartient à un ensemble fini de séries causales. La partie grisée de la figure 3.4, munie de la série ε , représente cet ensemble. Finalement, comme la suite décroît (voir remarque 14) dans un ensemble fini et pour des raisons d'intégrité des exposants, la convergence est assurée.

□

3.2.5 Exemple

Nous appliquons la commande en juste-à-temps sous contraintes proposée au paragraphe précédent au GET de la figure 3.5.

On veut effectuer 5 tirs de la transition de sortie y .

Les contraintes qui sont prises en compte sont les suivantes :

- contrainte de poursuite de trajectoire : $z = \gamma^0 \delta^{10}$.
- pas plus de 2 jetons dans la place située entre x_4 et x_5 .

La procédure itérative de la proposition 12 converge et fournit la commande

$$u_{opt} = \begin{pmatrix} \gamma^0 \delta^0 \oplus \gamma^2 \delta^4 \oplus \gamma^4 \delta^* \\ \gamma^0 \delta^4 \oplus \gamma^1 \delta^6 \oplus \gamma^3 \delta^* \end{pmatrix}.$$

Les détails du calcul sont disponibles en annexe A.1.

3.3 Commande de systèmes $(max, +)$ -linéaires contraints en minimisant les retards

Dans cette section, nous proposons une commande pour les systèmes $(max, +)$ -linéaires. L'objectif de commande considéré est celui présenté dans la section 3.1.1. Partant de la représentation du système dans le dioïde des séries formelles $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$, on souhaite plus précisément que l'état satisfasse un ensemble de contraintes. Celles-ci doivent s'exprimer sous la forme d'une inéquation du type $x \succeq f(x)$, où f est une application isotone. Plusieurs contraintes de cette forme ont été explicitées dans la section 3.1.1. On adopte le critère introduit dans la section 3.1.2, c'est-à-dire que l'on cherche à synthétiser un correcteur qui ralentit le moins possible le système. En d'autres termes, on vise à établir un correcteur de transfert minimal.

On expose ici une correction en boucle fermée, tout d'abord par l'intermédiaire d'un retour de l'état sur l'état, introduit dans [Houssin et al., 2006a]. Puis on propose deux autres structures de commande qui correspondent à un retour de l'état sur un ensemble restreint de composantes de l'état et un retour de l'état sur les entrées du système.

3.3.1 Formalisation du problème

Nous prendrons comme hypothèse, non restrictive, que le système $(max, +)$ -linéaire peut être décrit par les équations d'état (2.10) et que la matrice B présente une structure carrée et diagonale telle que $B \preceq Id$ (un simple agrandissement du vecteur d'état permet de se ramener sous cette forme). Plus précisément, B est définie telle que

$$B_{ii} = \begin{cases} e & \text{si } x_i \text{ est reliée à l'entrée } u_i \text{ du système,} \\ \varepsilon & \text{sinon.} \end{cases}$$

L'objectif de commande est ici de respecter certaines contraintes que l'on impose au système telles que celles définies dans le paragraphe 3.1.1. Pour la synthèse de la commande, ces dernières doivent être décrites par des inéquations de la forme $x \succeq \phi_i x$. L'ensemble de ces contraintes

peut se résumer sous la forme d'une simple inéquation puisque les inégalités

$$\left\{ \begin{array}{l} x \succcurlyeq \phi_1 x \\ x \succcurlyeq \phi_2 x \\ \vdots \\ x \succcurlyeq \phi_q x, \end{array} \right.$$

où chaque matrice ϕ_i , $i \in \{1, \dots, q\}$, modélise une contrainte, sont équivalentes à

$$x \succcurlyeq \phi x, \quad (3.19)$$

avec $\phi = \phi_1 \oplus \phi_2 \oplus \dots \oplus \phi_q$.

L'adjonction d'un correcteur à un système décrit par l'équation (2.10) ne peut que le ralentir, et l'état du système corrigé, noté x_c , satisfait donc nécessairement

$$x_c \succcurlyeq Ax_c \oplus Bu.$$

L'objectif de commande impose de respecter les contraintes spécifiées par l'équation (3.19). L'état du système corrigé doit alors satisfaire

$$x_c \succcurlyeq (A \oplus \phi)x_c \oplus Bu. \quad (3.20)$$

L'objectif étant de ralentir le moins possible le transfert du système, on cherche à atteindre le plus petit état corrigé x_c , à savoir, la plus petite solution de (3.20), d'où

$$x_c \succcurlyeq (A \oplus \phi)^* Bu.$$

Une reformulation de ce problème consiste à chercher un plus petit transfert état-entrée corrigé H_c tel que

$$H_c u \succcurlyeq (A \oplus \phi)^* Bu, \quad \forall u.$$

Puisque le système corrigé doit obéir à l'inéquation précédente quelle que soit l'entrée u , un problème équivalent est de rechercher le plus petit transfert corrigé H_c qui satisfait

$$H_c \succeq (A \oplus \phi)^* B. \quad (3.21)$$

3.3.2 Correction par retour d'état sur l'état

On expose maintenant la synthèse d'une correction par retour d'état sur l'état. La relation état-entrée d'un système corrigé de la sorte est donnée par $x = (A \oplus F)^* B u$, où F désigne le retour d'état (voir le paragraphe 2.5.3 et la figure 2.7(a)). Il est pris pour hypothèse que les événements internes du systèmes sont mesurables et que le correcteur a la possibilité de retarder ces événements. Le problème se résume alors en la recherche d'un plus petit feedback F tel que

$$(A \oplus F)^* B \succeq (A \oplus \phi)^* B. \quad (3.22)$$

Certaines contraintes dans l'objectif de commande peuvent se révéler impossibles à réaliser ou contradictoires entre elles, dès lors, aucun feedback causal ne peut corriger le système. La proposition suivante fournit une condition nécessaire et suffisante pour l'existence d'un feedback causal à partir de la matrice ϕ représentant les contraintes.

Proposition 14 *Un feedback causal F existe, si et seulement si, $(A \oplus \phi)^* B$ admet une sur-approximation causale notée G .*

Preuve : En accord avec la proposition 11, p.54, il existe une plus petite matrice G causale telle que $G \succeq (A \oplus \phi)^* B$ si et seulement si aucun élément de $(A \oplus \phi)^* B$ ne possède d'exposant négatif en γ . Il est alors toujours possible de trouver une matrice F causale telle que $(A \oplus F)^* \succeq G$. De plus, par isotonie du produit, on a $GB \succeq (A \oplus \phi)^* B^2 = (A \oplus \phi)^* B$ (la dernière égalité découle du fait que $B^n = B$ pour $n \geq 1$ car $B \preceq Id$). Le produit GB est causal puisque G et B sont causaux et G est la matrice causale la plus petite telle que $G \succeq (A \oplus \phi)^* B$, on peut donc affirmer $GB \succeq G$. Or, on sait que $B \preceq Id$, d'où l'inégalité $GB \preceq G$. En conséquence, il existe un feedback F causal tel que $(A \oplus F)^* B \succeq GB = G$.

□

La proposition 14 nous permet de ramener le problème à la recherche du plus petit F tel que

$$(A \oplus F)^* B \succeq G, \quad (3.23)$$

puisque l'ensemble des solutions causales de (3.22) coïncide avec l'ensemble des solutions causales de (3.23).

Proposition 15 *Les solutions de (3.22) sont les éléments de \mathcal{Q}_g (cf. notations 3, p.23) avec $g : F \mapsto G \oplus (A \oplus F)^*$.*

Preuve : On a les équivalences suivantes

$$\begin{aligned}
 & G \preceq (A \oplus F)^* B \\
 \Leftrightarrow & G \preceq (A \oplus F)^* \quad \text{puisque } GB = G \text{ (voir preuve de la prop. 14) et } B \preceq Id \\
 \Leftrightarrow & G \preceq (A \oplus F)^* \oplus F \quad \text{on a } (A \oplus F)^* \succeq F \\
 \Leftrightarrow & G \oplus (A \oplus F)^* \preceq F \quad \text{car } T_{(A \oplus F)^*} \text{ est dualement résiduable (cf. §1.5).}
 \end{aligned}$$

□

On peut remarquer que l'application g est antitone puisque si l'on considère deux éléments x_1 et x_2 de $\mathcal{M}_{in}^{ax}[[\gamma, \delta]]$ tels que $x_1 \succeq x_2$, on obtient

$$\begin{aligned}
 & A \oplus x_1 \quad \succeq \quad A \oplus x_2 \\
 \Rightarrow & (A \oplus x_1)^* \quad \succeq \quad (A \oplus x_2)^* \\
 \Rightarrow & G \oplus (A \oplus x_1)^* \quad \preceq \quad G \oplus (A \oplus x_2)^* \quad (\text{puisque } T_{(A \oplus F)^*} \text{ est antitone}).
 \end{aligned}$$

Les solutions de (3.23) sont donc les points préfixes de l'application g . Comme nous l'avons vu dans la section 1.6.2, l'ensemble \mathcal{Q}_g n'est pas un treillis (seulement un demi treillis supérieur). Cela laisse figurer qu'il n'existe pas nécessairement de "plus petit élément" mais seulement des éléments minimaux.

L'élément $v = Sup \mathcal{F}_{g^2}$ appartient à \mathcal{Q}_g (voir la proposition 8, p.29 et le corollaire 2, p.30). Il n'est pas nécessairement minimal, mais l'élément v est un feedback satisfaisant (3.23). Comme le mentionne le corollaire 3, si on a $Sup \mathcal{F}_{g^2} = Inf \mathcal{F}_{g^2}$, alors il n'existe qu'un seul point fixe. Autrement, on peut considérer v comme une approximation d'un feedback minimal satisfaisant (3.23).

Du reste, on peut avancer la proposition suivante qui caractérise la structure du feedback calculé et permet de relaxer l'hypothèse prise au départ qui consistait à considérer toutes les variables d'états mesurables.

Proposition 16 *Le feedback $v = \text{Sup } \mathcal{F}_{g^2}$ est tel que $v = vB$.*

Preuve : On montre en premier lieu que $g(F)B = g(F)$.

Puisque $B \preceq Id$, on a $(G \oplus (A \oplus F)^*)B \preceq G \oplus (A \oplus F)^*$, c'est-à-dire $g(F)B \preceq g(F)$. Il découle de la propriété (f.14) du tableau 1.1 que $GB \oplus (A \oplus F)^*B \preceq (G \oplus (A \oplus F)^*)B$. Or, la preuve de la proposition 14 révèle que $GB = G$, on a par conséquent

$$\begin{aligned} GB \oplus (A \oplus F)^*B &\preceq (G \oplus (A \oplus F)^*)B \\ \Leftrightarrow G \oplus (A \oplus F)^*B &\preceq (G \oplus (A \oplus F)^*)B \\ \Rightarrow G \oplus (A \oplus F)^* &\preceq G \oplus (A \oplus F)^*B \preceq (G \oplus (A \oplus F)^*)B = g(F)B. \end{aligned}$$

La dernière inéquation résulte du fait que $(A \oplus F)^* \succeq (A \oplus F)^*B$ d'où $G \oplus (A \oplus F)^* \preceq G \oplus (A \oplus F)^*B$. On a donc $g(F) \preceq g(F)B$ et en conséquence $g(F) = g(F)B$.

Par définition de v , on a $g^2(v) = v$, on obtient alors $vB = g(g(v))B = g(g(v)) = g^2(v) = v$.

□

La proposition 16 nous amène à reconsidérer l'équation d'évolution du système corrigé :

$$x = Ax \oplus Bu \oplus vBx.$$

La proposition 16 nous indique que le retour d'état calculé se fait à partir de certaines variables d'état (celles qui ne sont pas annulées dans le produit Bx). Les variables d'état mesurées pour le calcul de la commande sont, en fait, celles qui sont directement liées aux variables de commande. Plus formellement, une variable d'état x_i est employée pour le calcul de la correction si $B_{ii} = e$. Vis-à-vis du GET réalisant le feedback adjoint au système nominal, celui-ci ne comporte que des arcs qui partent des variables d'état qui se trouvent directement reliées aux entrées vers d'autres variables d'état.

3.3.3 Exemple

La figure 3.6 présente un GET avec une réalisation de son correcteur avec les contraintes suivantes :

- on ne doit pas avoir plus de 3 jetons dans le chemin allant de la transition x_3 à la transition x_6 ,

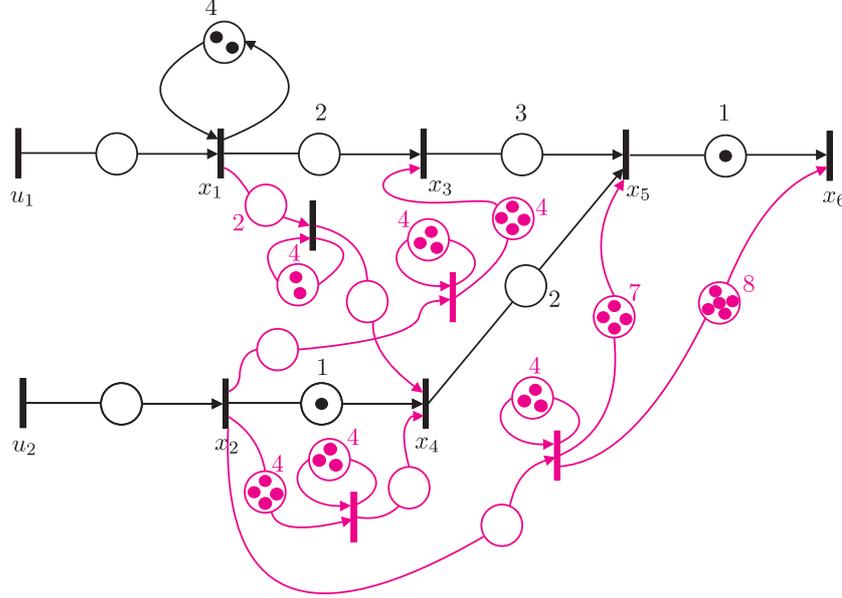


FIG. 3.6: Exemple de GET avec une réalisation de son correcteur.

