



Laboratoire d'Ingénierie des Systèmes Automatisés  
FRE 2656, Angers

Diplôme d'Etudes Approfondies en Automatique et Informatique  
Appliquée

RAPPORT DE RECHERCHE

---

PROPRIÉTÉS SPECTRALES DES FONCTIONS (Min, Max, +)

---

CALCUL D'ELEMENTS SPECTRAUX

---

BESSEM BOURAOUI

---

Responsable de stage : Philippe Declerck, maître de conférences

# Remerciements

J'adresse mes sincères remerciements à Monsieur Jean-Louis Ferrier, pour la confiance qu'il m'a accordée en m'accueillant au sein de son laboratoire.

Je remercie également Monsieur Philippe Declerck, pour la qualité de son encadrement, et son soutien dans la réalisation de ce travail.

Je tiens aussi à exprimer ma profonde gratitude et mes remerciements les plus sincères à tous les membres du LISA, et en particulier les doctorants.

Enfin, je remercie tous ceux qui n'ont pas été cités ci-dessus et qui pourtant ont eu un rôle à jouer dans cette belle année d'études.

A ma mère qui sans elle, ce travail n'aurait jamais vu le jour.

# Table des matières

<b>1</b>	<b>OUTILS ALGÈBRIQUES</b>	<b>8</b>
1.1	Structures ordonnées et treillis: . . . . .	8
1.1.1	Structures ordonnées: . . . . .	8
1.1.2	Demi-treillis et treillis: . . . . .	9
1.2	Monoïdes et dioïdes: . . . . .	11
1.2.1	Sous-dioïde: . . . . .	12
1.2.2	Calcul matriciel dans les dioïdes: . . . . .	12
1.2.3	Dioïdes et structures ordonnées: . . . . .	12
1.2.4	Conclusion: . . . . .	14
<b>2</b>	<b>COMPORTEMENT LINÉAIRE DES GRAPHES D'ÉVÉNEMENTS DISCRETS DANS LES DIOÏDES</b>	<b>15</b>
2.1	Introduction: . . . . .	15
2.2	Modélisation de systèmes à événement discrets par réseaux de pétri: . . . . .	15
2.2.1	Les réseaux de Petri: . . . . .	15
2.2.2	Modélisation des systèmes à événements discrets: . . . . .	21
2.2.3	Quelques propriétés des RdP: . . . . .	21
2.3	Les graphes d'événements temporisés: . . . . .	22
2.3.1	Propriétés des graphes d'événements: . . . . .	22
2.3.2	Les graphes d'événements P-temporisés: . . . . .	23
2.3.3	Prise en compte des contraintes temporelles: . . . . .	23
2.4	Validation d'un modèle: . . . . .	27
2.4.1	Validation par analyse: . . . . .	28
2.5	Conclusion: . . . . .	29
<b>3</b>	<b>THÉORIE SPECTRALE ET POINT FIXE</b>	<b>30</b>
3.1	Introduction: . . . . .	30
3.2	Fonction Min Max: . . . . .	32
3.2.1	Fonctions min max et Fonctions topicales: . . . . .	34
3.2.2	Fonctions topicales et temps de cycle: . . . . .	35
3.2.3	Fonctions max-plus, temps de cycle et projecteur spectral: . . . . .	36
3.3	Point fixe de fonctions (min, max, +): . . . . .	39
3.3.1	Rectangularité: . . . . .	39
3.3.2	Théorème de point fixe: . . . . .	41
3.4	Problèmes Algorithmiques: . . . . .	43
3.4.1	Technique de construction de point fixe: . . . . .	43
3.4.2	Algorithme 3.1: . . . . .	44

3.4.3	Script Scilab: . . . . .	44
3.4.4	Exemple d'application . . . . .	47
3.5	Exemple de généralisation: . . . . .	49
3.5.1	Algorithme de calcul d'mode propre généralisé: . . . . .	49
3.5.2	Algorithme d'amélioration de politique: . . . . .	50
3.5.3	Algorithme général: . . . . .	50
3.5.4	Exemple d'application: . . . . .	51
3.6	Conclusion: . . . . .	52

# INTRODUCTION

Le travail présenté dans ce mémoire a pour point de départ la théorie des systèmes  $(\max, +)$  linéaires également appelée théorie des systèmes linéaires dans les dioïdes. Cette théorie concerne la sous-classe des systèmes à événements discrets temporisés dont l'évolution de l'état est représentable par des équations récurrentes linéaires sur des structures algébriques particulières telles que l'algèbre  $(\max, +)$  ou l'algèbre  $(\min, +)$ . Le comportement des systèmes caractérisés par des délais et des synchronisations peut être décrit par de telles récurrences. Ces systèmes correspondent en fait aux Graphes d'Événements Temporisés (GET) qui constituent une sous-classe des Réseaux de Petri (RdP) dont chaque place n'admet qu'une transition en amont et une transition en aval.

Dans le cadre de cette théorie, une fonction  $(\min, \max, +)$  est établie, par application finie de plusieurs opérations  $\max$  et  $\min$  pour chaque composante. De telles fonctions sont homogènes, monotones, et non-expansives. Nous suivrons Gunawardena et Kaene et nous les appellerons des fonctions topicales [2, 23, 28, 31, 40]. Les fonctions topicales incluent (probablement après transformation appropriée) les matrices non négatives, les systèmes de substitution de Leontieff, les opérateurs de programmation dynamiques des jeux et des procédés de décision de Markov, les opérateurs non linéaires surgissant dans des problèmes de graduation de matrices... [12, 24]. Elles incluent également des exemples, tels que les fonctions  $(\min, \max, +)$ , qui résultent de la modélisation des systèmes à événements discrets, de circuits numériques, des réseaux informatiques...

N'importe quelle fonction topicale  $T$ , peut être rapprochée par les fonctions  $(\min, \max, +)$  de telle manière qu'un certain comportement dynamique de  $T$  soit hérité par des approximations (voir le lemme 3,1). Nous étudierons la dynamique des fonctions  $(\min, \max, +)$  motivés en partie par les applications aux systèmes à événements discrets.

Pour n'importe quelle fonction topicale, le vecteur de temps de cycle fournit la généralisation non-linéaire appropriée de la racine de Perron, ou le rayon spectral (voir le P. 3.3). La limite en question n'existe pas toujours, [23, théorème 3,1], et ceci constitue un problème important non résolu pour caractériser ces fonctions topicales. Le vecteur de temps de cycle fournit une mesure de performance dans les systèmes à événements discrets (voir le lemme 3.2). Dans le contexte des systèmes à événements discrets ceci représente le fait que le système peut seulement posséder un état d'équilibre quand le temps de cycle de chaque événement est asymptotiquement identique. Il est intéressant de se demander si l'inverse est vrai.

A partir de cette théorie, et particulièrement d'un théorème constructif de point fixe des fonctions  $(\min, \max, +)$ , nous présentons et développons, sous scilab, l'algorithme de construction de point fixe, un algorithme basé sur la preuve d'existence, qui n'est pas général mais peut être rendu suffisant dans un cas particulier important, où le graphe d'événements discrets est fortement connexe. Nous donnons un exemple d'application pour ce programme.

Les méthodes de cette technique s'appliquent sur une classe particulière des fonctions  $(\min, \max, +)$  qui peuvent être étudiées par des méthodes linéaires. L'intérêt pour la classe des fonctions  $(\min, \max, +)$ , et la classe plus large des fonctions topicales, est venu d'un certain nombre de directions et a rassemblés un certain nombre de thèmes distincts tels que les systèmes à événement discrets.

# Chapitre 1

## OUTILS ALGÈBRIQUES

Ce chapitre présente les outils algébriques utilisés dans le cadre de ce mémoire.

### 1.1 Structures ordonnées et treillis :

Cette section présente les grandes lignes de la théorie des treillis. Les treillis sont des concepts mathématiques que l'on peut manipuler en tant qu'ensembles ordonnés ou en tant que structures algébriques. Ces concepts sont bien étudiés dans [a] et [b].

#### 1.1.1 Structures ordonnées :

**Définition 1 (*Ensemble ordonné*):** Un ensemble ordonné est un ensemble  $S$  muni d'une relation d'ordre, c'est-à-dire une relation binaire qui est réflexive, antisymétrique et transitive. Cette relation sera notée  $\preceq$  et un ensemble ordonné sera noté  $(S, \preceq)$ .

Un ensemble est dit totalement ordonné si deux éléments quelconques  $s$  et  $s'$  sont toujours comparables, c'est-à-dire si l'on a  $s \preceq s'$  ou  $s' \preceq s$ . La notation  $s \preceq s'$  signifie  $s \preceq s'$  et  $s \neq s'$ .

**Remarque 1** En présence d'ambiguïtés sur l'ensemble considéré, nous désignerons par la notation  $\preceq_S$  l'ordre d'un ensemble  $S$ .

Tout sous-ensemble  $U$  d'un ensemble ordonné  $(S, \preceq)$  peut également être ordonné par la restriction de l'ordre de  $S$  aux éléments de  $U$ , notée  $\preceq_U$ . Cet ordre restreint est simplement défini par :

$$u, u' \in U \subset S, \quad u \preceq u' \Leftrightarrow u \preceq_U u'$$

**Remarque 2** Si  $(S, \preceq)$  est partiellement ordonné, un sous-ensemble  $U \subset S$ , ordonné par la restriction de  $\preceq$  à  $U$ , peut avoir tout ses éléments incomparables deux à deux. L'ensemble  $(U, \preceq)$  est alors dit totalement non ordonné.

Un ensemble ordonné fini  $(S, \preceq)$  peut être représenté par un graphe appelé diagramme de Hasse. Chaque élément de  $S$  est représenté par un sommet ( $\bullet$ ). Un arc reliant deux sommets du diagramme signifie que les éléments représentés par ces sommets sont comparables.

Pour la figure 1.1, l'ensemble  $S = \{a, b, c, d\}$  est partiellement ordonné pour l'ordre  $\preceq$  décrit par le diagramme. Le sous-ensemble  $U = \{a, b\} \subset S$  est un ensemble ordonné par la restriction



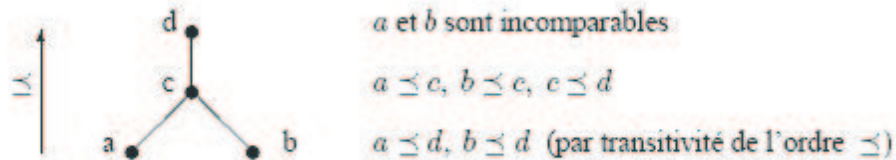


FIG. 1.1 – Diagramme de Hasse d'un ensemble ordonné  $(a, b, c, d, \preceq)$

de  $\preceq$  à  $U$ .

Néanmoins, dans ce cas précis,  $(U, \preceq)$  est totalement non ordonné (remarque 1.3).

**Remarque 3** Un ensemble totalement ordonné est également appelé une chaîne en référence à son diagramme de Hasse qui en est une.

### Exemple 1 (Ensembles ordonnés)

- ◇  $(\mathbb{R}, \leq)$ ,  $(\mathbb{Z}, \leq)$ ,  $(\mathbb{N}, \leq)$ ,  $(\mathbb{Q}, \leq)$  où  $\leq$  est l'ordre naturel, sont totalement ordonnés.
- ◇ Soit  $S$  un ensemble. L'ensemble des parties de  $S$ , noté  $P(S)$ , est un ensemble ordonné par l'inclusion. Cet ensemble ordonné est noté  $(P(S), \subset)$ . Il s'agit d'un ordre partiel. Par exemple, deux sous-ensembles disjoints de  $S$  ne sont pas comparables suivant l'ordre  $\subset$ .

**Remarque 4** On dira que deux ensembles ordonnés sont isomorphes si leurs diagrammes de Hasse ont la même forme.

**Définition 2 (Majorant, minorant)** Soit  $S$  un ensemble muni d'une relation d'ordre  $\preceq$  et  $U$  un sous-ensemble de  $S$ .

On appelle minorant de  $U$  tout élément  $s$  de  $S$  tel que  $\forall u \in U, s \preceq u$ .

On appelle majorant de  $U$  tout élément  $\forall u \in U, u \preceq s$ .

**Définition 3 (Bornes d'un ensemble)** Un sous-ensemble  $U \subset S$  est dit borné s'il admet un majorant et un minorant. Lorsque l'ensemble des majorants de  $U$  a un plus petit élément. Ce plus petit élément est appelé borne supérieure de  $U$ . On le note  $\sup(U)$  ou  $\bigvee U$ . De même lorsque l'ensemble des minorants de  $U$  a un plus grand élément, on l'appelle borne inférieure de  $U$  (notée  $\inf(U)$  ou  $\bigwedge U$ ).