- les jetons ne doivent pas séjourner plus de 3 unités de temps dans la place située entre x_4 et x_5 .

Dans cet exemple, on obtient le correcteur suivant

$$v = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \gamma^4 \delta^4 (\gamma^3 \delta^4)^* & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \delta^2 (\gamma^2 \delta^4)^* & \gamma^4 \delta^4 (\gamma^3 \delta^4)^* & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \gamma^4 \delta^7 (\gamma^3 \delta^4)^* & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \gamma^5 \delta^8 (\gamma^3 \delta^4)^* & \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{pmatrix}.$$

Les détails du calcul sont disponibles en annexe A.2.

Puisque $Sup \mathcal{F}_{g^2} \neq Inf \mathcal{F}_{g^2}$, on ne peut donc pas affirmer que le feedback représenté sur la figure 3.6 est minimal. En effet, on peut vérifier qu'il existe un feedback F' , plus petit que v , tel que $F'_{ij} = v_{ij}$ pour $(i, j) \neq (5, 2)$ et $(i, j) \neq (6, 2)$ qui satisfait les contraintes puisque on a bien $G \preceq (A \oplus F')^* B$. On remarque cependant que le transfert du système corrigé avec F' est identique à celui du système corrigé avec v , c'est-à-dire $(A \oplus v)^* B = (A \oplus F')^* B$.

3.3.4 Correction par retour d'état sur l'état : restriction à un ensemble de variables

Dans la section précédente, nous avons établi un feedback sur les variables d'état en considérant qu'elles pouvaient toutes être contrôlées, *i.e.* que les occurrences des événements correspondant pouvaient être retardées par l'adjonction d'un correcteur.

Dans ce paragraphe, nous étudions la synthèse d'un correcteur du type retour d'état sur l'état tel que la correction ralentisse seulement certaines variables d'état spécifiées. Ce type de correction présente un intérêt lorsque des variables internes au système ne sont pas directement commandables.

Afin de spécifier les variables d'état que l'on souhaite commander, on définit une matrice diagonale D telle que

$$D_{ii} = \begin{cases} e & \text{si l'on souhaite que le feedback s'applique à la variable d'état } x_i, \\ \varepsilon & \text{sinon.} \end{cases}$$

Nous complétons la proposition 15 par la proposition suivante qui permet, sous certaine hypothèse sur G (la sur-approximation causale de $(A \oplus \phi)^* B$), de calculer un feedback n'agissant que sur l'ensemble restreint par D des variables d'état.

Proposition 17 *Si $DG = G$, alors $v = \text{Sup } \mathcal{F}_{g^2}$ est tel que $Dv = v$.*

Preuve : La preuve est assez proche de celle de la proposition 16. On montre, tout d'abord que $Dg(F) = g(F)$ où $g(F)$ est définie comme dans la proposition 15.

Puisque $D \preceq Id$, on a logiquement $D(G \oplus (A \oplus F)^*) \preceq G \oplus (A \oplus F)^*$, c'est-à-dire $Dg(F) \preceq g(F)$. La propriété (f.14) du tableau 1.1 nous permet de poser $DG \oplus D(A \oplus F)^* \preceq D(G \oplus (A \oplus F)^*)$. Or, par hypothèse, on a $DG = G$ ce qui nous conduit à

$$G \oplus D(A \oplus F)^* \preceq D(G \oplus (A \oplus F)^*).$$

Puisque $D \preceq Id$ et g est antitone, on a également $G \oplus (A \oplus F)^* \preceq G \oplus D(A \oplus F)^*$. Il en découle

$$G \oplus (A \oplus F)^* \preceq D(G \oplus (A \oplus F)^*),$$

soit $g(F) \preceq Dg(F)$. Par conséquent, on a montré que $g(F) = Dg(F)$.

Par définition de v on a $g^2(v) = v$, on obtient alors $Dv = Dg(g(v)) = g(g(v)) = g^2(v) = v$.

□

Le retour obtenu satisfait donc $Dv = v$. L'équation d'évolution du système peut alors s'écrire :

$$x = Ax \oplus Bu \oplus Dvx.$$

D'après l'égalité précédente, seules les variables d'état x_i , telles que $D_{ii} = e$, sont atteinte par la correction.

Remarque 25 *Le retour d'état a donc la forme $v = DvB$ (en tenant compte de la proposition 16). Concernant le GET réalisant ce correcteur, la forme particulière de v a pour conséquences que :*

- seules les transitions x_i , telles que $B_{ii} = e$, seront utilisées pour le calcul de la correction,
- seules les transitions x_j , telles que $D_{jj} = e$, seront ralenties par la correction.

3.3.5 Correction par retour d'état sur l'entrée

On cherche maintenant à utiliser l'état du système pour l'élaboration d'une commande sur les entrées du système. Tout comme la correction par retour d'état restreint à un ensemble de variables d'état, le retour d'état sur l'entrée ne peut s'effectuer que sous certaines conditions.

Formellement, nous utilisons la structure de commande présentée sur la figure 2.6(b) pour laquelle le correcteur F est appliqué en feedback de l'état sur l'entrée. Le transfert du système corrigé est alors donné par

$$H_c = (A \oplus BF)^*B,$$

et l'on cherche donc à synthétiser un feedback F minimal tel que

$$H_c \succeq (A \oplus \phi)^*B. \quad (3.24)$$

Proposition 18 *Si $(A \oplus \phi)^*B$ admet une sur-approximation causale, notée G , telle que $BG = G$ alors il existe un feedback causal F tel que $(A \oplus BF)^*B \succeq (A \oplus \phi)^*B$.*

Preuve : Supposons que $(A \oplus \phi)^*B$ admette une sur-approximation causale, notée G , telle que $BG = G$. On peut toujours trouver une matrice causale F telle que $BF \succeq BG$. Cette matrice

est un feedback qui satisfait le problème de synthèse (3.24) puisque

$$(A \oplus BF)^*B \succeq (BF)^*B \succeq BFB \succeq BGB = GB = G.$$

Les deux dernières égalités sont obtenues par hypothèse sur G ($BG = G$) et grâce à la preuve de la proposition 14 ($GB = G$).

□

La proposition 18 fournit une condition suffisante pour l'existence d'un feedback F satisfaisant (3.24). En supposant celle-ci satisfaite, on explicite dans la proposition suivante l'ensemble des feedbacks solutions de (3.24).

Proposition 19 *On suppose que $(A \oplus \phi)^*B$ admet une sur-approximation causale, notée G , telle que $BG = G$. Les solutions de (3.24) sont alors les points préfixes de l'application g' définie par $g' : F \mapsto B(G \ominus (A \oplus BF)^*)$.*

Preuve : Les feedbacks causaux que l'on recherche sont tels que

$$\begin{aligned} (A \oplus BF)^*B &\succeq G \\ \Leftrightarrow (A \oplus BF)^* &\succeq G && \text{(puisque } GB = G \text{ et } B \preceq Id) \\ \Leftrightarrow BF \oplus (A \oplus BF)^* &\succeq G && \text{(puisque } (A \oplus BF)^* \succeq BF) \\ \Leftrightarrow BF &\succeq G \ominus (A \oplus BF)^* && (T_{(A \oplus BF)^*} \text{ est duallement résiduable)} \\ \Leftrightarrow F &\succeq B(G \ominus (A \oplus BF)^*). \end{aligned}$$

Pour la dernière équivalence :

(\Rightarrow)

$$\begin{aligned} BF \succeq G \ominus (A \oplus BF)^* &\Rightarrow B^2F \succeq B(G \ominus (A \oplus BF)^*) \\ &\Rightarrow F \succeq B(G \ominus (A \oplus BF)^*) \\ &\text{(puisque } F \succeq BF = B^2F) \end{aligned}$$

(\Leftarrow)

$$\begin{aligned}
 F \succeq B(G \oplus (A \oplus BF)^*) &\Rightarrow BF \succeq B^2(G \oplus (A \oplus BF)^*) \\
 &\Leftrightarrow BF \succeq B(G \oplus (A \oplus BF)^*) \\
 &\Rightarrow BF \succeq BG \oplus B(A \oplus BF)^* \\
 &\quad (a(x \oplus b) \succeq ax \oplus ab, \text{ formule (f14) du tableau 1.2}) \\
 &\Rightarrow BF \succeq BG \oplus (A \oplus BF)^* \\
 &\quad (x \mapsto a \oplus x \text{ est antitone}).
 \end{aligned}$$

□

Le calcul de $v' = \text{Sup } \mathcal{F}_{g'^2}$ fournit un feedback satisfaisant (3.24) puisque v' est un point préfixe de g' (voir la proposition 8 et le corollaire 2). La proposition 16, introduite dans le cadre de la commande par retour d'état sur l'état, reste d'actualité dans le cas de la correction par retour d'état sur les entrées, on a donc $g'(F)B = g'(F)$ et $v'B = v'$. Le feedback v' constitue donc un retour d'état tel que :

- les variables d'état mesurées pour le calcul de la commande sont celles qui se trouvent directement reliées aux entrées du système, *i.e.* celles qui sont conservées par la matrice B ,
- les variables ralenties par la correction sont les variables d'entrée du système.

Chapitre 4

Applications aux réseaux de transport

Par certains aspects, les réseaux de transport peuvent souvent être perçus comme des SED. En effet, leur évolution est conditionnée par l'apparition d'événements tels que le départ ou l'arrivée de véhicules de transport. De plus, les réseaux de transport sont soumis à des phénomènes de synchronisation, en l'occurrence, les correspondances. Pour ces raisons, une description linéaire dans l'algèbre $(max, +)$ peut être intéressante à réaliser pour décrire l'évolution de ces systèmes.

Dans ce chapitre, on s'attache à présenter une modélisation des réseaux de transport ainsi que des outils pour l'évaluation de performances et la commande de tels systèmes.

On commence par rappeler quelques généralités concernant les systèmes de transport en décrivant les éléments qui les composent.

Les principes de fonctionnement des réseaux de transport sont ensuite développés. Il s'agit d'une synthèse de différents rapports décrivant les pratiques d'exploitation de tels réseaux. Nous verrons que celles-ci se répartissent en deux phases successives : une phase d'optimisation et de planification qui permet d'aboutir à un *tableau de marche*, puis une phase dite de *régulation*.

Se basant sur cette description, on propose ensuite une modélisation dans l'algèbre des dioïdes du tableau de marche. On aboutit précisément à une description linéaire standard dans cette structure algébrique.

A partir de ce modèle, on propose dans la partie suivante une analyse de la stabilité des réseaux de transport. Plus explicitement, à partir de résultats non originaux sur la théorie spectrale (rappelée au §1.4), on met à jour une condition pour chaque ligne qui garantit que le réseau puisse absorber les éventuels retards survenant dans le système.

On s'intéresse ensuite plus particulièrement à une étape de construction du tableau de marche : la synthèse des tables d'horaires. En posant le problème sous la forme d'un ensemble

de contraintes, on montre que ce problème peut se ramener à un problème de commande en juste-à-temps de système $(max, +)$ -linéaires. La commande exposée au §3.2 permet de résoudre ce problème. Nous illustrons l'approche par le calcul des tables d'horaires d'une partie du réseau de bus de la ville d'Angers.

La dernière section de ce chapitre est consacrée à une ébauche sur la régulation d'un réseau de transport. On décrit succinctement un projet de fluidification d'un noeud ferroviaire. C'est ce problème qui a suscité le travail présenté au §3.3, à savoir la recherche d'une commande en boucle fermée minimisant les retards.

4.1 Généralités

Dans cette section, on décrit brièvement les principaux composants "physiques" d'un système de transport. Un réseau de transport est représenté par deux éléments :

- l'infrastructure,
- les entités de transport.

Cette section est inspirée des travaux de [Hayat and Maouche, 1997] et de [Nait-Sidi-Moh, 2003].

4.1.1 Infrastructure

L'infrastructure d'un réseau décrit l'ensemble des voies ouvertes aux entités de transport. Les éléments de bases de l'infrastructure d'un réseau sont les lignes et les arrêts ou les tronçons, selon le type de système auquel on s'intéresse.

Les lignes sont composées d'un ensemble fini d'arrêts ou de tronçons, selon le point de vue que l'on adopte.

La conception d'une ligne se fait sur des critères géographiques, sociaux et économiques.

La structure d'une ligne peut être plus ou moins complexe selon le rôle de ses arrêts par rapport à ceux des autres lignes et le nombre de véhicules qui la composent. Sur chaque ligne, les entités de transport réalisent une boucle au cours de laquelle elles s'arrêtent aux stations de cette ligne. Ce circuit est réalisé suivant une planification horaire imposée par une étude préalable avec une fréquence qui peut être fixe ou variable. Une ligne est caractérisée par les temps de parcours entre ses arrêts. Dans le cas d'un système de transport empruntant des voies publiques, ces temps sont sujets aux conditions de circulation.

Les arrêts sont les éléments de base d'un système de transport. Ils représentent les noeuds du réseau. Un arrêt représente un espace où les entités de transport effectuant une tournée

s'immobilisent pendant un certains temps pour permettre aux passagers d'embarquer ou de débarquer. On différencie trois types d'arrêt [Descamps, 2002] :

des arrêts simples, ils sont rattachés à une seule ligne, il ne peut donc pas s'effectuer d'échange de passagers sur ces arrêts.

des points d'échange de passagers, ce sont des points de correspondance desservis au moins par deux lignes.

des points de retournement, ces arrêts correspondent à des extrémités de ligne. Les véhicules de transport y font demi-tour.

Les particularités physiques de certains arrêts obligent les entités de transport à se soumettre à des contraintes. C'est, par exemple, le cas, pour un arrêt d'un réseau de bus dont la voie est commune à celles d'autres véhicules (voitures particulières), les bus ne doivent pas s'immobiliser trop longtemps à cet arrêt sous peine de créer un embouteillage. Pour les points d'échange de passagers communs à plusieurs lignes, les véhicules de transport doivent faire face à des problèmes de stationnement puisque les arrêts présentent généralement une capacité limitée pour l'accueil des véhicules.

Pour les systèmes tels que les réseaux ferroviaires, on considère préférablement des tronçons. Les arrêts étant matérialisés par des gares, parfois éloignées les unes des autres, on préfère décomposer l'infrastructure en tronçons. Certains tronçons du réseau peuvent présenter des particularités comme, par exemple, être en commun avec plusieurs lignes de chemin de fer, ou encore présenter un croisement avec d'autres tronçons (le terme utilisé pour décrire ce phénomène est le cisaillement). Un tronçon est caractérisé par sa longueur.

4.1.2 Entités de transport

Le système physique est, en outre, composé d'entités de transport qui sont des véhicules destinés à l'embarquement et au débarquement de passagers. Ils se présentent sous la forme de bus, tramways, trains, *etc.* On caractérise alors le système par :

- le nombre d'entités de transport dont il dispose,
- le type d'entités de transport, c'est-à-dire les caractéristiques techniques des véhicules comme leur capacité d'embarquement, leur encombrement ou leur caractéristique dynamique (accélération, vitesse).

L'étude de [Hayat and Maouche, 1997] montre également que les personnes qui conduisent ces véhicules ont également une influence sur le fonctionnement du réseau.

Les entités de transport sont réparties dans chacune des lignes du réseau. Cette distribution des ressources est fonction de l'importance de la ligne et est variable au cours de la journée de sorte à "renforcer" les lignes au moment où la charge des usagers est importante.

4.2 Pratiques d'exploitation

Le pilotage du système physique consiste à exploiter au mieux les infrastructures et les entités de transport afin de répondre aux objectifs définis par l'exploitant du réseau.

D'après [Hayat and Maouche, 1997] et [Soulhi, 2000], la gestion d'un système de transport se décompose en deux phases : le *tableau de marche* et la *régulation*. Le tableau de marche correspond à la phase de planification du système. Quant à la régulation, elle consiste à analyser les écarts existants entre le fonctionnement théorique défini par la planification et la réalité. De plus, elle met en oeuvre des actions susceptibles de les combler. Ces deux phases se réalisent à des périodes différentes : la planification est définie en temps anticipé tandis que la régulation intervient en temps réel sur l'exploitation du réseau.

4.2.1 Tableau de marche

Le tableau de marche est établi dans l'objectif d'optimiser l'offre de service tout en satisfaisant les contraintes d'exploitation (le nombre de chauffeurs, leurs horaires de travail, la flotte de véhicules, *etc.*). Il est calculé pour des conditions moyennes d'exploitation, et concrètement, cette optimisation permet de définir notamment les points suivants :

1. Le choix d'une qualité de service. Cette étape définit, en particulier, des fréquences de passage pour chaque ligne ainsi que l'allocation des ressources par ligne (nombre de véhicules et chauffeurs). La définition de l'offre de service nécessite au préalable une analyse de la demande de transport : niveau de charge, départs à assurer impérativement, heure de correspondance à respecter, *etc.* La qualité de service est généralement définie par période (typiquement deux heures) puisque la demande des passagers peut varier pendant la journée. De la même façon, la qualité de service recherchée est différente suivant le type de jour (jour de la semaine, week-end) ou suivant la période de l'année (période normale, vacances d'été).
2. La synthèse de la table d'horaires qui définit les heures théoriques de passage des véhicules de transport à chaque arrêt. La régularité de passage des véhicules (*i.e.* les fréquences de passage) à un arrêt est prise en compte lors de la construction de la planification. D'autres éléments comme l'enlèvement de la charge pendant des heures de pointes, *a priori* connues, ou la prise en compte des correspondances, en vue de minimiser l'écart temporel entre les arrivées des véhicules aux stations de correspondance sont intégrés dans le calcul des tables d'horaires.

Le succès du tableau de marche réside dans la pertinence du choix de ses paramètres et dans la bonne estimation des temps de parcours entre les arrêts pour des conditions normales de fonctionnement (une estimation biaisée des temps de parcours fournirait une table d'horaires qui serait difficilement respectable pour les véhicules de transport).

4.2.2 Régulation

Dès lors que l'on s'écarte des conditions moyennes de fonctionnement pour lesquelles le tableau de marche a été défini, il devient nécessaire de réguler le système. La régulation consiste en des ajustements ou des adaptations du tableau de marche en réactions aux conditions réelles d'exploitations. En effet, les conditions réelles de circulation peuvent révéler des divergences entre le modèle utilisé pour la réalisation du tableau de marche et le système réel (départs inutilement nombreux sur une ligne, insuffisance de départ sur une autre ligne, *etc.*). Des dégradations des conditions de circulation ou des incidents matériels sont également des facteurs qui conduisent à des modifications du système de pilotage du réseau.

Ces actions peuvent être décidées par les chauffeurs de bus et/ou un superviseur du réseau (visualisant l'évolution du réseau et pouvant communiquer avec les chauffeurs). On peut distinguer deux types de conditions amenant à des "manœuvres de régulation" :

- *réagir à des perturbations* (panne d'un véhicule, modifications des conditions de circulation, *etc.*). La régulation peut par exemple consister à dérouter un véhicule, ne pas effectuer certains arrêts, transférer les passagers vers une autre ligne. . .
- *améliorer opportunément l'offre de service au sein du réseau*. L'objectif peut, par exemple, être d'absorber un "pic de charge" qui apparaît au sein du réseau. La manœuvre consiste à retarder le départ du véhicule si l'arrivée d'un important contingent d'usagers est imminente (par exemple, à proximité d'une usine à l'heure de la sortie). Il peut aussi s'agir d'assurer des correspondances "intermodales" à certains points du réseau. A des arrêts où différents modes de transport convergent (métro, tramway et/ou train. . .), le but consiste à minimiser les temps de correspondance en attendant les contingents d'usagers en provenance des autres modes de transport.

Dans le cas où le retour du système à l'état initialement prévu est difficile, les modifications apportées sur le système conduisent à la conception d'un nouveau tableau de marche.

Dans cette étude, on se focalise sur la phase correspondant au tableau de marche. Plus précisément, on introduit dans la section suivante un modèle (*max, +*)-linéaire pour un réseau de transport opérant selon son tableau de marche. A partir de ce modèle, on propose :

- *des éléments d'analyse du tableau de marche* dans la section 4.4, en étudiant la stabilité du réseau,
- *des outils de conception du tableau de marche* dans la section 4.5, en présentant une méthode de génération de tables d'horaires.

Un travail d'approche d'applications à la régulation d'un réseau de transport est évoqué dans la section 4.6.

4.3 Modélisation d'un système de transport

Les systèmes de transport peuvent être perçus comme des SED soumis à des phénomènes de synchronisation dus aux correspondances entre les entités de transport. Dans la littérature concernant les systèmes $(max, +)$ -linéaires, on trouve plusieurs références traitant de la modélisation et de la commande de systèmes de transport notamment [Braker, 1993], [Goverde et al., 1998], [Nait-Sidi-Moh, 2003], [de Vries et al., 1998] et [Lahaye et al., 2003] sans être exhaustif. Dans cette section, on présente une modélisation du fonctionnement d'un réseau de transport opérant selon son tableau de marche. Ce modèle est simple et se situe dans la continuité de ceux considérés notamment dans [Braker, 1993] et [Goverde et al., 1998].

Les lignes, les arrêts et les véhicules sont des éléments du tableau de marche qui peuvent se représenter sous la forme d'un GET, ce qui permet une représentation de leur fonctionnement sous la forme d'équations d'état $(max, +)$ -linéaires :

$$\begin{aligned} x(k) &= Ax(k-1) \oplus Bu(k), \\ y(k) &= Cx(k), \end{aligned} \tag{4.1}$$

où $x(k)$ est un vecteur tel que $x_i(k)$ correspond à la date du $k + 1$ -ième départ de véhicule de transport à l'arrêt i du réseau. La matrice A est définie de la façon suivante : $A_{ij} = a_{ij}$ où a_{ij} désigne le temps de parcours de l'arrêt j à l'arrêt i si j précède i , $A_{ij} = \varepsilon$ sinon. Dans le cas de points de correspondance i et j , A_{ij} peut représenter un temps de transit entre deux arrêts en correspondance.

Le vecteur $y(k)$ correspond au vecteur des dateurs associés à des arrêts considérés comme "stratégiques" au sein du réseau (la qualité de service doit plus particulièrement y être respectée). Ces arrêts particuliers sont appréhendés comme les sorties du système.

La table d'horaires du système est modélisée par le vecteur d'entrée $u(k)$, ainsi la variable $u_i(k)$ désigne la date de départ prévue du $k + 1$ -ème véhicule à l'arrêt i . En pratique, les synchronisations des véhicules avec ces tables ne se font qu'à certains arrêts particuliers du réseau tels que les terminus et les départs de ligne. Pour les autres arrêts, la table d'horaires n'a qu'une valeur indicative. Il en découle la structure de la matrice B : $B_{ii} = e$ si la table d'horaires est respectée à l'arrêt x_i , $B_{ij} = \varepsilon$ sinon.

Dans [Braker, 1993], [Goverde et al., 1998], [de Vries et al., 1998], les auteurs prennent pour hypothèses que, pour deux points d'échange de passagers, les correspondances se font dans les deux sens, c'est-à-dire que les véhicules d'une ligne attendent ceux de l'autre ligne aux arrêts de correspondance et *vice versa*. Cette hypothèse a pour conséquence sur le réseau, un

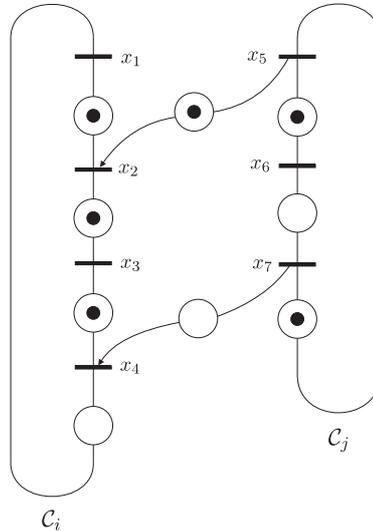


FIG. 4.1: Réseau de transport élémentaire composé de deux lignes.

fonctionnement totalement synchronisé des différentes lignes (la matrice d'évolution est alors irréductible). Plus précisément, le fonctionnement sera dépendant de la ligne la plus lente [Goverde et al., 1998][Nait-Sidi-Moh, 2003, §3.1.2], c'est-à-dire celle qui possède le temps de cycle le plus élevé.

D'autre part, les auteurs considèrent une planification fixe pour chacune des lignes, c'est-à-dire telle que :

$$u(k+1) = \tau \otimes u(k), \quad (4.2)$$

où τ est la période de planification du réseau.

Dans cette étude, nous considérons un fonctionnement plus conforme à la réalité dans le sens où, deux lignes comportant un point d'échange de passagers vont pouvoir n'être que partiellement synchronisées [Houssin et al., 2004] [Houssin et al., 2006b]. Ce mode de fonctionnement peut être illustré par la figure 4.1 qui représente un exemple académique de réseau composé de deux lignes. Dans ce GET, les deux lignes partagent deux arrêts. Ces arrêts communs ont pour but d'établir des correspondances entre les lignes. On prend pour hypothèse de fonctionnement, que les bus de la ligne \mathcal{C}_i doivent attendre ceux de la ligne \mathcal{C}_j pour quitter leur arrêt, autrement dit, les bus de la ligne \mathcal{C}_i sont en correspondance avec ceux de la ligne \mathcal{C}_j . Ce type de phéno-

mène est qualifié de "correspondance partielle" par opposition à "correspondance totale". Dans ces conditions de fonctionnement, la matrice d'évolution est réductible. Pour la figure 4.1, le système est décrit par l'équation (4.1) avec une matrice A réductible :

$$A = \begin{pmatrix} \boxed{A_{ii}} & \varepsilon \\ \boxed{A_{ij}} & \boxed{A_{jj}} \end{pmatrix}$$

Ce type de correspondance a pour conséquence de permettre un fonctionnement des lignes à des dynamiques différentes. Le tableau de marche peut donc être défini de façon plus "souple".

D'autre part, nous considérons que la planification du réseau de transport est décrite par les deux points suivants.