**Exemple 2** Soit l'ensemble  $S = \{a, c, d, e, f, g, h, j\}$  partiellement ordonné pour l'ordre  $\preceq$  décrit par le diagramme de la figure 1.2. Soit le sous-ensemble  $U = \{a, d, h, j\}$ . L'ensemble des majorants  $s' = \{f, j\}$  et l'ensemble des minorants  $s = \{c, d, g\}$  de  $U$  sont hachurés. Remarquons que  $\sup(U) = j$  et  $\inf(U) = d$ . Notons qu'un minorant (resp. majorant) de  $U$  n'appartient pas forcément à  $U$ .

## 1.1.2 Demi-treillis et treillis :

**Définition 4 (Demi-treillis) :** Un sup-demi-treillis est un ensemble  $S$  ordonné, dans lequel tout couple d'éléments  $(s, s')$  admet une borne supérieure (plus petit majorant) notée  $\sup(s, s')$  ou  $s \vee s'$ . De même, un inf-demi-treillis est un ensemble  $S$  ordonné, dans lequel tout couple d'éléments  $(s, s')$  admet une borne inférieure notée  $\inf(s, s')$  ou  $s \wedge s'$ .

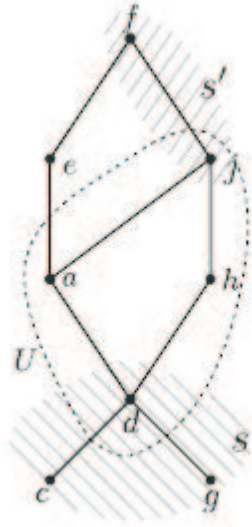


FIG. 1.2 – Représentation de l'ensemble des majorants (resp. minorants) du sous-ensemble  $U \subset S$ .

**Remarque 5 (Principe de dualité).** Notons  $\preceq_{op}$  l'inverse de la relation d'ordre  $\preceq$ . Si  $(S, \preceq)$  est un sup-demi-treillis, alors  $(S, \preceq_{op})$  est un inf-demi-treillis, et vice versa. Par conséquent, une relation impliquant  $\preceq$ ,  $\vee$  et  $\wedge$  reste vraie en remplaçant  $\preceq$  par  $\preceq_{op}$  et en permutant  $\vee$  et  $\wedge$ . Il s'agit du principe de dualité.

**Définition 5 (Treillis).** Un treillis est un ensemble ordonné  $(S, \preceq)$  qui est à la fois un sup-demi-treillis et un inf-demi-treillis; autrement dit, un treillis est un ensemble ordonné dans lequel tout couple d'éléments admet un plus petit majorant et un plus grand minorant.

**Exemple 3**  $(\mathbb{N}^*, \preceq_{div})$ , où l'ordre sur  $\mathbb{N} \times \mathbb{N}^*$  est défini par

$$a \preceq_{div} b \Leftrightarrow a \text{ divise } b, \quad (1.1)$$

est un treillis. Les lois de treillis de  $(\mathbb{N}^*, \preceq_{div})$  sont  $a \vee b = \text{ppcm}(a, b)$  et  $a \wedge b = \text{pgcd}(a, b)$ .

**Définition 6 (Sous-treillis).** On appelle sous-treillis d'un treillis  $S$ , un sous-ensemble  $U$  de  $S$  qui, avec chaque couple  $u, u'$  de  $U$  contient aussi  $u \vee u'$  et  $u \wedge u'$  dans  $U$ . On dit aussi que ce sous-ensemble est fermé pour les lois  $\vee$  et  $\wedge$ .

**Remarque 6** Il faut noter qu'un sous-ensemble  $U$  peut être un treillis sans être un sous-treillis de  $S$ .

**Exemple 4** Considérons le diagramme de Hasse d'un treillis fini  $S$  contenant 8 éléments (figure 1.3). On remarque que les ensembles  $\{h, d, e, b\}$ ,  $\{h, d, f, a\}$  sont des sous-treillis, il en est de même pour l'ensemble  $\{h, f, c, g\}$  qui en outre est une chaîne.

Par contre, l'ensemble  $\{h, d, e, a\}$  n'est pas un sous-treillis, car  $d \vee e$  (c-à-d.  $b$ ) ne fait pas partie de l'ensemble.

Néanmoins on voit que  $\{h, d, e, a\}$  est un treillis (pour la relation d'ordre existant entre  $a, d, e$  et  $h$  dans  $S$ ) mais pas un sous-treillis de  $S$  puisque dans  $\{a, d, e, h\}$ ,  $d \vee e = a$  alors que, dans  $S$ ,  $d \vee e = b$ .

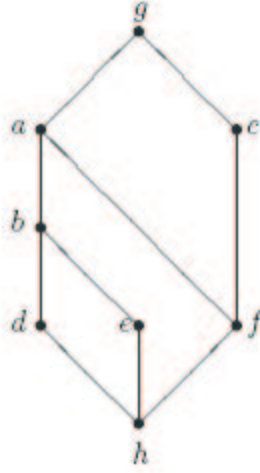


FIG. 1.3 – Diagramme de Hasse du Treillis  $(S, \preceq)$ .

**Définition 7 (Demi-treillis complet et treillis complet).** Un sup-demi-treillis (resp. inf-demi-treillis)  $S$  est dit sup-complet (resp. inf-complet) si tout sous-ensemble (fini ou infini) de  $S$  admet un plus petit majorant (resp. un plus grand minorant) dans  $S$ . Un treillis est dit complet s'il est à la fois inf-complet et sup-complet.

**Exemple 5**  $\diamond$  En ajoutant l'élément  $+\infty$  à  $Z$ , l'ensemble  $(Z \cup +\infty, \preceq)$  est totalement ordonné sup-complet.

$\diamond$  En revanche,  $(Q \cup +\infty, \preceq)$  est un ensemble totalement ordonné qui n'est ni sup-complet ni inf-complet. Par exemple, le sous-ensemble  $\{x \in Q \mid x \leq \sqrt{2}\}$  de  $Q$  n'a pas de plus petit majorant dans  $Q$ .

**Théorème 1** Un sup-demi treillis complet  $S$  est un treillis complet si, et seulement si, il a un plus petit élément  $\varepsilon_S$ .

## 1.2 Monoïdes et diïdes :

Cette section est un rappel sur les structures algébriques considérées. Le but de ce paragraphe est d'introduire les concepts et notations qui seront utiles pour l'étude de la commande de graphes d'événements temporisés. Les ouvrages de référence qui ont servi à l'étude de cette partie sont : [h] à travers [b], [c], [d], [e], [f] et [g].

**Définition 8 (Monoïde) :** Un monoïde est un ensemble  $C$  muni d'une loi de composition interne associative, et d'un élément neutre pour cette loi. Nous noterons la loi additivement

$$C \times C \rightarrow C(a,b) \mapsto a \oplus b$$

et l'élément neutre sera noté  $\varepsilon$ . Le monoïde est dit commutatif si l'on a

$$a \oplus b = b \oplus a \forall a, b \in C$$

**Remarque 7** Un élément  $a \in C$  est idempotent si  $a \oplus a = a$ . Si tous les éléments de  $C$  sont idempotents, le monoïde  $(C, \oplus)$  est dit idempotent.

**Exemple 6** L'ensemble des entiers naturels  $N$  muni de l'addition est un monoïde. Il s'agit d'un monoïde commutatif:  $\forall a, b \in N, a + b = b + a$ . L'élément neutre de  $N$  est  $0$ .

**Définition 9 (Demi-anneau, Dioïde).** On appelle demi-anneau un ensemble  $D$  muni de deux lois internes  $\oplus$  et  $\otimes$  tel que :

- ◊  $(D, \oplus)$  est un monoïde commutatif dont l'élément neutre  $\varepsilon$  est appelé élément nul.
- ◊  $(D, \otimes)$  est un monoïde. Son élément neutre est appelé unité et est noté  $e$ .
- ◊ La loi multiplicative  $\otimes$  est distributive à droite et à gauche par rapport à la loi additive  $\oplus$ .
- ◊ L'élément neutre  $\varepsilon$  est absorbant pour la loi  $\otimes$  ( $\forall a \in D, a \otimes \varepsilon = \varepsilon \otimes a = \varepsilon$ ).

Si en outre la loi  $\oplus$  est idempotente ( remarque 7), alors  $(D, \oplus, \otimes)$  est qualifié de demi-anneau idempotent ou dioïde.

**Exemple 7 (Algèbres  $(max, +)$  et  $(min, +)$ ).** On peut vérifier aisément que :

◊  $(\mathbb{R} \cup \{-\infty\}, max, +)$  est un dioïde commutatif pour lequel  $\varepsilon = -\infty$ , et  $e = 0$ . Ce dioïde est noté  $\mathbb{R}_{max}$  et est traditionnellement appelé "algèbre  $(max, +)$ ".

◊  $\mathbb{R}_{min} = (\mathbb{R} \cup \{+\infty\}, min, +)$  est un dioïde commutatif, appelé "algèbre  $(min, +)$ ", pour lequel  $\varepsilon = +\infty$  et  $e = 0$ .

### 1.2.1 Sous-dioïde :

**Définition 10 (Sous-dioïde).**

Soit  $(D, \oplus, \otimes)$  un dioïde. Un sous-ensemble  $C \subset D$  est un sous-dioïde de  $(D, \oplus, \otimes)$  si et seulement si :

- ◊  $\varepsilon \in C$  et  $e \in C$  ;
- ◊  $C$  est fermé pour les lois  $\oplus$  et  $\otimes$ .

Le second point signifie que  $\forall a, b \in C, a \oplus b \in C$  et  $a \otimes b \in C$ .

**Exemple 8**  $Z_{max} = (Z \cup \{-\infty\}, max, +)$  et  $Z_{min} = (Z \cup \{+\infty\}, min, +)$  sont respectivement des sous-dioïdes de  $\mathbb{R}_{max}$  et  $\mathbb{R}_{min}$ .

### 1.2.2 Calcul matriciel dans les dioïdes :

**Exemple 9 (Dioïde matriciel).** Soit  $(D, \oplus, \otimes)$  un dioïde, on note  $D_{n \times n}$  l'ensemble des matrices carrées à  $n$  lignes et  $n$  colonnes à coefficients dans  $D$ . La somme et le produit de matrices sont définis de façon classique par :

$$(A \oplus B)_{ij} = A_{ij} \oplus B_{ij} \quad \text{et} \quad (A \otimes B)_{ij} = \bigoplus_{1 \leq k \leq n} A_{ik} \otimes B_{kj}$$

$$\forall i, j = 1, \dots, n; \quad \forall A, B \in D^{n \times n}$$

L'ensemble  $D^{n \times n}$  muni de ces deux opérations est un dioïde. L'élément neutre pour la loi  $\oplus$  est la matrice dont tous les coefficients valent  $\varepsilon$ , laquelle sera aussi notée  $\varepsilon$ . L'élément neutre pour la loi  $\otimes$  est la matrice dont tous les coefficients valent  $\varepsilon$ , sauf ceux de la diagonale qui valent  $e$ . Cette matrice sera notée  $e$ , ou  $E$  lorsqu'il est utile de la distinguer du scalaire  $e$ .

### 1.2.3 Dioïdes et structures ordonnées :

L'objet de ce paragraphe est d'établir un lien entre les dioïdes présentés précédemment de façon purement combinatoire et la théorie des treillis.

## Dioïdes canoniquement ordonnés

L'idempotence de la loi additive  $\oplus$  permet de définir naturellement une relation d'ordre dans un dioïde. Le théorème suivant affirme de plus que cette relation d'ordre est compatible avec les lois du dioïde.

**Théorème 2** Dans un dioïde  $(D, \oplus, \otimes)$ , la relation  $\preceq$  définie par

$$a \preceq b \Leftrightarrow a \oplus b = b$$

est une relation d'ordre. De plus cette relation d'ordre est compatible avec les lois de structure de  $D$ , c'est-à-dire,

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

La relation d'ordre est dite totale si

$$\forall a, b \in D, \quad a \preceq b \quad \text{ou} \quad b \preceq a.$$

Une condition nécessaire et suffisante pour que l'ordre d'un dioïde soit total, s'écrit de façon évidente

$$\forall a, b \in D, \quad a \oplus b = b \quad \text{ou} \quad a.$$

C'est-à-dire que l'opération  $\oplus$  vérifie la propriété dite de sélectivité ; dans ce cas le monoïde  $(D, \oplus)$  et le dioïde  $(D, \oplus, \otimes)$  sont dits sélectifs [i].

**Remarque 8** L'ordre  $\preceq$  défini dans  $\mathbb{R}_{max}$  est total et coïncide avec l'ordre usuel  $\leq$ . En revanche, l'ordre total  $\preceq$  défini dans  $\mathbb{R}_{min}$  est l'inverse de l'ordre usuel  $\leq$ . (par exemple  $2 \preceq 1$ ).

## Dioïdes et treillis

Dans cette partie nous présentons les relations entre dioïdes, sup-demi-treillis et treillis. Le théorème 2 de la partie précédente permet d'établir que l'idempotence de la somme dans un dioïde induit une structure de sup-demi-treillis (définition 7), pour lequel la borne supérieure, notée  $\vee$ , correspond à la loi additive  $\oplus$  ( $a \oplus b$  est le plus petit majorant de  $a$  et  $b$ ) du dioïde. De même, on peut noter que la loi  $\vee$  d'un sup-demi-treillis est associative, commutative et idempotente, c'est-à-dire que  $\vee$  possède les mêmes axiomes que la loi additive  $\oplus$  d'un dioïde. De plus, on sait, d'après la définition 9, qu'un dioïde possède un élément minimum " (plus petit que tous les autres éléments du dioïde), ce qui confère au dioïde une structure de treillis (voir théorème 1).

## Dioïde complet

**Définition 11 (Dioïde complet).** Un dioïde  $(D, \oplus, \otimes)$  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  $b \in D$  et tout sous-ensemble  $A \subset D$ , on a :

$$b \otimes \left( \bigoplus_{a \in A} a \right) = \bigoplus_{a \in A} (b \otimes a) \quad \text{et} \quad \left( \bigoplus_{a \in A} a \right) \otimes b = \bigoplus_{a \in A} (a \otimes b).$$

Il résulte de cette définition que, pour tout  $A \subset D$  et  $B \subset D$ :

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

**Exemple 10** Le dioïde  $Z_{max}$  qui est complété par l'élément  $+\infty$  est un dioïde complet noté  $\bar{Z}_{max} = (Z \cup \{-\infty, +\infty\}, max, +)$ . De même  $\bar{R}_{max} = (R \cup \{-\infty, +\infty\}, max, +)$  est le dioïde  $R_{max}$  complété par l'élément  $+\infty$ .

Puisque un dioïde  $D$  a une structure de treillis  $(D, \preceq)$ , s'il est complet, il admet un plus grand élément. On notera  $\top$  ce plus grand élément. L'élément  $\top$  correspond à la somme de tous les éléments de  $D$ :

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

Cet élément est absorbant pour la loi additive  $\forall a, \top \oplus a = \top$ , et vérifie  $\top \otimes \varepsilon = \varepsilon \otimes \top = \varepsilon$ .

**Définition 12 (Borne inf).** Si  $D$  est un dioïde complet, alors  $\wedge$  est la loi associative commutative et idempotente vérifiant

$$a \wedge b = \bigoplus \{x \mid x \preceq a \text{ et } x \preceq b\},$$

et faisant de  $(D, \oplus, \otimes)$  un treillis complet. Cette loi  $\wedge$  vérifie en conséquence

$$a = a \oplus b \Leftrightarrow a \succeq b \Leftrightarrow b = a \wedge b.$$

#### 1.2.4 Conclusion :

Ce chapitre introductif a permis de dresser la liste des outils mathématiques qui seront utilisés par la suite.

# Chapitre 2

## COMPORTEMENT LINÉAIRE DES GRAPHES D'ÉVÉNEMENTS DISCRETS DANS LES DIOÏDES

### 2.1 Introduction :

L'étude des systèmes à événements discrets (SED) constitue, depuis bientôt 30 ans, un domaine de recherche très actif (de par son intérêt théorique et économique) ayant donné lieu à de nombreuses publications. De cette littérature se dégagent de multiples classes de systèmes mettant en jeu des phénomènes de natures différentes : parallélisme, saturation, synchronisation, exclusion mutuelle, choix, séquençement ..., et autant de modèles mathématiques. Dans ce rapport nous considérons les systèmes qui admettent un modèle linéaire dans les structures algébriques introduites dans le chapitre précédent. Les ouvrages de référence qui pour cette partie sont : [j], et [k] et la référence qui a servi à la rédaction de cette partie est : [h].

### 2.2 Modélisation de systèmes à événement discrets par réseaux de pétri :

#### 2.2.1 Les réseaux de Petri :

Les réseaux de Petri (RdP) ont été introduits par C.A. Petri en 1962. Ils constituent des modèles, à la fois graphiques et mathématiques, dédiés à la représentation du comportement des systèmes dynamiques à événements discrets. En raison de leur puissance de modélisation, ces modèles ont fait l'objet de très nombreux travaux de recherche ces quarante dernières années. Un bon comportement du système dépend donc de l'ordre d'obtention des résultats mais aussi du temps pris pour les obtenir. Il apparaît alors primordial de disposer de méthodes formelles permettant l'analyse de certaines propriétés comportementales afin de préserver le bon fonctionnement du système dans son environnement. Comme les contraintes temporelles apparaissant sur ces systèmes s'expriment par des intervalles de temps, notre choix s'est naturellement orienté vers les réseaux de Petri P-temporels pour lesquels les temporisations sont associés aux places.

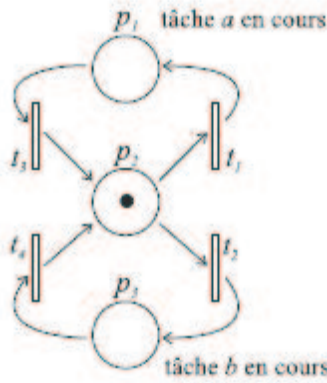


FIG. 2.1 – RdP modélisant le comportement d'un système à ressource partagée.

## Définitions et notations

**Définition 13 (Réseaux de Petri)** Un réseau de Petri est un quadruplet  $R = (P, T, Pré, Post)$ , où :

- ♠  $P$  est un ensemble non vide et fini dont les éléments sont appelés places.
- ♠  $T$  est un ensemble non vide et fini dont les éléments sont appelés transitions.
- ♠  $Pré: P \times T \rightarrow N$  une application d'incidence avant :  $Pré(p_i, t)$  contient la valeur entière  $n$  associée à l'arc allant de  $p_i$  à  $t$ .
- ♠  $Post: P \times T \rightarrow N$  une application d'incidence arrière :  $Post(p_i, t)$  contient la valeur entière  $w$  associée à l'arc allant de  $t$  à  $p_i$ .

**Exemple 11** Le RdP de la figure 2.1 représente un système, par exemple informatique, où deux tâches requièrent une même ressource renouvelable.

Les places sont ici  $P = \{p_1, p_2, p_3\}$  et les transitions  $T = \{t_1, t_2, t_3, t_4\}$ . On a :

$$\begin{aligned} Pré(p_1, t_3) &= 1 & Post(p_1, t_1) &= 1 \\ Pré(p_2, t_1) &= 1 & Post(p_3, t_2) &= 1 \\ Pré(p_2, t_2) &= 1 & Post(p_2, t_3) &= 1 \\ Pré(p_3, t_4) &= 1 & Post(p_2, t_4) &= 1 \end{aligned}$$

Le marquage  $M$  représenté ici est  $M(p_2) = 1$ ,  $M(p_1) = M(p_3) = 0$ .

Un jeton dans la place  $p_2$  signifie que la ressource est disponible. Un jeton dans la place  $p_1$  (resp.  $p_3$ ) signifie que la tâche  $a$  (resp.  $b$ ) dispose de la ressource commune. Les événements associés aux transitions  $t_1$  et  $t_2$  signifient respectivement "réservation de la ressource par la tâche  $a$ " ou "réservation de la ressource par la tâche  $b$ ", et ceux associés à  $t_3$  et  $t_4$  concernent la restitution de la ressource.

## Rôle des éléments de la structure de contrôle :

- ♠ Les places d'un réseau de Petri doivent être vues comme une extension du concept d'état des systèmes séquentiels. Cette extension a deux facettes :
  - ▷ La signification associée à une place est locale ou partielle. Par opposition, dans un graphe d'états, chaque état correspond à une situation précise de l'ensemble du système modélisé. Dans un réseau de Petri au contraire, l'information d'état associée à une



place est limitée au sous-ensemble d'éléments choisis par le concepteur. Il est courant de rencontrer une place représentant l'état de repos d'une ressource indépendamment de l'état du reste du système. Seule la vision de l'ensemble des états partiels (des places) donne une vision globale du système modélisé ;

- ▷ L'état d'une place d'un réseau de Petri n'est pas astreint à prendre une valeur booléenne. Le marquage d'une place est représenté par un entier naturel. Une place associée à la signification " outils disponibles " peut donc contenir autant de marques qu'il y a effectivement d'outils disponibles. En tout état de cause, les places servent à modéliser des états partiels qui peuvent être pris par les entités modélisées dans le réseau.
- ♠ Le nom des transitions est explicite et véhicule bien le concept de changement d'état associé à ces éléments. Une transition, lorsqu'elle est franchie, déplace les marques du réseau de ses places d'entrée vers les places de sortie. Naturellement, lors d'une modélisation d'un système par réseau de Petri, les transitions représentent les évolutions et les événements significatifs pour l'application considérée.
- ♠ Les arcs relient les places et les transitions dans le modèle. Ils apportent pour une transition donnée l'information sur le nombre de marques devant être ôtées des places d'entrée et le nombre de marques devant être déposées dans les places de sortie. Dans un modèle d'une application donnée, les arcs place d'entrée - transition  $t$  indiquent explicitement quels états partiels doivent être vérifiés simultanément pour que l'évolution modélisée par la transition  $t$  puisse avoir lieu. Symétriquement, les arcs transition  $t$  - places de sortie désignent les états partiels atteints après que l'évolution de la transition  $t$  a eu lieu.
- ♠ Les jetons et les entités modélisées : les éléments de la structure de contrôle ne modélisent pas les entités du système qui changent d'état. Ce rôle est dévolu aux jetons. Deux types d'entités peuvent être distingués dans une application. Cette partition s'appuie sur le caractère réversible ou non des changements d'état subis par l'entité considérée au sein du modèle que nous nous proposons d'établir.

Les entités qui subissent des changements d'état réversibles sont appelées des ressources et ne disparaissent jamais du modèle.

Les entités subissant des changements d'état définitifs sont appelées des produits et sont amenées à apparaître et à circuler dans le modèle pour enfin en disparaître.

Généralement, tout le but de l'application qui est modélisée consiste à " faire circuler les produits " comme l'impose le cahier des charges. Toutes les évolutions ne mettant pas en jeu un produit sont des évolutions nécessaires pour le bon fonctionnement des ressources, mais sans valeur réelle pour l'application considérée. Cette distinction ressource/produit n'est pas nécessaire pour établir un modèle à réseau de Petri, mais concourt toutefois à une clarification du rôle des entités gérées par le modèle. Cette partition est une manière d'aborder une des questions fondamentales devant tout problème de modélisation : que doit-on modéliser ?

## **Modélisation d'un cahier des charges**

N'intégrant pas explicitement de mécanisme d'abstraction dans son formalisme, il est difficile d'aborder un problème de modélisation dès que celui-ci comporte quelques traitements effectués en parallèle. En effet, on se sent souvent perdus au milieu des détails de chaque

processus pour avoir une vision globale du modèle et des rares points de synchronisation. Le concept d'activité nous permet de contourner cette difficulté en ne s'intéressant d'abord qu'à un modèle global qui peut ensuite être affiné. Une activité est par définition une boîte noire permettant d'encapsuler un ensemble d'évolutions. Elle est délimitée par les deux événements " début " et " fin ". L'exécution des opérations constituant le corps de l'activité ne réclame aucune interaction avec l'environnement autre que celles spécifiées globalement pour toute l'activité.

Ce concept d'activité est applicable à l'agrégation de traitements élémentaires comme dans l'exemple précédent, mais convient aussi parfaitement pour l'agrégation d'activités déjà abstraites (par exemple, une activité de fabrication d'un produit regroupe perçage, nettoyage, transport, traitement de surface, etc.). Une représentation d'un système par activités est donc toujours possible quelle qu'en soit la complexité.

La place activité en cours encapsule un réseau de Petri plus complexe décrivant précisément l'enchaînement des opérations élémentaires constituant l'activité. Les transitions " début " et " fin " modélisent les " frontières " de l'activité. Les places en amont de la transition " début " sont les preconditions. Elles représentent l'état dans lequel les entités impliquées dans l'activité doivent se trouver pour que son exécution ait lieu. Symétriquement, les places en aval de la transition " fin " sont les postconditions, ou états découlant de l'exécution de l'activité. Un modèle plus abstrait de l'activité est envisageable: une seule transition remplace la transition " début ", la place " en cours " et la transition " fin ".

## Démarche de modélisation

Le processus de modélisation est conduit suivant l'algorithme ci-après :

- ♠ Identifier les entités dont l'état doit être connu par le modèle de commande et si possible les caractériser (ressource ou produit) ;
- ♠ Identifier les activités du cahier des charges ;
- ♠ Pour chaque activité, identifier les preconditions, les postconditions et établir son modèle à réseau de Petri ;
- ♠ Assembler les modèles d'activités en mettant en commun les éléments apparaissant plusieurs fois (places et transitions) ;
- ♠ Considérer chaque activité comme un nouveau cahier des charges et répéter cette démarche jusqu'à l'obtention d'un modèle de la précision voulue ;
- ♠ Simplifier (éventuellement) le modèle général ainsi obtenu. Chaque point de cet algorithme à l'exception du premier mérite un développement avant que nous n'abordions un exemple complet de modélisation.