- Tout d'abord, il apparaît que les fréquences de passage des véhicules de transport ne sont généralement pas les mêmes sur les différentes lignes du réseau. En effet, en fonction du taux d'utilisation et/ou de la qualité de service désirée, la période planifiée entre deux passages de véhicules peut varier selon les lignes. Face à ce constat, nous allons être amenés à considérer des conditions de planification spécifiques à chaque ligne, c'est-à-dire une condition sur chaque ensemble d'éléments u_l où l'indice l traduit les différents arrêts d'une même ligne \mathcal{C}_i ¹.
- D'autre part, plutôt que de chercher à imposer une période constante entre deux départs (situation considérée avec la condition (4.2)), il est usuel pour les réseaux de transport [Lin et al., 1995][Strathman et al., 2001] de définir un intervalle de valeurs traduisant les "écarts temporels" possibles entre les départs successifs. Précisément, l'usage est de définir un "écart² minimum" et un "écart maximum" entre deux départs à un même arrêt :

$$\forall l \in \mathcal{C}_i, \quad u_l(k+1) \geq \Delta_i^{min} u_l(k) \quad (4.3)$$

$$\Delta_i^{max} u_l(k) \geq u_l(k+1) \quad (4.4)$$

où Δ_i^{min} et Δ_i^{max} sont les écarts minimum et maximum à imposer sur la ligne \mathcal{C}_i .

Soit de façon générale,

$$u(k+1) \geq \Delta^{min} u(k)$$

et

$$u(k+1) \leq \Delta^{max} u(k),$$

avec Δ^{min} et Δ^{max} deux matrices diagonales de même dimension que la matrice A . On retrouve sur les diagonales de ces matrices, les écarts minimum Δ_i^{min} et maximum Δ_i^{max}

¹Alors que Braker considère la condition (4.2) qui s'applique à la totalité du vecteur $u(k)$.

²Le terme anglais consacré est "headway".

attribués à chaque ligne du réseau C_i .

Remarque 26 *Pour un réseau de bus, l'objectif en imposant la contrainte de planification (4.3), c'est-à-dire en cherchant à fixer un écart minimum entre les bus, est d'éviter la formation d'accumulations de bus sur les lignes, autrement dit, des regroupements de plusieurs bus les uns à la suite des autres. Un tel scénario a tendance à se produire naturellement dès qu'un bus prend du retard ([Lin et al., 1995][Strathman et al., 2001]). En effet, un bus en retard a logiquement plus de passagers à embarquer à chaque arrêt et son retard a donc tendance à augmenter. Au contraire, le bus qui le suit a moins de passagers à enlever, il aura ainsi tendance à évoluer plus vite que prévu au sein du réseau et donc à rattraper le bus précédent. Lorsque l'accumulation est formée, on aboutit à une configuration "contre-productive" puisque généralement seul le bus en "tête de convoi" embarque des passagers et que les bus qui le suivent attendent inutilement. Du point de vue des passagers, ces dysfonctionnements provoquent des accroissements des temps de trajet et des non-respects des horaires prévisionnels. D'autre part, le choix d'un "écart maximum", dans la seconde contrainte de planification (4.4), traduit la volonté d'assurer une qualité de service en spécifiant une fréquence minimale de passage des bus aux arrêts.*

4.4 Stabilité

Une question d'intérêt dans l'analyse d'un système de transport est de pouvoir évaluer sa propension à "absorber" d'éventuels retards (induits par des pannes, ou des accidents,...). Lorsqu'un retard est constaté, il est notamment crucial de pouvoir discerner si celui-ci va se propager ou bien disparaître. Cette propriété a été définie dans [Braker, 1993] comme la propriété de *stabilité* d'un système de transport. Plus précisément, on dira qu'un système de transport planifié est stable si tout retard vis-à-vis de la table d'horaires n'apparaît plus après un nombre fini de cycles.

Remarque 27 *Le notion de stabilité introduite par [Braker, 1993] pour les systèmes de transport est différente de celle généralement considérée dans les systèmes (max, +)-linéaires [Baccelli et al., 1992, §6.6.2] qui considère un système comme instable si dans le GET associé, le fonctionnement peut conduire à une accumulation infinie de jetons dans une de ses places.*

Pour l'étude de la stabilité, nous distinguerons deux points de vue sur les systèmes de transport.

- Les systèmes à correspondance totale [Braker, 1993][Goverde et al., 1998][Heidergott and Vries, 2001] dont la planification est faite de manière régulière.
- Les systèmes à correspondances partielles décrits au §4.3.

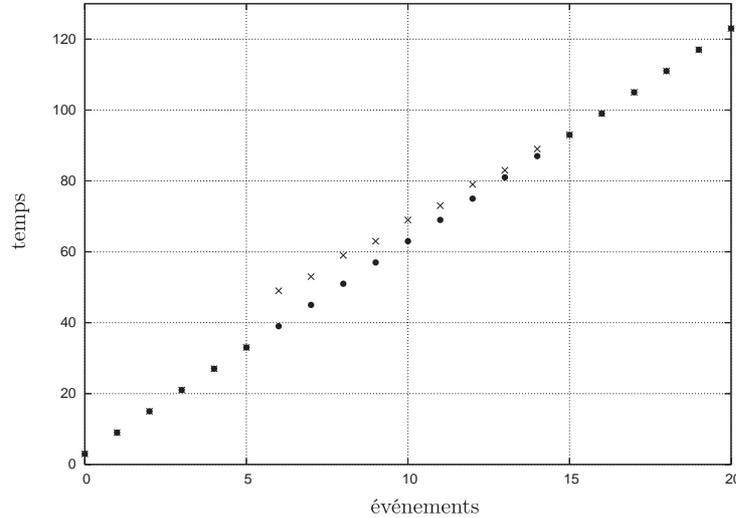


FIG. 4.2: Exemple d'absorption de retard par un réseau stable, ● : $u(k)$, × : $x(k)$.

Le théorème suivant permet de caractériser la stabilité d'un système à correspondance totale avec une période de planification fixe (voir l'équation (4.2)).

Théorème 17 (stabilité) [Braker, 1993, Th 2.4.2] *Un système planifié est stable si et seulement si*

$$\lambda < \tau$$

où τ est le temps de cycle de la table des horaires et λ est la valeur propre de la matrice d'évolution du système.

La figure 4.2 montre le résultat d'un retard introduit sur un réseau stable (planification fixe $\tau = 6$) décrit par l'équation d'évolution :

$$x(k) = \begin{pmatrix} 4 & 8 \\ 2 & 3 \end{pmatrix} x(k-1) \oplus u(k).$$

La valeur propre associée à la matrice d'évolution est $\lambda = 5$. Un retard de 10 unités de temps est introduit lors du départ n°7. Ce retard est totalement absorbé à partir du départ n°15.

Le théorème 17 fournit une condition nécessaire et suffisante de stabilité du système. Cependant, il ne permet pas d'apprécier le degré de stabilité du système, c'est-à-dire la capacité du système à annihiler les retards le plus rapidement possible (en termes de périodes). Pour

évaluer la capacité du système à annuler les retards, [Braker, 1993, § 3.5.2] introduit la notion de marge de stabilité.

Afin de se placer dans un cadre plus réaliste vis-à-vis des réseaux de transport, on considère maintenant la situation où les correspondances sont partielles. Dans ce contexte, et en s'inspirant du théorème 17, on propose une condition sur les écarts minimums Δ_i^{min} entre chaque bus d'une ligne \mathcal{C}_i , suffisante pour garantir la stabilité du réseau de bus.

Proposition 20 *Un système planifié est stable si pour toute ligne \mathcal{C}_i , on a*

$$\Lambda_i < \Delta_i^{min} .$$

On rappelle que Λ_i (introduit au §2.2.3) est défini comme le plus grand élément de l'ensemble formé par la valeur propre du bloc $\boxed{A_{ii}}$ et celles des blocs prédécesseurs de $\boxed{A_{ii}}$.

Preuve : On suppose qu'un retard a été initialement introduit dans le réseau (*i.e.* à $k = 0$) et on montre, afin de prouver la stabilité du système, que ce retard décroît strictement au cours de son évolution.

Pour un arrêt x_l d'une ligne \mathcal{C}_i , la planification impose

$$\begin{aligned} \Delta_i^{min} u_l(k) &\leq u_l(k+1), & \forall k \\ \Rightarrow (\Delta_i^{min})^m u_l(k) &\leq u_l(k+m), & \forall k. \end{aligned}$$

Dans l'algèbre usuelle, on traduit la dernière inégalité par

$$m \times \Delta_i^{min} + u_l(k) \leq u_l(k+m), \quad \forall k.$$

On peut en déduire l'inéquation suivante

$$x_l(k+m) - (m \times \Delta_i^{min} + u_l(k)) \geq x_l(k+m) - u_l(k+m), \quad \forall k.$$

Soit

$$x_l(k+m) - (m \times \Delta_i^{min} + u_l(k)) \geq w_l(k+m), \tag{4.5}$$

où la variable $w_l(k)$, traduisant le retard à l'arrêt l pour l'itéré k , est définie par $w_l(k) = x_l(k) - u_l(k)$.

En supposant que le retard est toujours présent lors du départ $n^{\circ}k + m$, on a : $x(k + m) = A^{k+m}x(0)$ (le retard a été introduit au premier départ). Or, nous avons montré au §2.2.3 qu'il existe deux entiers K_0 et c tels que $\forall k \geq K_0$, l'inéquation $x_l(k + c) \leq \Lambda_i^c x_l(k)$ (dans $\overline{\mathbb{Z}}_{max}$) soit vérifiée. Dans ces conditions et en prenant $m = c$ l'équation (4.5) devient

$$(c \times \Lambda_i + x_l(k)) - (c \times \Delta_i^{min} + u_l(k)) \geq w_l(k + c).$$

Par hypothèse $\Lambda_i < \Delta_i^{min}$, ce qui nous permet d'écrire

$$\begin{aligned} (c \times \Delta_i^{min} + x_l(k)) - (c \times \Delta_i^{min} + u_l(k)) &> w_l(k + c) \\ \Leftrightarrow x_l(k) - u_l(k) &> w_l(k + c) \\ \Leftrightarrow w_l(k) &> w_l(k + c). \end{aligned}$$

Ce qui prouve que les retards décroissent strictement au fil des départs. D'autre part, $x(k)$ est toujours supérieur ou égal à $u(k)$, le retard ne peut donc pas prendre une valeur négative. Finalement, on en déduit qu'il existe donc un entier k tel que $w(k) = 0$.

□

4.5 Génération de tables d'horaires

La synthèse des tables d'horaires constitue le dernier point du tableau de marche. Les tables d'horaires définissent les dates de passage théoriques des entités de transport. De nombreux travaux ont été menés sur le sujet, on peut citer par exemple [Palma and Lindsey, 2000], [Ceder et al., 2001] et dans l'algèbre $(max, +)$ [Katz, 2005].

En considérant les systèmes de transport comme des SED, la table d'horaires peut être vue comme la commande du réseau de transport. Ainsi, le problème de génération de tables d'horaire peut se ramener à un problème de commande de SED. Dans cette section, on applique précisément l'approche de commande proposée au §3.2 pour le résoudre. Le problème est donc présenté en le décomposant sous la forme de contraintes sur l'état du modèle proposé précédemment, celles-ci constitueront l'objectif de la commande.

4.5.1 Présentation du problème

Dans ce qui suit, nous décrivons un ensemble de conditions que doit satisfaire la table d'horaires à synthétiser. Comme nous avons pu le mentionner au §4.2.1, il peut s'agir d'assurer

la régularité de passage des véhicules, de prendre en compte des pics d'arrivées de passagers à des horaires connus, *etc.* En respectant le formalisme adopté au §3.1.1, nous les traduisons sous la forme de contraintes sur l'état du système. Par la suite, les variables k_{d_i} , respectivement k_{f_i} , désignent les numéros des premières occurrences, respectivement dernières occurrences, à contrôler pour l'arrêt x_i .

- Dans un premier temps, on spécifie une qualité de service désirée aux arrêts stratégiques du réseau (ceux définis par la matrice C). Cette qualité de service s'exprime sous la forme d'échéances (par exemple le départ du k -ième véhicule à l'arrêt i doit avoir lieu avant la date t). Dans ces conditions, on peut utiliser la contrainte (3.12) pour mettre en équation la qualité de service attendue aux arrêts stratégiques. Si on note $z(k)$ le vecteur représentant ces dates au plus tard pour les arrêts stratégiques, il en découle dans $\mathcal{M}_{in}^{ax}[\gamma, \delta]$:

$$x \preceq C \backslash z \quad (4.6)$$

- Le respect d'une contrainte d'écart-temps minimum permet d'éviter les phénomènes d'accumulation. Dans la partie 4.3, nous avons spécifié la condition (4.3) pour la table d'horaires. Cependant, les véhicules de transport ne respectent pas la table d'horaires à tous les arrêts du réseau, ce qui a pour conséquence directe que cette contrainte, bien que spécifiée par la table d'horaires, n'est pas nécessairement respectée par les véhicules. Plutôt que de fixer la contrainte sur le vecteur de commande, on propose de la spécifier directement sur l'état du système.

Ce type de spécification est décrit par la contrainte (3.3). On rappelle ici son expression dans le dioïde des séries formelles :

$$x \preceq \gamma^{-1}(\Delta^{min})^{-1}x, \quad (4.7)$$

$$\text{où } \Delta^{min} = \begin{pmatrix} \delta^{\Delta_1^{min}} & \varepsilon & \varepsilon \\ \varepsilon & \delta^{\Delta_2^{min}} & \varepsilon \\ \varepsilon & \varepsilon & \ddots \end{pmatrix}.$$

On rappelle que l'écart minimum entre deux départs de bus peut être différent d'une ligne à une autre.

- D'autre part, le respect de la contrainte d'écart-temps maximum entre deux véhicules permet d'assurer une fréquence minimum du passage des bus dans le réseau. Pour les mêmes raisons que celles évoquées pour la contrainte d'écart temps minimum, on spécifie la contrainte directement sur l'état. L'équation (3.5) décrit cette spécification.