On notera  $t^0$  (resp  ${}^0t$ ) l'ensemble des places de sorties (respectivement d'entrées) de la transition  $t$ .

On notera  $p^0$  (resp  ${}^0p$ ) l'ensemble des transitions de sorties (respectivement d'entrées) de la place  $p$ .

Le marquage  $M$  d'un Réseau de Petri est une application  $M: P \rightarrow N$  La quantité  $M(p)$  détermine le marquage de la place  $p$ . Un réseau marqué est déterminé par le couple  $N = (R, M_0)$  formé d'un réseau de Petri  $R$  et d'un marquage initial  $M_0$ . Une transition sans place amont est dite transition source et une transition sans place aval est dite transition puits.

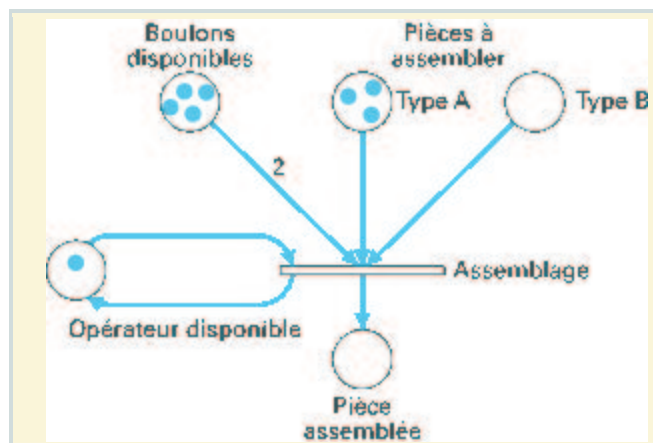


FIG. 2.2 – Assemblage de deux pièces par un opérateur.

L'un des intérêts des réseaux de Petri est qu'il est extrêmement aisé de les visualiser. En effet on peut représenter un réseau de Petri comme un graphe orienté biparti, dont les sommets sont les places et les transitions du réseau. Un arc relie une place  $p$  à une transition  $t$  si et seulement si  $\text{Pré}(p, t) \neq 0$ . Un arc relie une transition  $t$  à une place  $p$  si et seulement si  $\text{Post}(p, t) \neq 0$ . Les valeurs non nulles des applications  $\text{Pré}$  et  $\text{Post}$  sont associées aux arcs comme valuations (par défaut on prend la valeur 1). On représente une place par un cercle et une transition par un rectangle. Un marquage  $M$  est représenté sur le graphe par  $M(p)$  points ou jetons en chaque place  $p$ .

**Exemple 12** *Pour illustrer les différents concepts évoqués jusqu'ici, prenons l'exemple d'un processus d'assemblage de deux pièces par un opérateur à l'aide d'un boulon . La figure 2.2 donne le modèle à réseau de Petri correspondant.*

*Il est simple de retrouver dans ce modèle :*

- ▷ *Les deux places Type A et Type B correspondant à l'état partiel que doivent occuper les deux éléments principaux de l'assemblage ;*
- ▷ *La nécessaire présence de boulons dont deux sont consommés au moment de l'assemblage ;*
- ▷ *La présence requise de l'opérateur qui peut renouveler l'opération à l'infini (ce modèle ne représente pas toutes les facettes du travail de l'opérateur) ;*
- ▷ *La conséquence de l'assemblage qui crée une " pièce assemblée " et fait disparaître deux " boulons disponibles ", une pièce " Type A " et une pièce " Type B " .*

*Notons enfin que, vu l'état courant du poste de travail, une opération d'assemblage n'est pas possible puisqu'il manque une pièce " Type B " .*

## Tir des transitions

L'évolution au cours du temps des marques des places dans un réseau de Petri se fait selon le processus d'activation (ou de tirage) des transitions décrit ci-après. Étant donné un réseau

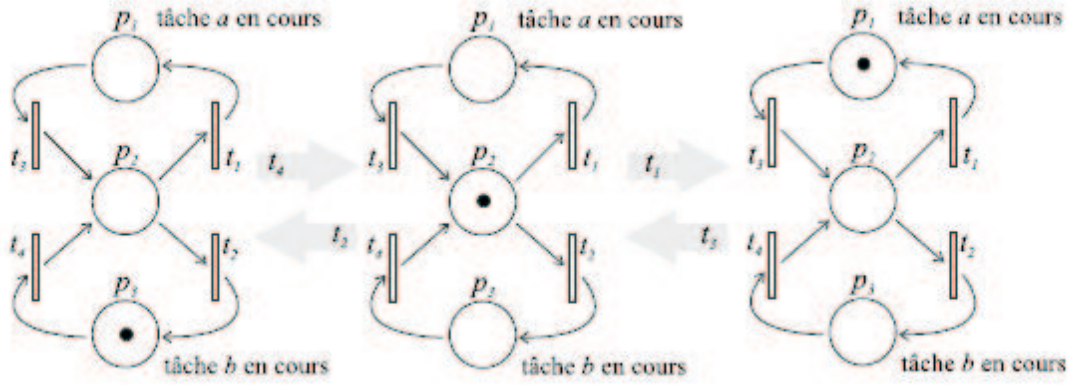


FIG. 2.3 – Evolution de marquage d'un réseau de pétri.

R et un marquage M, on dit que la transition  $t \in T$  est franchissable pour M si l'on a :

$$\forall p \in P, M(p) \geq pr(p,t)$$

Lorsque cette condition est satisfaite, l'activation (le tirage) de la transition  $t \in T$  conduit à un nouveau marquage  $M_0$  défini par :

$$M_0(p) = M(p) - Pr(p,t) + Post(p,t)$$

Dans la figure 2.3 on peut voir l'évolution d'un réseau de pétri. Plus généralement, pour un Rdp, en posant,  $W^- = [Pr(p,t_j)]$  (la matrice d'incidence avant),  $W^+ = [Post(p,t_j)]$  (la matrice d'incidence arrière),  $W = W^+ - W^-$  (la matrice d'incidence) et en considérant S une séquence de franchissements réalisable à partir d'un marquage  $M_i$ , il est alors possible de donner l'équation fondamentale suivante :

$$M_k = M_i + W.S,$$

où S est le vecteur caractéristique de la séquence de franchissement S, dont la dimension est égale au nombre des transitions du réseau de Petri et dont la composante numéro j correspond au nombre de franchissements de la transition  $t_j$  de la séquence S.

### Ensemble des marquages accessibles

Soit  $N = (R, M_0)$  un réseau de Petri. L'ensemble des marquages accessibles  $A(R, M_0)$  d'un réseau de Petri marqué est l'ensemble des marquages que l'on peut atteindre à partir du marquage initial  $M_0$  par une séquence de franchissement, c.-à-d.:

$$A(R, M_0) = \{M_i, \exists t M_0 \rightarrow M_t\}$$

On peut, lorsque cet ensemble est fini, le représenter sous la forme d'un graphe. Les sommets de ce graphe correspondent aux marquages accessibles de  $A(R, M_0)$ . Un arc orienté relie deux sommets  $M_i$  et  $M_j$  s'il existe une transition t franchissable permettant de passer d'un marquage à un autre:  $M_i \rightarrow M_j$ .

**Exemple 13** La figure 2.4 présente le graphe de marquage accessible du réseau de pétri de la figure 2.1

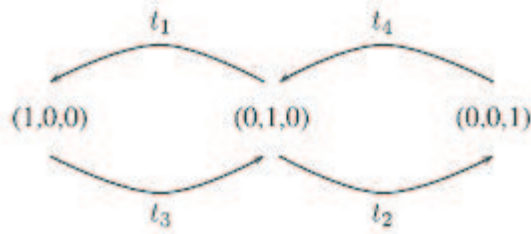


FIG. 2.4 – Graphe des marquages accessibles du réseau de Petri de la figure 2.1.

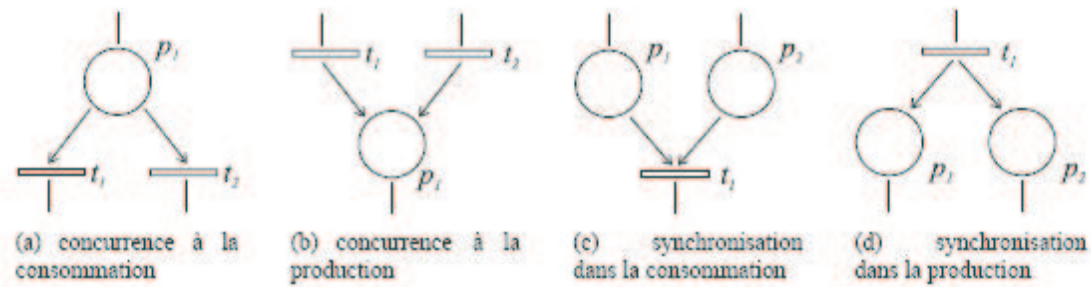


FIG. 2.5 – Concurrency et synchronisation dans les RDP

## 2.2.2 Modélisation des systèmes à événements discrets :

Le formalisme des réseaux de Petri est très puissant, il permet de représenter une grande variété de comportement des systèmes réels mettant en jeu des phénomènes de concurrence et de synchronisation. Sur la figure 2.5, nous avons représenté quatre structures de réseaux de Petri permettant de décrire ces phénomènes.

- ♠ La figure 2.5(a) représente une structure de choix. La place  $p_1$  a deux transitions en aval, notées  $t_1$  et  $t_2$ . Le tir de  $t_1$  ou de  $t_2$  consommera le jeton de la place  $p_1$  et exclura donc le tir de l'autre. Une telle configuration permet de modéliser un phénomène de concurrence à la consommation comme par exemple le partage d'un processeur entre tâches concurrentes dans un système informatique.
- ♠ La configuration de la figure 2.5(b) permet de représenter une concurrence à l'apport de jetons dans une place, ici  $p_1$ .
- ♠ La configuration de la figure 2.5(c) permet de modéliser un phénomène de synchronisation. La transition  $t_1$  n'est franchissable que lorsque les places  $p_1$  et  $p_2$  contiennent toutes les deux au moins un jeton.
- ♠ Dans la structure de la figure 2.5(d), l'apport de jetons dans les places  $p_1$  et  $p_2$  est synchronisé par le tir de la transition  $t_1$  et permet, par exemple, de modéliser l'initiation simultanée de différentes tâches dans un système à événements discrets.

## 2.2.3 Quelques propriétés des RdP :

Le graphe des marquages associé à un RdP fournit des indications essentielles sur le fonctionnement du système qu'il représente. Suivant que le graphe des marquages est fini ou

infini, il reflète certaines des propriétés du système modélisé.

**Définition 14 (Bornitude)** Une place  $p \in P$  d'un réseau de Petri marqué  $(R, M_0)$  est  $k$ -bornée ( $k \in \mathbb{N}$ ) si pour tout marquage accessible  $M \in A(R, M_0)$ , le marquage de cette place vérifie  $M(p) \leq k$ . Dans le cas contraire, nous dirons que  $p$  est non-bornée. Enfin si  $p$  est 1-bornée, on dit que  $p$  est binaire (safe). Si on considère le Réseau de Petri de la figure AA on voit que pour le marquage initial  $M_0$  les places  $p_1$ ;  $p_2$  et  $p_3$  sont binaires (1-bornées) et le réseau est dit sauf. Lorsqu'un RdP modélise un système manufacturier, certaines places représentent des convoyeurs ou bien des zones de stockage intermédiaires. La Bornitude du marquage du modèle est alors synonyme de limitation de la taille des stocks internes du système. La Bornitude du RdP reflète alors en quelque sorte une propriété de "stabilité" du système de production modélisé.

**Définition 15 (Vivacité)** Étant donné un réseau de Petri  $R$  et un marquage initial  $M_0$ , une transition  $t$  est vivante pour le réseau marqué  $(R, M_0)$  si pour tout marquage  $M$  accessible depuis  $M_0$ , il existe une suite de transitions  $s$ , comportant au moins une fois la transition  $t$ , et telle que  $s$  soit franchissable pour  $M$ . Un réseau de Petri marqué  $(R, M_0)$  est dit vivant si toutes ses transitions sont vivantes. La non-vivacité révèle souvent un problème de conception du système modélisé; elle montre qu'une partie du système n'est, à partir d'un état donné, plus sensible aux événements externes.

## 2.3 Les graphes d'événements temporisés :

On peut définir formellement les graphes d'événements à partir des réseaux de Petri.

**Définition 16 (Graphe d'événements)** Un graphe d'événements est un réseau de Petri tel que toute place  $p \in P$  a exactement une transition amont et une transition aval (nous nous restreignons au cas où tous les arcs ont une valuation de 1). Notons que les graphes d'événements ne permettent pas de modéliser les situations de concurrence telles que celles illustrées par les figures 2.5(a) et 2.5(b). Néanmoins, cette sous-classe de réseaux de Petri est intéressante pour de nombreuses applications où l'on doit essentiellement modéliser des contraintes de synchronisation entre plusieurs processus (voir figure 2.5(c) et figure 2.5(d)).

### 2.3.1 Propriétés des graphes d'événements :

Nous rappelons brièvement quelques caractéristiques des graphes d'événements au travers des propositions suivantes.

**Proposition 1** Dans un graphe d'événements, le nombre de jetons d'un circuit élémentaire est constant.

Rappelons qu'un circuit élémentaire est un chemin qui commence et se termine au même sommet. Alors, si une transition franchissable appartenant à un circuit élémentaire est franchie, son franchissement prend un jeton dans une des places amont du circuit pour le remettre immédiatement dans une autre place aval du circuit. L'opération de franchissement d'une transition d'un circuit laisse donc invariant le nombre de jetons du circuit.

**Proposition 2** Soit  $R$  un graphe d'événements et  $M_0$  son marquage initial, alors  $(R, M_0)$  est vivant si et seulement si tout circuit élémentaire contient une place initialement marquée.

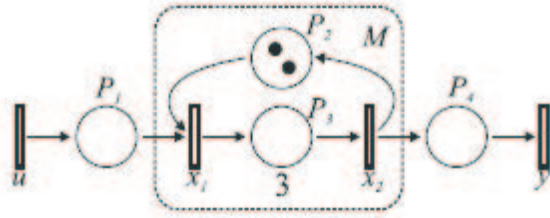


FIG. 2.6 – Modèle GET d'une machine

Supposons qu'un circuit élémentaire d'un graphe d'événements ne soit pas initialement marqué. En référence à la proposition 1, ce circuit ne contiendra jamais de jeton et donc toutes ses transitions sont en permanence non franchissables : le graphe d'événements n'est donc pas vivant. Inversement, dans un graphe d'événements non vivant, une transition morte (qui n'est jamais tirée) possède obligatoirement en amont une transition également morte. En remontant ainsi d'une transition morte à une autre située en amont, on aboutit inéluctablement à une transition appartenant à un circuit, circuit qui est donc nécessairement sans jeton.

### 2.3.2 Les graphes d'événements P-temporisés :

Un graphe d'événements est dit P-temporisé (GET) si à chaque place  $p \in P$  est associé un temps  $\Theta(p)$ , où  $\Theta: P \rightarrow \mathbb{N}$  est l'application qui à toute place  $p \in P$  associe sa temporisation. Un temps  $\Theta(p)$  est interprété comme la durée minimale de séjour d'un jeton dans la place  $p$ .

**Exemple 14** Par exemple pour le graphe d'événements temporisé de la figure 2.6, le marquage initial de la place située entre les transitions  $x_1$  et  $x_2$  étant nul, si la transition  $x_1$  est franchie à la date  $t_0$ , la transition  $x_2$  n'est pas franchissable avant la date  $t_0 + 3$ . De même, à chaque transition  $t \in T$  il est possible d'associer un temps  $\Theta(t)$  représentant la durée minimale d'activation de la transition  $t$ . Il est toujours possible de se ramener au cas où seules les places sont temporisées.

Le graphe d'événements p-temporisé de la figure 2.6 peut représenter une cellule de production fonctionnant comme suit : deux machines sont disponibles (les 2 jetons dans la place  $P_2$  signifient que les 2 machines sont libres) pour usiner des pièces. Les pièces sont amenées à l'une ou l'autre des deux machines travaillant en parallèle (place  $P_3$ ), le traitement d'une pièce prend au moins 3 unités de temps. A l'issue du traitement, la pièce finie est déposée dans un stock  $P_4$  (supposé de capacité infinie) et la machine ayant traité la pièce redevient disponible (1 jeton revient dans  $P_2$ ), en attendant une nouvelle pièce.

Notons qu'au plus deux pièces peuvent être simultanément traitées, puisque d'après la proposition 2.8, le nombre de jetons dans un circuit est constant.

### 2.3.3 Prise en compte des contraintes temporelles :

Il n'existe pas de référence de temps commune à toutes les places d'un réseau de Petri. Chaque place évolue indépendamment des autres (sauf si elles partagent une transition). En revanche, une horloge locale peut être associée à une place pour prendre en compte des temporisations de manière explicite. Le temps commence à s'écouler au moment où un jeton

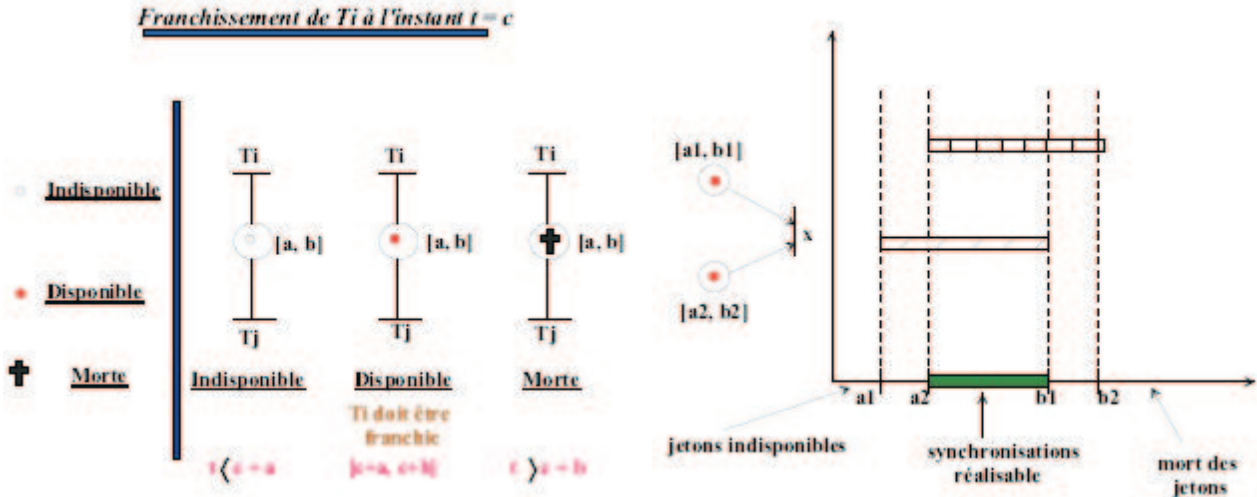


FIG. 2.7 – Présentation formelle; États des marques

est présent dans la place et une fenêtre de franchissement est définie par un intervalle fermé associé à la transition, tenant compte de toutes les places en amont de celle-ci.

Des intervalles de temps peuvent donc être associés aux places de ces réseaux. La sémantique de ces intervalles est la durée de séjour admissible d'une marque dans une place. La violation de cette spécification se traduira par la mort des marques. Cette notion des marques-mortes est spécifique à ce nouveau modèle. L'existence et la détection de jetons morts sont un point important du modèle: par exemple, dans un contexte de chaîne de production avec trempage de pièces dans des bains chimiques, un jeton mort représente une pièce qui restée trop longtemps dans un bac.

**Définition 17 (Réseaux de Petri P-temporels)**

Un réseau de Petri P-temporel (p-RdP) est une paire  $\langle R, IS \rangle$  où :

- ♠  $R$  est un réseau de Petri  $\langle P, T, Pre, Post \rangle$  muni d'une marquage initiale  $M_0$ .
- ♠  $IS$  est une fonction qui à chaque place  $p_i$  fait correspondre un intervalle fermé rationnel.

$$IS : P \rightarrow (Q^+ \cup 0) \times (Q^+ \cup \infty)$$

$$p_i \rightarrow IS_i = [a_i, b_i] \text{ tel que } 0 \leq a_i \leq b_i$$

$IS_i$  décrit des durées de sensibilisation. En effet,  $IS_i$  définit l'intervalle statique de temps de séjour d'une marque dans la place  $p_i$  ( $Q^+$  est l'ensemble des nombres rationnels positifs). Une marque dans la place  $p_i$  participe à la validation de ses transitions de sortie que si elle a séjourné au moins la durée  $a_i$  dans cette place. Elle doit alors quitter la place  $p_i$ , donc franchir l'une des transitions de sortie au plus tard quand sa durée de séjour devient  $b_i$ . Après ce temps ( $b_i$ ), la marque sera "morte" et ne participera plus à la validation des transitions.

**Définition 18 (Sensibilisation)**

Soit  $R$  un réseau de Petri,  $M$  un marquage, et  $t$  une transition. On dit que la transition  $t$  est sensibilisée par le marquage  $M$ , ce que l'on note  $M[t >$  ou  $M \xrightarrow{t}$  ssi le marquage des places en entrée de  $t$  est suffisant pour le tir.



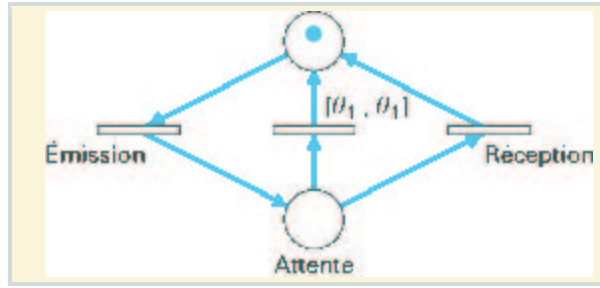


FIG. 2.8 – Mécanisme du chien de garde

**Exemple 15** Nous citons ici l'utilisation des réseaux de Petri temporisés pour la modélisation de mécanismes de surveillance de base : les chiens de garde (time out ).

Supposons que dans un système de communication, un émetteur envoie un message sur un réseau et se place en attente d'une réponse. Sans mécanisme particulier, en cas de perte de message, l'émetteur se bloque en attente d'une réponse qui ne viendra jamais. L'adjonction d'un temporisateur qui déclenche une réémission au bout d'un temps donné permet d'éviter la situation de blocage. Le modèle de ce mécanisme est montré sur la figure 2.8.

Les bornes de l'intervalle de la transition du chien de garde sont identiques. Cette transition doit donc être franchie après avoir été sensibilisée exactement pendant une durée  $\theta_1$ . Le seul événement qui puisse interdire son franchissement est sa désensibilisation par le tir de la transition "réception". On a bien modélisé une attente de durée maximale  $\theta_1$ . Notons que le traitement devant être appliqué dans ces circonstances est un problème très délicat qui relève de la sûreté de fonctionnement. Dans l'exemple, l'émetteur ne se bloque plus jamais en attente, pourtant il peut rentrer dans un cycle sans fin d'émission-attente...

### Règle de fonctionnement :

Pour pouvoir déterminer le fonctionnement, il faut d'abord caractériser la situation du réseau à un instant donné, puis déterminer les conditions de franchissement des transitions. Enfin le passage d'une situation à une autre devra être établi. On utilisera la notion "d'état" pour caractériser la situation du réseau à un instant donné. En fait, l'évolution dynamique d'un p-RdP dépend des marques et de leurs situations temporelles.