Dans $\mathcal{M}_{in}^{ax}[\gamma, \delta]$, la contrainte se traduit par l'inégalité suivante

$$x \preceq \gamma \Delta^{max} x \oplus x_d, \quad (4.8)$$

$$\text{où } \Delta^{max} = \begin{pmatrix} \delta \Delta_1^{max} & \varepsilon & \varepsilon \\ \varepsilon & \delta \Delta_2^{max} & \varepsilon \\ \varepsilon & \varepsilon & \ddots \end{pmatrix} \text{ et } x_d \text{ est défini par } x_{d_i} = \gamma^{k_{d_i}} \delta^{x_i(k_{d_i})}.$$

Tout comme la contrainte d'écart-temps minimum, la contrainte d'écart temps maximum peut être différente suivant les lignes.

- A certaines heures (connues) de la journée des pics de charge apparaissent. Ceux-ci peuvent être dus à l'intermodalité des transports ou aux activités urbaines (sorties d'école, sortie d'usine...). Dans ce cas, on souhaite qu'un ou plusieurs départs aient lieu à l'arrêt concerné dans un intervalle de dates données afin d'absorber un tel pic de charge. La contrainte (3.11) nous permet de spécifier cet objectif. Plus précisément, pour un arrêt x_i elle se traduit par :

$$val(Pr_{t_i+r+1}(x_i)) \preceq m \otimes val(Pr_{t_i}(x_i)), \quad (4.9)$$

où m est le nombre de départs souhaités à l'arrêt x_i pendant l'intervalle $[t_j, t_j + r]$ pour absorber le pic.

- On peut souhaiter limiter le temps d'attente des véhicules de transport à certains arrêts. Il peut s'agir d'un objectif de qualité de service (une attente supérieure à un "seuil" donné peut être perçue négativement par les usagers) ou d'une contrainte matérielle (cas d'un arrêt de bus situé sur une voie commune avec les voitures). Dans un SED modélisé par un GET, cette contrainte revient à considérer que le temps de séjour des jetons ne doit pas excéder une durée maximale dans certaines places. Dans le cas des système de transport, il suffit d'adapter la contrainte (3.7). Plus précisément, cette contrainte s'exprime par :

$$x \preceq \Phi x, \quad (4.10)$$

où Φ est définie dans (3.8). En se référant aux notations de la définition de Φ dans (3.8), il faut garder à l'esprit que τ_{ji} représente la somme du temps de parcours de l'arrêt x_i à l'arrêt x_j et le temps d'attente maximum souhaité à l'arrêt x_j .

4.5.2 Résolution

Pour résoudre ce problème de synthèse des tables d'horaires, on applique la commande de la section 3.2. Les contraintes décrites au paragraphe précédent constituent l'objectif de commande. En tenant compte des conditions initiales et finales du système, on peut mettre l'ensemble de ces contraintes sous la forme

$$x \preceq (g_i(x) \wedge \omega) \oplus \nu,$$

où $g_i(x)$ désigne la contrainte i et, ω et ν sont des vecteurs définis au §3.2.1. Plus précisément, ω est déterminé par l'état initial du système, (c'est-à-dire le nombre caractéristique de chaque variable d'état) et ν est calculé à partir des dernières occurrences désirées pour les sorties, ce qui, dans notre cas, correspond aux derniers départs désirés des arrêts stratégiques.

Comme on l'a vu au §3.2.4, l'hypothèse du fonctionnement au plus tôt permet de traduire les contraintes précédentes sous la forme d'une inéquation sur le vecteur de commande, *i.e.* la table d'horaires dans notre cas. La table d'horaire optimale est donc la plus grande solution de l'inéquation

$$u \preceq g'_1(u) \wedge \dots \wedge g'_5(u) \wedge Pr_+(u) = f(u),$$

où chaque g'_i désigne une des contraintes. La proposition 12 fournit une méthode itérative pour le calcul de la table d'horaires. De plus, la proposition 13 nous assure la convergence de cette méthode puisque, parmi les contraintes, nous considérons une trajectoire de référence à poursuivre (elle correspond à la contrainte de qualité de service attendue aux arrêts stratégiques).

4.5.3 Discussion et remarques

Les écart-temps minimum et maximum (qui déterminent les fréquences de passage des véhicules de transport sur les lignes) sont généralement fixés par période (typiquement deux heures pour les réseaux de bus). La méthode de calcul exposée précédemment doit alors être utilisée pour synthétiser les tables d'horaires pour chacune de ces périodes.

Les contraintes spécifiées dans les inéquations (4.6), (4.8), (4.9) et (4.10) sont utilisées pour améliorer la qualité de service perçue par les voyageurs car elles contribuent principalement à minimiser les temps d'attente.

La contrainte (4.7) présente plutôt un intérêt pour la compagnie de transport. Elle permet de définir une fréquence maximale de départ de bus et donc, par conséquent, de limiter les

coûts liés à la circulation des bus. De même, le critère considéré induit des départs de véhicule les plus tardifs possibles (notamment au terminus des lignes), ce qui permet à la compagnie de minimiser le nombre de véhicules en circulation dans le réseau. La compagnie peut alors limiter les coûts grâce à une meilleure allocation des véhicules et des chauffeurs.

A notre connaissance, notre approche est assez semblable de celle adoptée dans [Ceder et al., 2001]. Les auteurs présentent une procédure pour la synthèse de tables d'horaires maximisant les correspondances, ce qui correspond à la maximisation du nombre d'arrivées simultanées de véhicules aux points de correspondances. Dans un premier temps, ils déterminent une procédure d'optimisation linéaire qui se montre efficace sur de petits réseaux (5 lignes, 5 arrêts). Une méthode heuristique est aussi proposée pour résoudre des problèmes de taille plus importante en un temps raisonnable. Les auteurs présentent un exemple réel qui se compose d'une partie d'un réseau de bus en Israël : 3 arrêts de correspondances auxquels les passagers ont accès à 7 lignes de bus. Plusieurs différences existent avec notre approche, ce qui rend difficile la comparaison. En premier lieu, les tables d'horaires sont calculées dans [Ceder et al., 2001] sans tenir compte de l'allocation des véhicules aux lignes, tandis que nous considérons dans notre modèle que les véhicules sont alloués aux lignes. Dans [Ceder et al., 2001], les synchronisations peuvent avoir lieu à n'importe quel arrêt du réseau et à n'importe quel moment. Notre approche permet d'obtenir des synchronisations, partielles ou totales, à des points de correspondance du réseau pour tous les départs de véhicule. Enfin, les contraintes telles que les échéances des derniers départs des arrêts stratégiques (4.6), les pics de charge (4.9) et les temps d'attente maximum (4.8) ne sont pas considérés dans [Ceder et al., 2001].

4.5.4 Exemple

Nous considérons trois lignes du réseau de bus de la ville d'Angers (France, Maine-et-Loire) représentées sur la figure 4.3. Seuls les arrêts auxquels les bus se synchronisent avec la table d'horaires ont été représentés.

Au niveau des points d'échange, nous avons les correspondances suivantes :

Ralliement : Les arrêts x_{29} et x_{35} sont respectivement en correspondance avec les arrêts x_{15} et x_{23} .

Foch : Les arrêts x_4 et x_{10} sont respectivement en correspondance avec les arrêts x_{16} et x_{22} .

Churchill : Les arrêts x_6 et x_8 sont respectivement en correspondance avec les arrêts x_{31} et x_{33} .

On considère que le temps de correspondance entre ces arrêts est nul, c'est-à-dire $A_{29\ 15} = \gamma^0\delta^0$, $A_{35\ 23} = \gamma^0\delta^0$, $A_{4\ 16} = \gamma^0\delta^0$, $A_{10\ 22} = \gamma^0\delta^0$, $A_{6\ 31} = \gamma^0\delta^0$ et $A_{8\ 33} = \gamma^0\delta^0$.

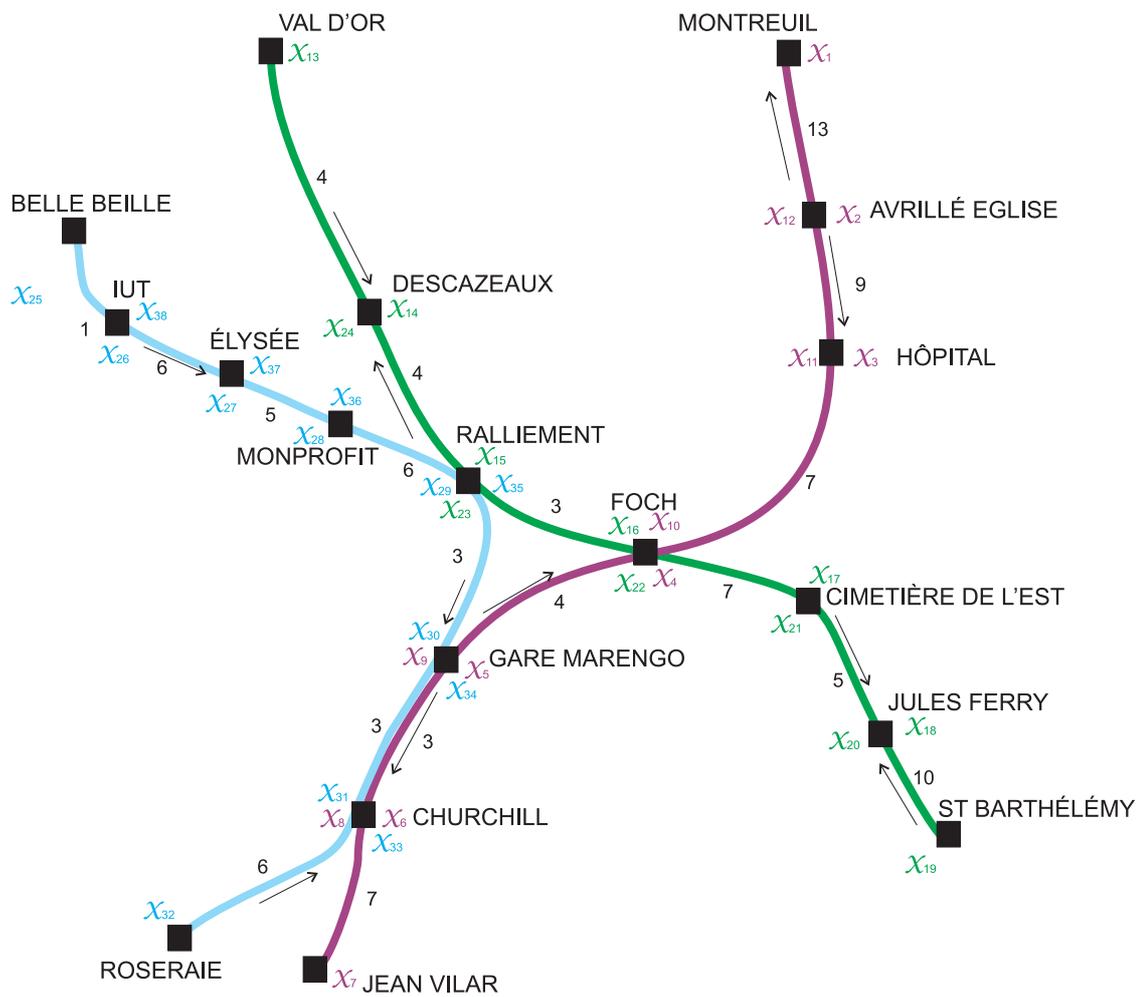


FIG. 4.3: Trois lignes du réseau de bus de la ville d'Angers.

Nous appliquons la méthode proposée au §4.5 pour calculer la table d'horaires de la première période de deux heures de la journée, soit de 6:30 à 8:30.

Le comportement dynamique du système est tel qu'il est décrit par le modèle d'état (4.1) avec $B = Id$ (la table d'horaires est respectée à tous les arrêts qui sont représentés sur la figure 4.3) et on définit C tel que les arrêts stratégiques soient les terminus, à savoir, x_1 , x_{13} et x_{25} .

On considère que, initialement, 6 bus sont stationnés à l'arrêt x_1 , 6 bus à l'arrêt x_7 , 6 bus à l'arrêt x_{13} , 6 bus à l'arrêt x_{19} , 5 bus à l'arrêt x_{25} et 5 bus à l'arrêt x_{32} . Puisque l'on commande tout les arrêts représentés sur la figure 4.3, le calcul des conditions initiales conduit à un vecteur ω tel que $\forall i, \omega_i = \gamma^0 \delta^*$.

On souhaite 14 (respectivement 17 et 15) départs aux arrêts de la ligne Montreuil - Jean Vilar (respectivement aux arrêts de la ligne Val d'or - St Barthélemy et aux arrêts de la ligne Belle Beille - Roseraie) pendant la période considérée. Le tableau 4.1 fournit la qualité de service attendue aux arrêts $y_1 = x_1$, $y_2 = x_{13}$ et $y_3 = x_{25}$ (un départ à l'arrêt x_1 est attendu entre 6:30 et 6:50, quatre départs sont attendus à l'arrêt x_1 entre 6:50 et 7:50, et ainsi de suite).

	6:30	6:50	7:10	7:30	7:50	8:10	8:30
x_1 (Montreuil)	1	4	3	3	1	2	
x_{13} (Val d'or)	1	2	3	3	4	4	
x_{25} (Belle Beille)	1	2	4	4	2	2	

TAB. 4.1: Départs prévus aux arrêts stratégiques.