### Définition 19 ( État d'un p-RdP )

A un instant donné, l'état est totalement déterminé par la paire  $E = \langle M, I \rangle$  telle que :

- ♠  $M$  est une application de marquage, assignant à chaque place du réseau un certain nombre de marques tels que

$$(\forall p \in P, M(p) \geq 0).$$

- ♠  $I$  est une application "intervalle potentiel" de tir, associant à chaque marque  $k$  dans une place  $p_i$  un intervalle  $[a_i^k, b_i^k]$ . On l'appellera "intervalle dynamique" de tir pour le distinguer de l'intervalle statique (initial) associé à la place qui contient cette marque. Ces intervalles (les intervalles dynamiques) sont relatifs à l'instant d'arrivée de la marque dans la place. Supposons que la marque  $k$  arrive dans la place  $p_i$  (son intervalle

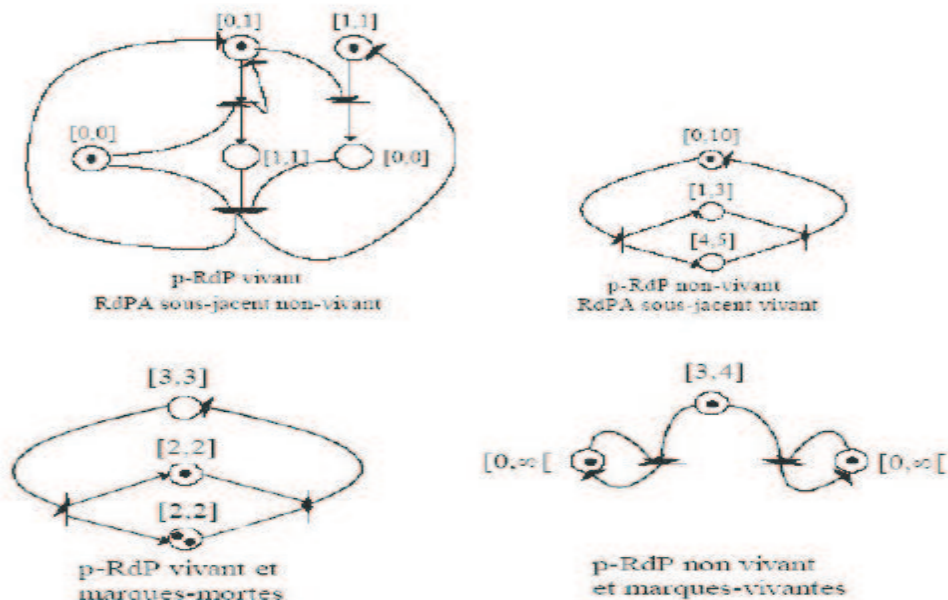


FIG. 2.9 – Vivacité des marques, Non conservation de vivacité entre les p-RdP et les RdPA sous-jacents

statique est  $[a_i, b_i]$ ) à l'instant  $c$ , à l'instant  $c + d$  ( $a_i \leq d \leq b_i$ ) l'intervalle dynamique de  $k$  est

$$[a_i^k, b_i^k] = [\max(a_i - d, 0), b_i - d]$$

Le tir d'une transition dépend des intervalles dynamiques associés aux marques dans toutes ses places d'entrée. Par conséquent, la connaissance de l'intervalle dynamique associé à une marque dans une de ses places d'entrée ne donne pas les informations suffisantes sur le tir de cette transition (on a besoin de tous les intervalles dynamiques associés aux marques dans ses places d'entrée). C'est la raison pour laquelle on a utilisé le mot "potentiel". On constate que deux problèmes peuvent survenir dans les p-RdP :

- ♠ Le blocage classique (un marquage tel qu'aucune transition n'est validée).
- ♠ La mort des marques (l'intervalle dynamique d'une marque devient  $[0,0]$  et aucune transitions de sortie de la place contenant cette marque n'est validée à cet instant). Ce problème est spécifique aux p-RdP. On aura donc un réseau marques mortes ou un réseau marques-vivantes.

#### Condition de franchissement d'une transition :

Dans les p-RdP, la détermination de l'intervalle de tir d'une transition est compliquée. En effet, on détermine l'intervalle potentiel de tir d'une transition en ne considérant que, dans un premier temps, les intervalles potentiels associés aux marques dans les places d'entrée de cette transition. On détermine, dans un second temps, l'intervalle réel de tir de cette transition en considérant cette fois ci tous les intervalles potentiels associés aux marques dans les autres places.

**Définition 20** Une transition  $t_1$  est potentiellement tirable à partir de l'état  $E(M,I)$  si et seulement si :

1- elle est validée au sens des RdP autonomes dans cet état.

2-  $\forall p_i \in {}^0 t_1$ , il existe au moins  $Pr(p_i, t_1)$  marques dans cette place telles que

$$\bigcap_k [a_i^k, b_i^k] \neq \emptyset, \quad \text{ou} \quad k = 1, 2, \dots, Pr(p_i, t_1)$$

De plus, il n'existe pas des marques ( qui ne participent pas au franchissement de la transition  $t_1$  ) dans cette place dont les bornes supérieures de leur intervalles dynamiques sont strictement inférieure à  $a_i^j$ . Sinon, il y a des marques qui meurent et par conséquent, le réseau sera marques-mortes. Alors chaque place  $p_i$  a un intervalle potentiel  $[a_i^j, b_i^j] \neq \emptyset$ . Enfin, l'intervalle d'intersection de toutes les places d'entrée de  $t_1$ ,  $[a_1, b_1]$  doit être non vide ( sinon il y a des marques qui seront mortes ).  $[a_1, b_1]$  c'est l'intervalle dans lequel  $t_1$  reste potentiellement tirable.

### Vivacité et blocage :

Les notions de vivacité des transitions et de blocage sont similaires à celles évoqués pour les RdPA (Réseau de Petri Autonome) . En plus de ces deux propriétés précédentes des p-RdP, un autre problème est l'étude de la mort des marques (Vivacité des marques). Ce problème découle du fait qu'une marque peut mourir dans une place précédant immédiatement une transition de synchronisation avant la validation de celle-ci. Un état  $E_i$  d'un p-RdP est marques-vivantes, si toutes les marques dans  $M_i$  sont vivantes. Un p-RdP est marques-vivantes pour un marquage initial  $M_0$  (l'état initial  $E_0$ ) si toutes les marquages des états accessibles depuis  $M_0$  sont des états marques-vivantes. Si une marques dans un marquage d'un état accessible depuis  $E_0$  est morte, alors le réseau est marques-mortes.

Rappelons qu'une marque qui meurt ne participe plus à la validation des transitions.

**Remarque 9**  $\diamond$  La propriété de vivacité ne se conserve pas forcément entre les p-RdP et leurs RdPA (Réseau de Petri Autonome) sous-jacent. La figure 2.9 illustre ce résultat.

$\diamond$  Notons qu'un p-RdP peut être vivant sans être marques-vivantes. L'inverse est aussi vrai : un p-RdP peut être marques-vivantes sans être vivant (figure 2.9).

## 2.4 Validation d'un modèle :

Un réseau de Petri représentant la commande du procédé a été obtenu par l'étape d'analyse et modélisation. Il a été construit par composition de fonctions ou de comportements qui caractérisent le fonctionnement de parties du système à commander.

Le modèle ainsi obtenu est réputé complet et censé répondre au fonctionnement attendu. Toutefois, est-ce que ce modèle fournit une description correcte de ce qui est souhaité? Est-ce que la composition d'éléments de base n'a pas introduit de perturbation croisée de ces éléments? Pour cela, il faut valider ce résultat, pour tenter de mettre en évidence des erreurs éventuelles de modélisation, et confirmer la conformité du modèle aux spécifications du cahier des charges par la vérification de toutes les contraintes de départ.

Le modèle à réseau de Petri présente la particularité de pouvoir être analysé : à sa représentation graphique est associée une représentation numérique sur laquelle il est possible de mener des opérations analytiques formelles.

Comme il représente le comportement du système de commande, il peut aussi être simulé, c'est-à-dire être parcouru sans pour autant être relié au procédé, à la seule fin de vérifier que les cheminements qu'il est possible de parcourir restent cohérents au regard du fonctionnement attendu. Il faut toutefois garder présent à l'esprit que la modélisation graphique manuscrite est nécessairement traduite par une méthode ou une autre de mise en oeuvre, et il est alors très facile d'oublier de décrire (ou d'ajouter) un arc, de se tromper dans une pondération ou une expression logique.

En cela, les méthodes de validation peuvent aussi être trompeuses dans la mesure où un réseau à analyser doit être traduit dans le langage de l'outil de vérification. La plus grande attention doit donc être portée à la matérialisation du réseau de Petri, aussi bien dans sa description lors de la mise en oeuvre que pour son analyse afin que, dans ce contexte, les méthodes de validation et vérification puissent apporter tout leur intérêt pour conclure sur la bonne description du modèle.

### 2.4.1 Validation par analyse :

La validation par analyse consiste à vérifier quelques indicateurs en appliquant des techniques mathématiques à la représentation du système modélisé. L'exploitation de ces résultats permet de conclure sur la validité de la représentation et sur son adéquation avec le cahier des charges initial.

C'est une étape de vérification formelle de la structure du réseau, dont l'objectif est de retrouver pour le réseau un certain nombre de propriétés générales, appelées les "bonnes propriétés", indépendantes de la signification donnée par l'enchaînement des places et transitions du réseau. Elles garantissent, lorsqu'elles sont vérifiées, que le réseau est bien structuré. Toutefois, elles ne permettent pas de confirmer que le réseau respecte le cahier des charges qui en a conditionné l'obtention. Cela ressort du domaine des "propriétés spécifiques", inhérentes à chaque modélisation, et spécifiques à chaque système modélisé.

Il existe malgré tout des pistes pour vérifier cette spécificité, les différents systèmes ayant tout de même des bases communes de modélisation. Cette forme de validation ne s'appuie que sur la structure du réseau de Petri. Il faut donc en enlever toute l'interprétation qui constitue une partie du support d'information (bien que la structure de commande représente elle aussi de l'information : l'information structurelle du réseau).

Trois " bonnes propriétés " sont définies pour les réseaux de Petri :

- ♠ réseau borné : cette propriété est une caractérisation de la finitude des ressources. Même très importante, la capacité disponible n'est pas infinie, et cette borne est une information fournie lors du dimensionnement de l'application ; elle doit absolument être respectée ;
- ♠ réseau réinitialisable : hormis une phase de démarrage, facilement identifiable, la commande d'un procédé est toujours décrite par un fonctionnement cyclique. De toute

situation atteinte à un instant donné, il faut pouvoir retourner à la situation de départ, et redémarrer un nouveau cycle de commande ;

- ♠ réseau vivant : par sa vérification, cette propriété donne l'assurance qu'il n'y a pas de partie de réseau inutile ou inaccessible. Cela permet de mettre en évidence le fait, par exemple, qu'une erreur a été commise lors de la modélisation (ressource utilisée, mais non restituée en fin d'utilisation, par exemple).

Si l'une des trois propriétés n'est pas vérifiée, alors il y a lieu de s'inquiéter sur la modélisation du système de commande ou sur la saisie du réseau de Petri.

## 2.5 Conclusion :

Les systèmes dynamiques d'événement discrets impliquant la synchronisation, peuvent être modélisés par plusieurs types de réseaux de Pétri. Parmi ces réseaux de Pétri, nous pouvons citer des réseaux de Pétri P-temporels, les réseaux de Pétri à contraintes temporelles qui prolongent le champ d'application de l'événement synchronisé. L'événement de temps représenté graphiquement par exemple, laisse définir les conditions de synchronisation des applications et peut décrire des synchronisations complexes. Les réseaux de Pétri à contraintes de temps présentent différents types de sémantique, comme les " et ", " faibles-ET ", " forts-OU ", " OU ", " maître " et leurs variations, qui correspondent aux différentes évolutions temporelles.

Les graphes d'événement P-temporels et les graphes d'événement à contraintes temporelles peuvent être modélisés par une nouvelle classe des systèmes appelés le système de descripteur d'intervalle pour lequel l'évolution de temps n'est pas strictement déterministe mais dépend des intervalles. Généralement, les limites inférieures et supérieures dépendent des opérations de maximisation, de minimisation et d'addition.

Dans la suite de ce mémoire, on considère les systèmes d'équations déterministes (Min, Max, +), présenter la théorie spectrale ainsi que des outils de résolution basés sur le point fixe.

# Chapitre 3

## THÉORIE SPECTRALE ET POINT FIXE

### 3.1 Introduction :

L'intérêt pour la classe des fonctions (min, max, +), et la classe plus large des fonctions topicales, est venu d'un certain nombre de directions et a rassemblé un certain nombre de thèmes distincts, tels que les systèmes à événement discrets, la théorie de Perron-Frobenius, l'algèbre de matrice max-plus, des théorèmes de point fixe pour des fonctions nonexpansive, la dynamique non-linéaire, etc... En raison de l'apparition récente de ce secteur, des études ont été consacrées pour amplifier les grandes lignes de la théorie spectrale et celle du point fixe. Des cas spéciaux des fonctions (min, max, +) ont été étudiés par Olsder [33]. Suite à certains résultats incorporés [6, 19, 20], nous présentons une description détaillée de la technique spectrale et de la théorie de points fixe aussi bien que la description d'un nouveau matériel de construction de point fixe, comme nous développons sous scilab cet outil, pour finir par un exemple d'application obtenu à partir de ce programme.

Une fonction (min, max, +)  $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$  est établie à partir des termes de la forme  $x_i + a$ , où  $1 \leq i \leq n$  et  $a \in \mathbb{R}$ , par application finie de plusieurs opérations max et min pour chaque composante. Par exemple:

$$F_1(x_1, x_2) = \max(\min(\max(x_1 + 1, x_2 - 1.2), \max(x_1, x_2 + 2)), \min(x_1 + 0.5, x_2 + 1))$$

$$F_1(x_1, x_2) = \min(\max(x_1 + 7, x_2 + 4.3), \min(x_1 - 5, x_2 - 3))$$

De telles fonctions sont homogènes,  $F_i(x_1 + h, \dots, x_n + h) = F_i(x_1, x_n) + h$ , monotones,  $\vec{x} \leq \vec{y} \Rightarrow F(\vec{x}) = F(\vec{y})$ , et nonexpansive  $\|F(\vec{x}) - F(\vec{y})\| \leq \|x - y\|$ , nous suivons Gunawardena et Kaene et nous les appellerons des fonctions topicales [2, 23, 28, 31, 40]. Elles incluent (probablement après transformation appropriée) les matrices non négatives, les systèmes de substitution de Leontieff, les opérateurs de programmation dynamique des jeux et des procédés de décision de Markov, les opérateurs non linéaires surgissant dans des problèmes de graduation de matrices... [12, 24]. Elles incluent également des exemples, tels que les fonctions (min, max, +), qui résultent de la modélisation des systèmes à événements discrets, circuits numériques, des réseaux informatiques...