En tenant compte du nombre de départs désirés pour chacune des lignes, on obtient les conditions finales suivantes : $\nu_i = \gamma^{14} \delta^\top$ pour $1 \leq i \leq 12$ (ligne Montreuil - Jean Vilar), $\nu_i = \gamma^{17} \delta^\top$ pour $13 \leq i \leq 24$ (ligne Val d'or - St Barthélémy) et $\nu_i = \gamma^{15} \delta^\top$ pour $25 \leq i \leq 38$ (ligne Belle Beille - Roseraie).

Les contraintes suivantes sont spécifiées pour le réseau :

▷ La contrainte (4.6) se déduit directement du tableau 4.1, on obtient :

$$z_1 = \gamma^0 \delta^{410} \oplus \gamma^1 \delta^{430} \oplus \gamma^5 \delta^{450} \oplus \gamma^8 \delta^{470} \oplus \gamma^{11} \delta^{490} \oplus \gamma^{12} \delta^{510},$$

$$z_2 = \gamma^0 \delta^{410} \oplus \gamma^1 \delta^{430} \oplus \gamma^3 \delta^{450} \oplus \gamma^6 \delta^{470} \oplus \gamma^9 \delta^{490} \oplus \gamma^{13} \delta^{510}$$

et

$$z_3 = \gamma^0 \delta^{410} \oplus \gamma^1 \delta^{430} \oplus \gamma^3 \delta^{450} \oplus \gamma^7 \delta^{470} \oplus \gamma^{11} \delta^{490} \oplus \gamma^{13} \delta^{510}$$

pour lesquels les exposants en δ correspondent aux dates données en minutes depuis 00:00 (par exemple δ^{410} représente 6:50).

- ▷ Les écarts-temps minimums sont données par $\Delta_{ii}^{min} = \delta^5$ pour $1 \leq i \leq 12$, $\Delta_{ii}^{min} = \delta^4$ pour $13 \leq i \leq 24$ et $\Delta_{ii}^{min} = \delta^5$ pour $25 \leq i \leq 38$.
- ▷ Le temps maximum entre deux départs de bus est $\Delta_{ii}^{max} = \delta^{15}$ pour $1 \leq i \leq 24$ et $\Delta_{ii}^{max} = \delta^{16}$ pour $25 \leq i \leq 38$.
- ▷ A certaines périodes connues de la matinée, des pics de charge apparaissent dans le réseau, particulièrement pour les arrêts x_5 et x_9 . Des départs à ses arrêts doivent permettre d'absorber la charge. Pour x_5 , on désire :
- un départ entre 7:20 et 7:25,
 - un départ entre 7:30 et 7:35,
 - un départ entre 7:40 et 7:42.
- Pour x_9 , on désire :
- un départ entre 7:15 et 7:20,
 - un départ entre 7:25 et 7:30,
 - un départ entre 7:55 et 7:60.
- On désire également 3 départs et donc (3 arrivées) de bus entre 7:30 et 7:50 à l'arrêt x_{26} (IUT).
- ▷ En raison du nombre limité de places, les bus ne sont pas autorisés à stationner plus de 2 minutes à la station Foch qui correspond aux arrêts x_4 , x_{10} , x_{16} et x_{22} . En considérant les temps de trajet, on obtient $\phi_{4\ 3} = \gamma^0 \delta^9$, $\phi_{10\ 9} = \gamma^0 \delta^6$, $\phi_{16\ 15} = \gamma^0 \delta^5$ and $\phi_{22\ 21} = \gamma^0 \delta^9$.

La méthode itérative de la proposition 12 a été implémentée à l'aide de la librairie [SWtool, 2005b] pour notre exemple. Le calcul converge en 16 itérations (pour un temps de calcul de 198 sec. sur un Pentium 4 2.4 GHz).

Montreuil → Jean Vilar														
Montreuil	6:47	6:55	7:00	7:05	7:10	7:20	7:25	7:30	7:40	7:45	7:50	8:05	8:20	8:30
Avrillé église	7:00	7:15	7:22	7:37	7:52	8:07	8:22	8:37	8:52	9:07	9:22	9:37	9:52	10:07
Hôpital	7:09	7:24	7:31	7:46	8:01	8:16	8:31	8:46	9:01	9:16	9:31	9:46	10:01	10:16
Foch	7:16	7:31	7:38	7:53	8:08	8:23	8:38	8:53	9:08	9:23	9:38	9:53	10:08	10:23
Gare Marengo	7:20	7:35	7:42	7:57	8:12	8:27	8:42	8:57	9:12	9:27	9:42	9:57	10:12	10:27
Churchill	7:23	7:38	7:53	8:08	8:23	8:38	8:53	9:08	9:23	9:38	9:53	10:08	10:23	10:38

Jean Vilar → Montreuil														
Jean Vilar	6:42	6:47	6:57	7:02	7:07	7:20	7:30	7:45	8:00	8:15	8:30	8:45	9:00	9:15
Churchill	6:49	6:54	7:04	7:09	7:14	7:27	7:37	7:52	8:07	8:22	8:37	8:52	9:07	9:22
Gare Marengo	6:52	6:57	7:07	7:12	7:17	7:30	7:40	7:55	8:10	8:25	8:40	8:55	9:10	9:25
Foch	6:56	7:01	7:11	7:16	7:21	7:36	7:46	8:01	8:16	8:31	8:46	9:01	9:16	9:31
Hôpital	7:03	7:08	7:18	7:23	7:28	7:43	7:58	8:08	8:23	8:38	8:53	9:08	9:23	9:38
Avrillé église	7:12	7:17	7:27	7:32	7:37	7:52	8:07	8:17	8:32	8:47	9:02	9:17	9:32	9:47

Val d'or → St Barthélemy																	
Val d'or	6:49	6:54	7:09	7:14	7:24	7:30	7:42	7:46	7:50	7:58	8:02	8:06	8:10	8:18	8:22	8:26	8:30
Descazeaux	6:53	6:58	7:13	7:18	7:28	7:43	7:58	8:13	8:28	8:43	8:58	9:13	9:28	9:43	9:58	10:13	10:28
Ralliement	6:57	7:02	7:17	7:22	7:32	7:47	8:02	8:17	8:32	8:47	9:02	9:17	9:32	9:47	10:02	10:17	10:32
Foch	7:00	7:05	7:20	7:25	7:35	7:50	8:05	8:20	8:35	8:50	9:05	9:20	9:35	9:50	10:05	10:20	10:35
Cimetière	7:07	7:12	7:27	7:32	7:42	7:57	8:12	8:27	8:42	8:57	9:12	9:27	9:42	9:57	10:12	10:27	10:42
Jules Ferry	7:12	7:17	7:32	7:37	7:47	8:02	8:17	8:32	8:47	9:02	9:17	9:32	9:47	10:02	10:17	10:32	10:47

St Barthélemy → Val d'or																	
St Barthélemy	6:34	6:39	6:49	6:54	6:59	7:14	7:22	7:27	7:42	7:47	7:57	8:12	8:27	8:42	8:57	9:12	9:27
Jules Ferry	6:44	6:49	6:59	7:04	7:09	7:24	7:32	7:37	7:52	7:57	8:07	8:22	8:37	8:52	9:07	9:22	9:37
Cimetière	6:49	6:54	7:04	7:09	7:14	7:29	7:37	7:42	7:57	8:02	8:12	8:27	8:42	8:57	9:12	9:27	9:42
Foch	6:56	7:01	7:11	7:16	7:21	7:36	7:44	7:49	8:04	8:09	8:19	8:34	8:49	9:04	9:19	9:34	9:49
Ralliement	7:12	7:22	7:27	7:32	7:37	7:42	7:47	7:52	8:07	8:12	8:22	8:37	8:52	9:07	9:22	9:37	9:52
Descazeaux	7:38	7:42	7:46	7:54	7:58	8:02	8:06	8:14	8:18	8:22	8:26	8:41	8:56	9:11	9:26	9:41	9:56

Belle Beille → **Roseraie**

Belle Beille	6:48	6:58	7:04	7:19	7:24	7:30	7:40	7:45	7:50	7:55	8:00	8:05	8:10	8:25	8:30
IUT	6:49	6:59	7:05	7:20	7:25	7:41	7:57	8:13	8:29	8:45	9:01	9:17	9:33	9:49	10:05
Élysée	6:55	7:05	7:11	7:26	7:31	7:47	8:03	8:19	8:35	8:51	9:07	9:23	9:39	9:55	10:11
Monprofit	7:00	7:10	7:16	7:31	7:36	7:52	8:08	8:24	8:40	8:56	9:12	9:28	9:44	10:00	10:16
Rallieinent	7:06	7:16	7:22	7:37	7:42	7:58	8:14	8:30	8:46	9:02	9:18	9:34	9:50	10:06	10:22
Gare	7:12	7:22	7:28	7:43	7:48	8:04	8:20	8:36	8:52	9:08	9:24	9:40	9:56	10:12	10:28
Churchill	7:15	7:25	7:31	7:46	7:51	8:07	8:23	8:39	8:55	9:11	9:27	9:43	9:59	10:15	10:31

Roseraie → **Belle Beille**

Roseraie	6:43	6:48	6:58	7:03	7:08	7:21	7:31	7:37	7:52	7:57	8:13	8:29	8:45	9:01	9:17
Churchill	6:49	6:54	7:04	7:09	7:14	7:27	7:37	7:43	7:58	8:03	8:19	8:35	8:51	9:07	9:23
Gare	7:06	7:16	7:21	7:26	7:31	7:36	7:41	7:46	8:01	8:06	8:22	8:38	8:54	9:10	9:26
Rallieinent	7:12	7:22	7:27	7:32	7:37	7:42	7:47	7:52	8:07	8:12	8:28	8:44	9:00	9:16	9:32
Monprofit	7:18	7:28	7:33	7:38	7:43	7:48	7:53	7:58	8:13	8:18	8:34	8:50	9:06	9:22	9:38
Élysée	7:23	7:33	7:38	7:43	7:48	7:53	7:58	8:03	8:18	8:23	8:39	8:55	9:11	9:27	9:43
IUT	7:29	7:39	7:44	7:49	7:54	7:59	8:04	8:09	8:24	8:29	8:45	9:01	9:17	9:33	9:49

4.6 Une ébauche pour la régulation d'un nœud ferroviaire

Les travaux présentés au §3.3 (sur une commande en boucle fermée minimisant les retards) ont à l'origine été développés à des fins applicatives dans le domaine des transports. Cette étude a en effet été initiée suite à une présentation des activités du LISA à la direction de l'ingénierie de la SNCF. Un projet de pilote de fluidification ferroviaire sur le futur nœud de Baudrecourt (Jonction ligne Metz-Rémilly / Rémilly-Forbach / Ligne TGV Est / Rémilly Strasbourg) nous a, à cette occasion, été soumis.

Le problème peut se résumer schématiquement comme suit : à un croisement de plusieurs lignes ferroviaires, établir les consignes de vitesse des trains en amont de celui-ci afin d'y éviter tout conflit³.

Plus précisément, nous avons orienté notre étude sur la phase de régulation du réseau de transport que constitue le nœud ferroviaire : supposant que la planification et l'ordonnancement des trains ont été réalisés en temps différé, on a cherché à apporter des outils permettant d'actualiser en temps-réel les consignes de vitesse des trains en fonction des mesures de localisation des véhicules.

Le problème a été abordé comme suit :

- nous avons pensé à un modèle $(\max,+)$ linéaire pour traduire le fonctionnement planifié du nœud (supposé sans "conflit" ou violation des règles de sécurité). Les entrées de ce modèle correspondent aux commandes envoyées aux trains, et l'état traduit leur localisation.
- à partir de la localisation réelle des trains (mesure d'une partie de l'état), l'idée a été d'établir un correcteur (sur une boucle de retour de l'état) qui ajuste les commandes envoyées aux trains en vue de minimiser leurs retards tout en satisfaisant les règles de sécurité au sein du nœud⁴.

Cette démarche nous a finalement amené à considérer le problème de commande présentée au §3.3, à savoir :

- formuler l'objectif de commande sous la forme de contraintes sur l'état d'un système $(\max,+)$ linéaire (en vue de prendre en compte les règles de sécurité du nœud).
- synthétiser un correcteur, situé sur une boucle de retour de l'état (pour exploiter les données réelles de localisation des trains), de transfert minimal (afin de minimiser les retards induits par la correction).

³L'intérêt prospectif de la SNCF pour ce nouveau nœud réside dans "l'obsolescence" des techniques existantes avec l'apparition de nouveaux éléments technologiques (notamment la possibilité de localisation GPS de certains trains).

⁴Ces règles peuvent, par exemple, indiquer que deux trains doivent être séparés d'un tronçon.

Les résultats présentés au §3.3 ne répondent que partiellement au problème initial de fluidification d'un nœud ferroviaire :

- seulement une partie des règles de sécurité du nœud a pu être traduite dans notre formalisme de contraintes sur l'état : les règles visant à exclure l'occupation simultanée de tronçons par plusieurs trains (phénomènes d'exclusion mutuelle) restent notamment à être traduites sous la forme espérée (3.2),
- nous sommes seulement en mesure d'approcher le transfert minimal recherché pour le correcteur.

Ces constats expliquent que nous n'ayons pas ici détaillé plus encore l'application.

Conclusion

Ce mémoire porte sur la commande des systèmes $(max, +)$ -linéaires.

L'originalité de notre première contribution réside dans la définition d'un nouvel objectif pour la commande en juste-à-temps des systèmes $(max, +)$ -linéaires. Contrairement aux commandes proposées jusque-là qui visent à imposer un transfert entrée-sortie ou une poursuite de trajectoire en sortie, nous avons considéré comme objectif le respect d'un ensemble de contraintes définies sur l'état du système. L'apport de ce nouvel objectif est de pouvoir spécifier, de façon plus précise, le comportement désiré du système. Les résultats sur les points fixes d'applications isotones définies sur des dioïdes sont utilisés pour la synthèse de la loi de commande.