N'importe quelle fonction topicale  $T$ , peut être rapprochée par les fonctions  $(\min, \max, +)$  de telle manière qu'un certain comportement dynamique de  $T$  soit hérité par des approximations (voir le lemme 3,1). Pour n'importe quelle fonction topicale, un point fixe est un vecteur,  $x = (x_1, \dots, x_n) \in \mathbb{R}^n$ , pour lequel il existe  $h \in \mathbb{R}$ , tels que  $F(x) = (x_1+h, \dots, x_n+h)$ . Le vecteur de temps de cycle,  $\chi(F) = \lim_{k \rightarrow \infty} F^k(\vec{x})/k \in \mathbb{R}$ , fournit la généralisation non-linéaire appropriée de la racine de Perron, ou le rayon spectral (voir le 1.3). La limite en question n'existe pas toujours, [23, théorème 3,1], et cela constitue un problème important non résolu pour caractériser ces fonctions topicales. La propriété nonexpansive de  $F$  garantit que  $\chi$ , quand elle existe, est indépendante de l'état initial,  $\vec{x}$ . Le vecteur propre fournit une mesure de performance dans les systèmes à événements discrets, où il mesure la latence moyenne asymptotique du système (voir le lemme 3.2).

À la différence du rayon spectral conventionnel, le temps du cycle est un vecteur et donne immédiatement une condition nécessaire pour l'existence de point fixe. Si  $F$  a un point fixe, alors la propriété de homogénéité implique que  $\chi(F)$  existe et que  $\chi(F) = (h, \dots, h)$ : le temps de cycle a la même valeur dans chaque composante. (dans le contexte des systèmes à événement discrets ceci représente le fait que le système peut seulement posséder un état d'équilibre quand la latence moyenne de chaque événement est asymptotiquement identique.) Il est intéressant de demander si l'inverse est vraie. En d'autres termes:

$$\exists x \in \mathbb{R}, \text{ tel que } F(x) = (x_1 + h, \dots, x_n + h) \text{ si et seulement si, } \chi(F) = (h, \dots, h). (1)$$

En 1994, une Conjecture sur l'existence de la durée de cycle pour les fonctions  $(\min, \max, +)$  a été proposée. La conjecture affirme que n'importe quelle fonction  $(\min, \max, +)$  a une durée de cycle [17, 13]. La conjecture a été montrée pour prouver le résultat de (1) mais la méthode de preuve était nonconstructive et n'a donné aucun algorithme de recherche du point fixe, [19]. La question de constructibilité est importante dans les applications; par exemple, dans l'étude des circuits numériques, la solution du problème de verification de programme d'horloge exige le calcul du point fixe d'une fonction  $(\min, \max, +)$ . Plus loin on va présenter un théorème constructif de point fixe pour les fonctions  $(\min, \max, +)$  qui est indépendant de la conjecture de dualité et de l'existence des durées de cycle pour les fonctions  $(\min, \max, +)$ , [18]. Nous présentons ci dessous une condition nécessaire et suffisante, en termes de durées de cycle des fonctions de composant  $(\max, +)$ , pour que n'importe quelle fonction  $(\min, \max, +)$  ait un point fixe. L'algorithme basé sur la preuve n'est pas général mais peut être rendu suffisant dans un cas particulier important. Deux corollaires du théorème principal seront récupérés, les deux précédents résultats de point fixe pour les fonctions  $(\min, \max, +)$ , avec des hypothèses plus restrictives [19 théorème 3.1], l'autre est dû à Olsder, applicable pour les fonctions  $\min \max$  séparées satisfaisant certaines conditions, [33 théorème 2.1]. Tous les deux premiers résultats étaient nonconstructifs.

Les méthodes s'appliquent sur une classe spéciale des fonctions  $(\min, \max, +)$  qui peuvent être étudiées par des méthodes linéaires, quoique d'une nature peu habituelle. C'est la classe matrices max-plus,  $\mathbb{R}_{max} = \mathbb{R} \cup -\infty$ , où l'addition et la multiplication sont définies comme  $\max$  et plus, respectivement, la dernière devenue distributive par rapport a la première. Les matrices dans  $\mathbb{R}_{max}$  correspondent aux fonctions  $(\min, \max, +)$  dans lesquels l'opérateur  $\min$  n'est jamais utilisé, donc appelée fonction  $(\max, +)$ . L'algèbre de matrice sur  $\mathbb{R}_{max}$  a été intensivement étudiée [1, 9, 10, 29, 32, 42]. Les fonctions  $(\min, \max, +)$  peuvent être

représentées par des collections d'un nombre fini de matrices max-plus et les propriétés dynamiques de ces dernières, connues par la théorie linéaire sous  $\mathbb{R}_{max}$ .

## 3.2 Fonction Min Max :

Nous commençons par une certaine notation. Les vecteurs dans  $\mathbb{N}$  seront dénotés:  $\vec{x}$ ,  $\vec{a}$  etc... Pour le vecteur évalué en quantités en général, tel que les fonctions  $F: X \rightarrow \mathbb{R}$ , la notation  $F_i$  dénotera le composant  $i$ :  $F(\vec{x}) = (F_1(x); \dots; F_n(x))$ . Nous employons  $x_i$  pour les composants du  $\vec{x}$ . L'ordre partiel sur  $\mathbb{R}$  sera dénoté de la manière habituelle par  $a \leq b$ , mais il sera commode d'utiliser les opérations de treillis de borne supérieure et de borne inférieure:

$$a \vee b = lub(a,b)$$

$$a \wedge b = glb(a,b)$$

La même notation sera employée pour des treillis dérivés de  $\mathbb{R}$ , tel que l'espace de fonction  $X \rightarrow \mathbb{R}$ . L'ordre partiel ici est le plus ordonnant les fonctions:  $f \leq g$  si, et seulement si,  $f(x) \leq g(x)$  pour tous  $x \in X$ . Si  $\mathbb{R}$  est identifiée avec l'ensemble de fonctions  $\{1, \dots, n\} \rightarrow \mathbb{R}$ , alors ceci se spécialise au produit d'ordre sur les vecteurs. Pour réduire l'écriture, nous emploierons la convention vecteur-scalaire suivante: si, dans une opération ou une relation binaire, un vecteur et une grandeur scalaire sont mélangés, l'opération appropriée est effectuée, ou la relation appropriée est exigée pour se tenir, sur chaque composant du vecteur. Par exemple, si  $h \in \mathbb{R}$  and  $\vec{x} \in \mathbb{R}$ , then  $\vec{x}+h$  dénotera le vecteur  $(x_1 +h, \dots, X_n +h)$ , et  $\vec{x} \leq h$  va impliquer  $x_i \leq h$  pour chaque  $1 \leq i \leq n$ . Dans ce qui suit, nous emploieront  $h$  pour dénoter un vrai nombre sans l'indiquer explicitement. Les formules telles que le  $\vec{x}=h$  devraient donc toujours être interprétés en utilisant la convention vecteur-scalaire:  $x_i = h$  pour chaque  $1 \leq i \leq n$ .

**Définition 21** Une fonction (*min, max, +*) du type  $(n,1)$  est n'importe quelle fonction, qui peut être écrite comme limite dans la grammaire suivante:

$$f := x_1, x_2, \dots, x_n | f + a | f \wedge f | f \vee f \quad \text{avec } a \in \mathbb{R} \quad (2)$$

La notation utilisée ici est la forme familière de Backus-Naur en informatique. Les barres verticales séparent les différentes manières dont des termes peuvent être récursivement construites. La limite la plus simple est l'un des  $n$  variables  $x_i$ , considérée comme la  $i^{eme}$  composante de la fonction. Ayant n'importe quel terme, un autre peut être construit en ajoutant des  $a \in \mathbb{R}$ ; ayant deux termes, un nouveau peut être construit en prenant une plus grande borne inférieure ou la plus petite borne supérieure. Seulement ces règles peuvent être employées pour établir des termes. Parmi ces trois termes:

$$(((x_1 + 2) \vee (x_2 - 0.2)) \wedge x_3) \vee (x_2 + 3.5) - 1$$

$$x_1 \vee 2$$

$$(x_1 + x_2) \wedge (x_3 + 1)$$

Le premier est une fonction (*min, max, +*) mais ni le deuxième ni le troisième ne peut être produit par la grammaire.



Nous supposons que  $+$  a une priorité plus élevée que  $\vee$  ou  $\wedge$ , nous permettant d'écrire plus simplement l'exemple comme :

$$(((x_1 + 2 \vee x_2 - 0.2)) \wedge x_3) \vee x_2 + 3.5) - 1$$

Bien que la grammaire fournisse une syntaxe commode pour l'écriture des termes, nous en sommes intéressés seulement en tant que fonctions  $\mathbb{R}^n \rightarrow \mathbb{R}$ . Des termes peuvent donc être réarrangés en utilisant l'associativité et la distributivité des opérations de treillis, aussi bien que le fait que l'addition distribue  $\wedge$  et  $\vee$  et l'exemple précédent devient :

$$(x_1 + 1 \vee x_2 - 2.5) \wedge (x_3 - 1 \vee x_2 + 2.5)$$

**Définition 22** Une fonction  $(\min, \max, +)$  du type  $(n, m)$  est n'importe quelle fonction  $F: \mathbb{R}^n \rightarrow \mathbb{R}^m$ , tel que chaque composant  $F_i$  est une fonction  $(\min, \max, +)$  du type  $(n, 1)$ .

L'ensemble de fonctions  $(\min, \max, +)$  du type  $(n, m)$  sera dénoté  $MM(n, m)$ . Nous serons la plupart du temps concernés par des fonctions du type  $(n, n)$ , qui réfèrent aux fonctions de dimension  $n$ .

Il est commode de distinguer quelques cas particuliers. Soit  $f \in MM(n, 1)$ , Si  $f$  peut être représenté par un terme qui n'utilise pas  $\wedge$ , elle serait  $(\max, +)$ . Si  $f$  peut être représenté par un terme qui n'emploie pas  $\vee$ , c'est une fonction  $(\min, +)$ . Si  $f$  est  $(\max, +)$  et  $(\min, +)$ , elle est simple. La même terminologie est prolongée aux fonctions  $F \in MM(n, m)$  en exigeant que chaque composant  $F_i$  a la propriété en question. Si  $F \in MM(n, m)$  et chaque  $F_i$  est max-seulement ou min-seulement,  $F$  sera appelée séparée.

Parmi les fonctions suivantes en  $MM(2, 2)$   $S$  et  $T$  sont simples et  $U$  est  $(\max, +)$ . D'ailleurs,  $S$  est  $(\min, +)$  et est séparé.

$$S = \begin{pmatrix} x_1 + 1 \\ x_2 - 1 \end{pmatrix} \quad T = \begin{pmatrix} x_2 + 1 \\ x_2 - 1 \end{pmatrix} \quad U = \begin{pmatrix} x_1 + 1 \vee x_2 + 1 \\ x_1 + 2 \end{pmatrix}$$

**Proposition 3** Soit  $F, G \in MM(n, n)$  et  $\vec{a}, \vec{x}, \vec{y} \in \mathbb{R}^n$  et  $h \in \mathbb{R}$ . On a:

- ♠  $F + \vec{a}, FG, F \wedge G$  et  $F \vee G \in MM(n, n)$ .
- ♠ Homogénéité:  $F(\vec{x} + h) = F(\vec{x}) + h$ .
- ♠ Monotonie: si  $\vec{x} \leq \vec{y}$  alors  $F(\vec{x}) \leq F(\vec{y})$ .
- ♠ Nonexpansive:  $\|F(\vec{x}) - F(\vec{y})\| \leq \|\vec{x} - \vec{y}\|$

**Proposition 4** Si  $F: \mathbb{R}^n \rightarrow \mathbb{R}^n$  satisfait  $H$  alors  $M$  est équivalent à  $N$ .

La propriété de l'homogénéité exige une généralisation de la notion conventionnelle du point fixe ou périodique de la fonction.

**Définition 23** Supposons que  $F: \mathbb{R}^n \rightarrow \mathbb{R}^n$  satisfait la propriété  $H$ . On dit que

- ♠  $x \in \mathbb{R}^n$  est un point fixe de  $F: \mathbb{R}^n \rightarrow \mathbb{R}^n$ , si  $F(\vec{x}) = \vec{x} + h$  pour un  $h \in \mathbb{R}$ .
- ♠  $\vec{x}$  est un point périodique de  $F$  ayant  $p$  comme période, si  $\vec{x}$  est un point fixe de  $F^p$  mais pas de  $F^k$  pour tout  $0 \leq k \leq p$ .