Les correcteurs adjoints aux systèmes $(max, +)$ -linéaires peuvent seulement ralentir le système. Les critères employés jusqu'à présent visent à ralentir au maximum le système tout en assurant un objectif de commande. Dans ce mémoire, nous avons proposé un correcteur qui au contraire ralentit le moins possible la dynamique du processus tout en satisfaisant le nouvel objectif de commande décrit au paragraphe précédent (le respect d'un ensemble de contraintes sur l'état du système). Des développements nouveaux sur les points fixes d'applications antitones sont utilisés pour la synthèse de ce correcteur.

Puisque leur évolution est soumise à des événements, les réseaux de transport peuvent être abordés comme des SED, et plus particulièrement comme des systèmes dont l'évolution présente une description linéaire dans certains dioïdes. A partir de ce modèle, nous avons proposé des éléments d'analyse de performances. Plus précisément, on a étudié la stabilité. Une autre contribution à l'étude des réseaux de transport a été de proposer une méthode de synthèse des tables d'horaires. Cette méthode repose sur la commande en juste-à-temps avec pour objectif de commande la prise en compte de contraintes sur l'état du système. Nous avons également appliqué cette méthode de calcul de tables d'horaires à une partie du réseau de bus de la ville d'Angers. Il conviendrait maintenant de l'appliquer à l'ensemble d'un réseau réel, et de la confronter aux approches (reposant sur de la programmation linéaire en variables mixtes) utilisées par les exploitants de réseau.

La commande en boucle fermée minimisant les retards a été étudiée en vue de répondre à un problème de fluidification d'un nœud ferroviaire. Cette étude devra être approfondie pour pouvoir être appliquée à ce problème. En effet, seule une partie de l'ensemble des règles de sécurité à prendre en compte dans le nœud ferroviaire ont pu être traduites dans le formalisme adopté pour la spécification des contraintes. Une perspective à court terme serait de compléter ce jeu de contraintes afin de rendre applicable la commande.

De plus, le correcteur obtenu ne possède pas nécessairement un transfert minimal. Une extension possible de notre travail serait d'approfondir les résultats sur les points fixes d'applications antitones en vue d'affiner la méthode de synthèse dans ce sens. Les structures de commande, que sont le retour d'état sur un ensemble restreint de variables d'état et le retour d'état sur l'entrée, semblent pouvoir être améliorées. En particulier, on peut envisager de mettre à jour des conditions moins restrictives pour l'utilisation de ces deux structures.

D'autre part, il serait intéressant d'explorer le comportement des commandes proposées ici dans le cadre des systèmes incertains. Les systèmes de transport, par exemple, peuvent être perçus comme des systèmes incertains puisque les temps de trajet sont dépendants de conditions extérieures (conditions climatiques, circulation dans le réseau). La mise à jour d'éléments sur la robustesse des commandes semble opportune. A l'aide d'outils existants comme les dioïdes d'intervalles ou les systèmes $(max, +)$ stochastiques, on peut envisager d'adapter les structures de contrôle réalisées ici en considérant des incertitudes relatives au système.

Annexe A

Détails des exemples

A.1 Exemple 3.2.5, p.81

Dans cet exemple, l'évolution du système est régie par le modèle d'état (2.10) avec

$$A = \begin{pmatrix} \gamma^2 \delta^4 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \delta^2 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \gamma \delta & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \delta^3 & \delta^2 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \gamma \delta & \varepsilon \end{pmatrix},$$

$$B = \begin{pmatrix} e & \varepsilon \\ \varepsilon & e \\ \varepsilon & \varepsilon \\ \varepsilon & \varepsilon \\ \varepsilon & \varepsilon \\ \varepsilon & \varepsilon \end{pmatrix},$$

et

$$C = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & e \end{pmatrix}.$$

Le tableau suivant nous renseigne sur les conditions terminales de la commande :

x_i	k_{d_i}	k_{f_i}
x_1	0	3
x_2	0	2
x_3	0	3
x_4	1	3
x_5	0	3
x_6	1	4

Les vecteurs ν et ω sont alors définis par

$$\omega = \begin{pmatrix} \gamma^0 \delta^* \\ \gamma^0 \delta^* \\ \gamma^0 \delta^* \\ \gamma^1 \delta^* \\ \gamma^0 \delta^* \\ \gamma^1 \delta^* \end{pmatrix} \text{ et } \nu = \begin{pmatrix} \gamma^4 \delta^* \\ \gamma^3 \delta^* \\ \gamma^4 \delta^* \\ \gamma^4 \delta^* \\ \gamma^4 \delta^* \\ \gamma^5 \delta^* \end{pmatrix}.$$

On a la contrainte $x_5 \preceq \gamma^{-2} x_4$ (pas plus de 2 jetons entre x_4 et x_5). Sur le vecteur d'état, la contrainte s'exprime par

$$x \preceq \begin{pmatrix} e & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & e & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & e & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & e & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \gamma^{-2} & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & e \end{pmatrix} x.$$

A.2 Exemple 3.3.3, p.86

Concernant cet exemple, les matrices A et B sont définies de la façon suivante

$$A = \begin{pmatrix} \gamma^2\delta^4 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \delta^2 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \gamma\delta & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \delta^3 & \delta^2 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \gamma\delta & \varepsilon \end{pmatrix}$$

et

$$B = \begin{pmatrix} e & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & e & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{pmatrix}.$$

On présente maintenant la traduction des contraintes dans $\mathcal{M}_{in}^{ax}[\gamma, \delta]$.

- on ne doit pas avoir plus de 3 jetons dans le chemin allant de la transition x_3 à la transition x_6 :

$$\gamma^{3-1}x_6 = \gamma^2x_6 \preceq x_3$$

- les jetons ne doivent pas séjourner plus de 3 unités de temps dans la place située entre x_4 et x_5 :

$$\delta^{-3}x_5 \preceq x_4.$$

La matrice ϕ correspond alors à

$$\phi = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \gamma^2 \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \delta^{-3} & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{pmatrix}.$$

Le calcul des éléments u et v donne

$$v = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \gamma^4 \delta^4 (\gamma^3 \delta^4)^* & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \delta^2 (\gamma^2 \delta^4)^* & \gamma^4 \delta^4 (\gamma^3 \delta^4)^* & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \gamma^4 \delta^7 (\gamma^3 \delta^4)^* & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \gamma^5 \delta^8 (\gamma^3 \delta^4)^* & \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{pmatrix}$$

et

$$u = \varepsilon,$$

où ε désigne ici une matrice $n \times n$.

Le correcteur F' est un correcteur plus petit que v et il satisfait l'objectif de commande.

$$F' = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \gamma^4 \delta^4 (\gamma^3 \delta^4)^* & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \delta^2 (\gamma^2 \delta^4)^* & \gamma^4 \delta^4 (\gamma^3 \delta^4)^* & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{pmatrix}.$$

On a bien $G \preceq (A \oplus F')^*B$, puisque

$$G = \begin{pmatrix} (\gamma^2\delta^4)^* & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & e & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \delta^2(\gamma^2\delta^4)^* & \gamma^4\delta^4(\gamma^3\delta^4)^* & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \delta^2(\gamma^2\delta^4)^* & \gamma^1\delta^1 \oplus \gamma^4\delta^4(\gamma^3\delta^4)^* & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \delta^5(\gamma^2\delta^4)^* & \gamma^1\delta^3(\gamma^3\delta^4)^* & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \gamma^1\delta^6(\gamma^2\delta^4)^* & \gamma^2\delta^4(\gamma^3\delta^4)^* & \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{pmatrix}$$

et

$$(A \oplus F')^*B = \begin{pmatrix} (\gamma^2\delta^4)^* & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & e & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \delta^2(\gamma^2\delta^4)^* & \gamma^4\delta^4(\gamma^3\delta^4)^* & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \delta^2(\gamma^2\delta^4)^* & \gamma^1\delta^1 \oplus \gamma^4\delta^4(\gamma^3\delta^4)^* & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \delta^5(\gamma^2\delta^4)^* & \gamma^1\delta^3(\gamma^3\delta^4)^* & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \gamma^1\delta^6(\gamma^2\delta^4)^* & \gamma^2\delta^4(\gamma^3\delta^4)^* & \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{pmatrix}.$$

Le transfert du système corrigé avec F' est identique à celui du système corrigé avec v , c'est-à-dire $(A \oplus v)^*B = (A \oplus F')^*B$. Le calcul de $(A \oplus v)^*B$ donne

$$(A \oplus v)^*B = \begin{pmatrix} (\gamma^2\delta^4)^* & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & e & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \delta^2(\gamma^2\delta^4)^* & \gamma^4\delta^4(\gamma^3\delta^4)^* & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \delta^2(\gamma^2\delta^4)^* & \gamma^1\delta^1 \oplus \gamma^4\delta^4(\gamma^3\delta^4)^* & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \delta^5(\gamma^2\delta^4)^* & \gamma^1\delta^3(\gamma^3\delta^4)^* & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \gamma^1\delta^6(\gamma^2\delta^4)^* & \gamma^2\delta^4(\gamma^3\delta^4)^* & \varepsilon & \varepsilon & \varepsilon & \varepsilon \end{pmatrix}.$$

Bibliographie

- [Amari et al., 2005] Amari, S., Loiseau, J. J., and Demongodin, I. (2005). Control of temporal constraints based on dioid algebra for timed event graphs. In *IPDPS '05: Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium (IPDPS'05)*, page 140.2, Denver, Colorado. IEEE Computer Society. pages 70
- [Baccelli et al., 1992] Baccelli, F., Cohen, G., Olsder, G. J., and Quadrat, J. P. (1992). *Synchronization and Linearity*. Wiley. pages 9, 19, 22, 24, 26, 34, 40, 44, 48, 51, 53, 101
- [Baclawski and Björner, 1979] Baclawski, K. and Björner, A. (1979). Fixed points in partially ordered sets. *Advances in Mathematics*, 31(3):263–287. pages 26
- [Berman and Plemmons, 1979] Berman, A. and Plemmons, R. (1979). *Nonnegative Matrices in the Mathematical Sciences*. Academic Press. pages 16
- [Birkhoff, 1940] Birkhoff, G. (1940). Lattice theory. In *American Mathematical Society Colloquium Publications*, volume xxv. Providence. pages 8
- [Blyth and Janowitz, 1972] Blyth, T. S. and Janowitz, M. F. (1972). *Residuation Theory*. Pergamon press. pages 19
- [Boimond and Ferrier, 1996] Boimond, J. L. and Ferrier, J. L. (1996). Internal Model Control and Max-Algebra: Controller Design. *IEEE Trans. on Automatic Control*, 41(3):457–461. pages 40, 64
- [Borne et al., 1990] Borne, P., Dauphin-Tanguy, G., Richard, J., Rotella, F., and Zambettakis, I. (1990). *Commande et optimisation des processus*. Technip. pages 75
- [Boudec and P., 2001] Boudec, J. Y. L. and P., T. (2001). *Network Calculus*. Springer Verlag LNCS 2050. pages 33
- [Braker, 1993] Braker, H. (1993). *Algorithms and applications in timed discrete event systems*. PhD thesis, Delft University of Technology. pages 33, 98, 101, 102, 103
- [Ceder et al., 2001] Ceder, A., Golany, B., and Tal, O. (2001). Creating bus timetables with maximal synchronization. *Transportation Research Part A: Policy and Practice*, 35:913–928. pages 104, 108

- [Cochet-Terrasson et al., 1998] Cochet-Terrasson, J., Cohen, G., Gaubert, S., Gettrick, M. M., and Quadrat, J. P. (1998). Numerical computation of spectral elements in max-plus algebra. In *Proceedings of the IFAC Conference on System Structure and Control*, Nantes. pages 17
- [Cohen, 1995] Cohen, G. (1995). *Théorie algébrique des systèmes à événements discrets*. Poly-copié de cours de l'École des Mines de Paris. pages 16
- [Cohen et al., 1983] Cohen, G., Dubois, D., Quadrat, J. P., and Viot, M. (1983). Analyse du comportement périodique des systèmes de production par la théorie des dioïdes. Rapport de recherche 191, INRIA, Le Chesnay, France. pages 17, 33
- [Cohen et al., 1989] Cohen, G., Moller, P., Quadrat, J. P., and Viot, M. (1989). Algebraic tools for the performance evaluation of discrete event systems. volume 77. pages 45, 46, 57, 60, 74, 75
- [Cottenceau, 1999] Cottenceau, B. (1999). *Contribution à la commande de systèmes à événements discrets : synthèse de correcteurs pour les graphes d'événements temporisés dans les dioïdes*. Thèse de doctorat, ISTIA - Université d'Angers. pages 53, 61
- [Cottenceau et al., 1999] Cottenceau, B., Hardouin, L., Boimond, J. L., and Ferrier, J. L. (1999). Synthesis of greatest linear feedback for TEG in dioids. *IEEE Trans. on Automatic Control*, 44(6):1258–1262. pages 63, 64
- [Cottenceau et al., 2001] Cottenceau, B., Hardouin, L., Boimond, J. L., and Ferrier, J. L. (2001). Model reference control for timed event graphs in dioids. *Automatica*, (37). pages 63
- [Cousot and Cousot, 1979] Cousot, P. and Cousot, R. (1979). Constructive versions of Tarski's fixed point theorems. *Pacific Journal of Mathematics*, 81(1):43–57. pages 24
- [Cuninghame-Green, 1979] Cuninghame-Green, R. (1979). *Minimax Algebra*. Number 166 in Lecture notes in Economics and Mathematical Systems. Springer. pages 9
- [Dacić, 1983] Dacić, R. M. (1983). On fixed edges of antitone self-mappings of complete lattices. *Publications de l'Institut Mathématique*, 34:49–53. pages 26, 29
- [Davey and Priestley, 1990] Davey, B. and Priestley, H. (1990). *Introduction to lattices and order*. Cambridge University press. pages 8
- [David and Alla, 1989] David, R. and Alla, H. (1989). *Du Grafset aux réseaux de Petri*. Hermès, Paris. pages 33
- [De Schutter and van den Boom, 2000] De Schutter, B. and van den Boom, T. (2000). Model predictive control for max-plus-linear systems. In *Proceedings of American Control Conference 2000*, pages 4046–4050, Chicago, Illinois. pages 58, 61
- [de Vries et al., 1998] de Vries, R., de Schutter, B., and de Moor, B. (1998). On max-algebraic models for transportation networks. In *Proceedings of WODES'98*, Cagliari, Italy. pages 98
- [Descamps, 2002] Descamps, P. (2002). Environnement de modélisation et de simulation de réseaux de transport urbain de passagers. Diplôme d'ingénieur du cnam, Conservatoire National des Arts et Métiers. pages 95

- [Gaubert, 1992] Gaubert, S. (1992). *Théorie des systèmes linéaires dans les dioïdes*. Thèse de doctorat, Ecole des Mines de Paris. pages 16, 17, 18
- [Gaubert, 1994a] Gaubert, S. (1994a). On rational series in one variable over certain dioids. Tech. Rep. 2162, INRIA. pages 19
- [Gaubert, 1994b] Gaubert, S. (1994b). Rational series over dioids and discrete event systems. In *Proc. of the 11th Conf. on Anal. and Opt. of Systems: Discrete Event Systems*, number 199 in Lect. Notes. in Control and Inf. Sci, Sophia Antipolis. Springer. pages 19
- [Gaubert, 1997] Gaubert, S. (1997). Methods and applications of $(\max, +)$ linear algebra. Rapport de recherche 3088, INRIA. pages 18
- [Gondran and Minoux, 1979] Gondran, M. and Minoux, M. (1979). *Graphes et algorithmes*. Eyrolles, Paris. Engl. transl. *Graphs and Algorithms*, Wiley, 1984. pages 16
- [Gondran and Minoux, 2001] Gondran, M. and Minoux, M. (2001). *Graphes, Dioïdes et semi-anneaux*. Technique et documentation, Paris. pages 9
- [Goverde et al., 1998] Goverde, R., Bovy, P., and Olsder, G. J. (1998). The max-plus algebra approach to transportation problems. In *Proceedings of WCTR*, Antwerp. pages 98, 99, 101
- [Grätzer, 2003] Grätzer, G. (2003). *General Lattice Theory*. Birkhäuser Verlag. pages 8
- [Hamaci et al., 2006] Hamaci, S., Boimond, J.-L., and Lahaye, S. (2006). On Modeling and Control of Hybrid Timed Event Graphs with Multipliers Using $(\min, +)$ Algebra. *Journal of Discrete Event Dynamic Systems*, 16:241–256. pages 61
- [Hardouin, 2004] Hardouin, L. (2004). *Sur la commande linéaire de systèmes à événements discrets dans l'algèbre $(\max, +)$* . Rapport d'habilitation à diriger des recherches, ISTIA - Université d'Angers. pages 65
- [Hardouin et al., 1997] Hardouin, L., Menguy, E., Boimond, J. L., and Ferrier, J. L. (1997). S.I.S.O. discrete event systems control in dioid algebra. *JESA*, 31(3):433–452. pages 61
- [Hayat and Maouche, 1997] Hayat, S. and Maouche, S. (1997). Régulation du trafic des autobus : amélioration de la qualité des correspondances. Technical Report LI-TU0192, INRETS. pages 94, 95, 96
- [Heidergott and Vries, 2001] Heidergott, B. and Vries, R. D. (2001). Towards a $(\max, +)$ control theory for a public transportation networks. *Journal of Discrete Event Dynamic Systems*, 11:371–398. pages 101
- [Houssin et al., 2004] Houssin, L., Lahaye, S., and Boimond, J. L. (2004). Modelling and control of urban bus networks in dioids algebra. In *Proceedings of WODES'2004*, Reims, France. pages 99
- [Houssin et al., 2005] Houssin, L., Lahaye, S., and Boimond, J. L. (2005). Commande en juste-à-temps sous contraintes de système $(\max, +)$ -linéaires. *Journal Européen des Systèmes Automatisés, Actes de MSR 2005*, 39(1-2-3):335–350. pages 73, 75

- [Houssin et al., 2006a] Houssin, L., Lahaye, S., and Boimond, J. L. (2006a). Control of constrained $(max, +)$ -linear systems minimizing delays. In *Proceedings of INCOM'2006*, Saint-Etienne, France. pages 54, 82
- [Houssin et al., 2006b] Houssin, L., Lahaye, S., and Boimond, J. L. (2006b). Timetable synthesis using $(max, +)$ algebra. In *Proceedings of INCOM'2006*, Saint-Etienne, France. pages 99
- [Houssin et al., 2007] Houssin, L., Lahaye, S., and Boimond, J. L. (2007). Just-in-time control of constrained $(max, +)$ -linear systems. *Journal of Discrete Event Dynamic Systems (à paraître)*. pages 75
- [Karp, 1978] Karp, R. M. (1978). A characterization of the minimum cycle mean in a digraph. *Discrete Math.* pages 17
- [Katz, 2005] Katz, R. D. (2005). Max-plus (A, B) -invariant spaces and control of timed discrete event systems. Rapport de Recherche 5521, INRIA. pages 104
- [Lahaye, 2000] Lahaye, S. (2000). *Contribution à l'étude des systèmes linéaires non stationnaires dans l'algèbre des dioïdes*. Thèse de doctorat, ISTIA - Université d'Angers. pages 36, 61
- [Lahaye et al., 2004] Lahaye, S., Cottenceau, B., and Corréia, A. (2004). Commande de graphes d'événements temporisés avec contrainte de temps critique. In *Proceedings of CIFA'2004*, Douz, Tunisie. pages 70
- [Lahaye et al., 2003] Lahaye, S., Houssin, L., and Boimond, J. L. (2003). Modelling of urban bus networks in dioids algebra. In *Proceedings of POSTA 2003*, pages 23–30, Roma, Italy. Springer Verlag. pages 98
- [Lassez et al., 1982] Lassez, J.-L., Nguyen, V. L., and Sonenberg, L. (1982). Fixed point theorems and semantics: A folk tale. *Inf. Process. Lett.*, 14(3):112–116. pages 23
- [Lhommeau, 2003] Lhommeau, M. (2003). *Etude 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 de doctorat, ISTIA - Université d'Angers. pages 61, 63, 65
- [Lhommeau et al., 2003] Lhommeau, M., Hardouin, L., and Cottenceau, B. (2003). Optimal control for $(max, +)$ -linear systems in the presence of disturbances. In *Proceedings of POSTA 2003*, Roma, Italy. Springer Verlag. pages 63
- [Libeaut and Loiseau, 1996] Libeaut, L. and Loiseau, J. J. (1996). On the control of timed event graphs. In *Proceedings of WODES'96*, Edimburgh. pages 61
- [Lin et al., 1995] Lin, G.-S., Liang, P., Schonfeld, P., and Larson, R. (1995). Adaptive control of transit operations. Rapport de recherche MD-26-7002, University of Maryland. pages 100, 101

- [Lotito et al., 2005] Lotito, P., Mancinelli, E., and Quadrat, J.-P. (2005). A minplus derivation of the fundamental car-traffic law. *IEEE Trans. on Automatic Control*, 50(5):699–705. pages 33
- [Lüders and Santos-Mendes, 2002] Lüders, R. and Santos-Mendes, R. (2002). Generalized multivariable control of discrete event systems in dioid. In *Proceedings of WODES 2002*, Zaragoza, Spain. pages 63, 64
- [Maia et al., 2003] Maia, C., Hardouin, L., Mendes, R. S., and Cottenceau, B. (2003). Optimal closed-loop control for timed event graphs in dioid. *IEEE Trans. on Automatic Control*, 48:2284–2287. pages 64
- [Menguy et al., 2000a] Menguy, E., Boimond, J. L., Hardouin, L., and Ferrier, J. L. (2000a). A first step towards adaptative control for linear systems in max algebra. *J. of Discrete Event Dynamic Systems*, 10(4):347–367. pages 65
- [Menguy et al., 2000b] Menguy, E., Boimond, J. L., Hardouin, L., and Ferrier, J. L. (2000b). Just in time control of linear systems in dioid: cases of an update of reference input and uncontrollable input. *IEEE Trans. on Automatic Control*, 45(11):2155–2159. pages 61
- [Murata, 1989] Murata, T. (1989). Petri Nets : Properties, Analysis and Applications. *IEEE Proceedings: Special issue on Discrete Event Systems*, 77:541–580. pages 33
- [Nait-Sidi-Moh, 2003] Nait-Sidi-Moh, A. (2003). *Contribution à la modélisation, à l'analyse et à la commande des systèmes de transport public par les réseaux de Petri et l'algèbre (Max,plus)*. Phd thesis, UTBM - Université de Franche-Comté. pages 94, 98, 99
- [Palma and Lindsey, 2000] Palma, A. D. and Lindsey, R. (2000). Optimal timetables for public transportation. *Transportation Research*, B35:789–813. pages 104
- [Soulhi, 2000] Soulhi, A. (2000). *Contribution de l'intelligence artificielle à l'aide à la décision dans la gestion des systèmes de transport urbain collectif*. Ph. d. thesis, Université des sciences et technologies de Lille. pages 96
- [Strathman et al., 2001] Strathman, J., Kimpel, T., and Dueker, K. (2001). Bus transit operations control. *Journal of Public Transportation*, 31(1):1–26. pages 100, 101
- [SWtool, 2005a] SWtool (2005a).
<ftp://ftp.inria.fr/inria/scilab/contrib/maxplus>. pages 39
- [SWtool, 2005b] SWtool (2005b).
www.istia.univ-angers.fr/~hardouin/outils.html. pages 55, 111
- [Tarski, 1955] Tarski, A. (1955). A lattice theoretical fixed point theorem and its applications. *Pacific Journal of Mathematics*, 5(2):285–309. pages 23
- [van den Boom and De Schutter, 2001] van den Boom, T. and De Schutter, B. (2001). Mpc for max-plus-linear systems: Closed loop behavior and tuning. In *Proceedings of American Control Conference 2001*, pages 325–330, Arlington, Virginia. pages 61

- [Zad et al., 1999] Zad, S. H., Kwong, R. H., and Wonham, W. M. (1999). Supremum operators and computation of supremal elements in system theory. *SIAM Journal on Control and Optimization*, 37(3):695–709. pages 24

Résumé

Les systèmes à événements discrets qui mettent en jeu des phénomènes de synchronisation possèdent une description linéaire dans l'algèbre $(max, +)$. Une théorie des systèmes linéaires dans les dioïdes (notamment $(max, +)$), développée par analogie à la théorie conventionnelle des systèmes, a alors vu le jour.

Dans ce mémoire, nous abordons des problèmes de commande des systèmes $(max, +)$ -linéaires. Jusque-là, les commandes de systèmes linéaires dans les dioïdes ont principalement considéré le critère du juste-à-temps et des objectifs de commande comme la poursuite d'une trajectoire de référence ou la poursuite d'un transfert de référence. La première contribution à la commande réside dans la prise en compte d'un nouvel objectif pour la commande en juste-à-temps. Plus précisément, on considère des spécifications sous la forme de contraintes sur l'état du système.

Ensuite, nous abordons un nouveau critère de commande. On s'intéresse à la synthèse d'un correcteur qui ralentit le moins possible le transfert du système tout en assurant un objectif de commande défini par un ensemble de contraintes sur l'état. Plusieurs structures de commande sont présentées dans cette optique.

Les réseaux de transport admettent une description linéaire dans l'algèbre $(max, +)$. La dernière partie de ce mémoire est consacrée à leur étude. Plus précisément, on propose des outils pour l'évaluation de performances et la commande de ces systèmes. Le problème de synthèse de tables d'horaires est notamment formulé en un problème de commande de systèmes $(max, +)$ -linéaires.

Mots Clés : systèmes à événements discrets, commande, algèbre $(max, +)$, réseaux de transport.

Abstract

The functioning of Discrete event dynamic systems subject to synchronization and delay phenomena can be described by linear models in a particular algebraic structure called $(max, +)$ algebra. A linear system theory has been developed over dioids (such as $(max, +)$) by analogy with the conventional one.

This report is devoted to control problem of $(max, +)$ -linear systems. Some control problems for these systems have previously been tackled. In all these works, the problematic was to compute an optimal solution in regards of the just-in-time criterion while considering control objective such as reference model or reference output. For our concern, the first contribution to the control aim at delaying as less as possible the controlled system while ensuring some given specifications. More precisely, we consider specifications which fit on system state inequalities.

Then, we study a new control criterion. We are interested in controller synthesis which delay as less as possible the system while ensuring a control objective defined by constraints on system state.

Transportation networks can be described by a linear representation in $(max, +)$ -algebra. The last part of this report is devoted to their study. We propose some tools for the evaluation performance and the control of these systems. Notably, the timetables synthesis problem is formulated and solved as $(max, +)$ -linear system control.

Mots Clés : discrete event dynamic systems, control, $(max, +)$ algebra, transportation networks.