### 3.2.1 Fonctions min max et Fonctions topicales :

Un système à événements discrets est, en général, un système comportant un ensemble fini d'événements qui se produisent à plusieurs reprises, un circuit numérique, dans lequel un événement pourrait être un changement binaires de tension sur un fil, de 1 à 0 ou vice versa; un système informatique réparti, dans lequel un événement pourrait être l'arrivée d'un paquet de messages à un ordinateur; une usine automatisée, dans laquelle un événement pourrait être l'accomplissement d'un travail sur une machine.

Les systèmes à événements discrets sont omniprésents dans la vie moderne et sont le centre de beaucoup d'intérêt pour des domaines de technologie [1, 7, 15, 25]. Ils sont les systèmes dynamiques, dans le sens qu'ils évoluent dans le temps, mais leur analyse mène aux mathématiques tout à fait différentes à celles employée pour modeler le comportement dynamique dans les systèmes continus et différentiable.

Si  $n$  est le nombre d'événements dans le système, on a alors  $\vec{x} \in \mathbb{R}$  tel que  $x_i$  soit le temp de la première occurrence de l'événement  $i$ , relativement à une certaine origine arbitraire de moment où le système est commencé. Supposons plus loin que le système peut être modelé de telle manière que, pour une certaine fonction  $F : \mathbb{R} \rightarrow \mathbb{R}$ ,  $F_i(\vec{x})$  donne le temp de la prochaine occurrence de l'événement  $i$ . Dans ce cas le comportement dynamique du système peut être modélisé par le système dynamique discret  $F$ .

On pourrait croire qu'un modèle de cette sorte est trop simplifié pour se produire dans la pratique. Ce n'est pas le cas. Par exemple, le problème de la vérification de programme d'horloge dans des circuits numériques mène directement à un tel modèle. À chaque circuit synchrone peut être associé un élément  $F \in \text{MM}(n,n)$ . Nous soulignons, cependant, l'importance de savoir quand les points fixes de fonctions  $(\min, \max, +)$  existent et comment les calculer quand c'est le cas. Les axiomes de l'homogénéité et de la monotonie ont une interprétation intéressante dans le contexte présenté ci-dessus.

L'homogénéité est équivalente à dire que l'origine de la référence globale du temps est non pertinente. La Monotonie affirme que donner un délai à quelques événements ne peut accélérer aucune partie du système. Cette dernière condition est intuitivement raisonnable et est souvent observée dans la pratique.

**Définition 24 Fonctions topicales** *C'est n'importe quelle fonction  $F:R^n \rightarrow R^n$  satisfaisant les propriétés  $H$  et  $M$ , [23].*

Quel rôle les fonctions  $(\min, \max, +)$  jouent-elles dans la classe plus grande des fonctions topicales? Il s'avère qu'elles ont un rôle capital, comme le montre l'observation suivante de Gunawardena, de Keane et de Moineau.

**Lemme 3.1 :** Soit  $T:R^n \rightarrow R^n$  une fonction topicale et soit  $S \subseteq \mathbb{R}$  un ensemble de points fini. Il existe  $H \in \text{MM}(n,n)$  tels que  $T \leq H$  et  $T(\vec{u}) = H(\vec{u})$  pour tout  $\vec{u} \in S$ .

Il en découle que les fonctions topicales sont une approximation des fonctions  $(\min, \max, +)$  : n'importe quelle fonction topicale est l'enveloppe inférieure d'une famille des fonctions  $(\min, \max, +)$ .

D'une manière primordiale, cette approximation préserve quelques aspects de la dynamique. En utilisant la notation du lemme 1,1, il découle de la propriété M que  $T^k \leq H^k$ . En particulier, le vecteur de temps de cycle de T sera lié à celui de H ( sous entendu que tous les deux existent). Il découle également du lemme 1,1 que chaque orbite périodique d'une fonction topicale est l'orbite d'une certaine fonction (min, max, +).

Les fonctions topicales sont une tentative d'un modèle mathématique pour les systèmes d'événement discrets. Cependant, ils ont également l'intérêt mathématique intrinsèque parce qu'ils incluent un certain nombre d'exemples classiques qui ont été intensivement étudiés dans des contextes tout à fait différents.

### 3.2.2 Fonctions topicales et temps de cycle :

Soit A une matrice  $n \times n$  avec des éléments non-négatives. Un élément de  $\mathbb{R}^n$  peut être vu comme un vecteur colonne. On supposera plus loin qu'aucune ligne de A ne soit égale à zéro.

$$\forall 1 \leq i \leq n, \exists 1 \leq j \leq n, \text{tel que } A_{ij} \neq 0 \quad (3)$$

La dynamique des fonctions topicales inclue comme cas spécial les matrices non négatives. On peut montrer qu'un certain nombre d'exemples classiques dans la commande optimale, la théorie des jeux rectangulaires et les sciences économiques mathématiques provoquent également des fonctions topicales. La géographie de l'espace des fonctions topicales est détaillé dans [12, 24].

Une demande fréquente des concepteurs de systèmes est d'estimer la performance. Si le système peut être modélé par une fonction simple,  $F: \mathbb{R}^n \rightarrow \mathbb{R}^n$ , comme décrit ci-dessus, une évaluation peut être faite sur la base du temps écoulé entre les occurrences successives:  $F(\vec{x}) - \vec{x}$  et la moyenne sur plusieurs occurrences.

$$(F^k(\vec{x}) - F^{k-1}(\vec{x}) + \dots + F(\vec{x}) - \vec{x})/k$$

$\lim_{k \rightarrow \infty} /k$  est un vecteur quantité qui mesure la latence moyenne asymptotique dans chaque composante. Cette limite existe-t-elle?

**Lemme 3.2 :** Soit  $F: \mathbb{R}^n \rightarrow \mathbb{R}^n$  satisfaisant la propriété N. Si  $\lim_{k \rightarrow \infty} F^k(\vec{x}) / k$  existe quelque part, alors il existe partout et a la même valeur.

**Définition 25** Soit  $F: \mathbb{R}^n \rightarrow \mathbb{R}^n$  satisfaisant la propriété N. Le vecteur de temps de cycle de F, dénoté  $\chi(f) \in \mathbb{R}$ , est défini par :

$$\lim_{k \rightarrow \infty} F^k(\vec{x})/k \quad (4)$$

Quand cette limite existe pour un certain  $\vec{x} \in \mathbb{R}$ , elle est définie ailleurs.

Le vecteur temp de cycle rapporte immédiatement une condition nécessaire pour un point fixe. Supposons que F est une fonction topicale avec un point fixe, de sorte que  $F(\vec{x}) = \vec{x} + h$ , par application répétée de H, on remarque que  $F^k(\vec{x}) = \vec{x} + k h$ ; par conséquent,  $\chi$  existe

et  $\chi(f) = h$ . Par la convention vecteur-scalaire ceci signifie que chaque composant de  $\chi(f)$  a la même valeur  $h$ , qui, par Lemma 1,2, est indépendante du choix du point fixe. Comme discuté dans l'introduction, le résultat inverse, (1), tient également quand  $F$  est une fonction (min, max, +).

Toutes les fonctions topicales n'ont pas des durées de cycle. Le problème consistant à caractériser ceux qui en ont reste un problème important ouvert. En outre, on ne peut pas s'attendre à un théorème de point fixe si puissant tel que (1) même pour ces fonctions topicales qui ont des durées de cycle.

### 3.2.3 Fonctions max-plus, temps de cycle et projecteur spectral :

Il devrait être clair des remarques précédentes que si  $f \in MM(n,1)$  est une fonction max-plus, alors elle peut être réduite à la forme

$$f = x_1 + a_1 \vee \dots \vee x_n + a_n$$

où l'absence d'un terme  $x_j + a_j$  est indiquée par le l'instanciation  $j = -\infty$ . Si  $F \in MM(n,1)$  est max plus, elle peut donc être représentés par une matrice  $A$  avec des entrées dans  $\mathbb{R} \cup \{-\infty\}$ :

$$\begin{aligned} F &= x_1 + A_{11} \vee \dots \vee x_n + A_{1n} \\ &\quad \vdots \\ &\quad \vdots \\ F &= x_1 + A_{n1} \vee \dots \vee x_n + A_{nn} \end{aligned} \tag{5}$$

Puisque chaque composant de  $F$  doit avoir une certaine valeur,  $A$  satisfait le propriété de non dégénérescence formellement semblable à (3) dans  $\mathbb{R} \cup \{-\infty\}$ :

$$\forall 1 \leq i \leq n, \exists 1 \leq j \leq n, \text{ tels que } A_{ij} \neq -\infty$$

Supposons que les opérations algébriques sur  $\mathbb{R} \cup \{-\infty\}$  sont maintenant définies, l'opération de somme devient donc le maximum et l'opération de multiplication devient l'addition. L'élément  $-\infty$  devient alors le zéro de la somme et 0 devient l'élément neutre de la multiplication. Si des vecteurs dans  $\mathbb{R}_{max}$  sont considérés comme des vecteurs de colonne,  $F$  peut être réécrite comme équation de matrices :

$$F(\vec{x}) = A\vec{x}$$

dans laquelle les opérations de matrice sont interprétées dans  $R_{max}$ . Il suit que  $F^k(\vec{x}) = A^k \vec{x}$ . Nous n'adopterons pas la notation max-plus. C'est-à-dire,  $+$  et  $\times$  auront toujours leurs significations usuelles. Nous utiliserons  $\vee$  et  $+$  pour la notation des opérations max-plus. De même, 0 aura toujours son signification usuelle et nous emploierons  $-\infty$  pour le zéro de  $R_{max}$ . si  $A$  et  $B$  sont deux matrices de dimensions respectivement  $n \times p$  et  $p \times m$  sur  $R_{max}$ , alors  $AB$  sera la matrice produit dans  $R_{max}$  tels que:

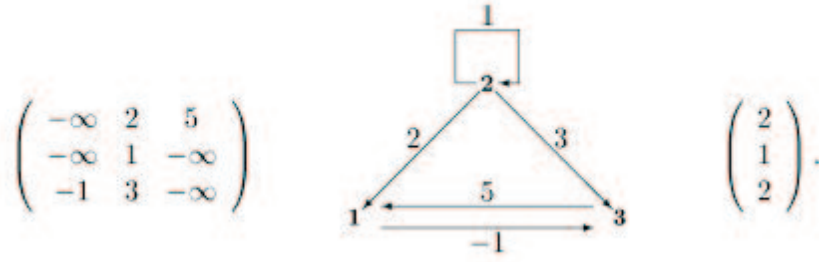


FIG. 3.1 – Matrice max-plus, Graphe de precedance et Vecteur  $\mu$

$$(AB)_{ij} = \bigvee_{1 \leq k \leq p} A_{ik} + A_{kj}.$$

Nous rappelons divers résultats standards dans la théorie max-plus.

**Définition 26 (Chemin, circuit)**

- ♠ Un chemin de  $i_1$  à  $i_m$  est une suite de noeuds  $i_1, \dots, i_m$  telle que  $1 < m$  et  $(i_j, i_{j+1})$  pour  $1 \leq j < m$ .
- ♠ Un circuit est un chemin qui commence et termine sur le même noeud:  $i_1 = i_m$ .
- ♠ Un circuit est élémentaire si les noeuds  $i_1, \dots, i_m$  sont tous distincts .
- ♠ Un noeud  $i$  est en amont de  $j$ , et noté  $j \Leftarrow i$ , si  $i = j$  ou bien s'il y a un chemin dans  $G$  de  $i$  à  $j$ . (Un noeud est toujours en amont de lui-même).
- ♠ Un circuit  $g$  est en amont du noeud  $i$ , et on note  $i \Leftarrow g$ , si un noeud du circuit est en amont de  $i$ .

**Définition 27 (Poids et longueur d'un chemin)**

- ♠ Le poids d'un chemin  $p$ ,  $|p|_w$ , est la somme des valeurs des arcs du chemin :

$$|p|_w = \sum_{j=1}^{m-1} A_{i_j i_{j+1}}$$

la multiplication des matrices à une interprétation en termes de poids de chemins :  $A_{ij}^s$  est le poids maximum sur tout les chemins de longueur  $s$  de  $j$  à  $i$ .

- ♠ La longueur d'un chemin,  $|p|_l$ , est le nombre d'arcs du chemin :  $|p|_l = m-1$ .

**Définition 28 (Poids moyen maximal)** Si  $g$  est un circuit, son poids moyen, noté  $m(g)$  est défini par

$$m(g) = |p|_w / |p|_l.$$

Si  $A$  une matrice  $n \times n$  sur  $\mathbb{R}_{max}$ , soit  $\mu(A) \in \mathbb{R}_{max}^n$  défini par

$$\mu_i(A) = \max\{m(g) / i \Leftarrow g\}$$

**Définition 29 (fonction t,b:  $\mathbb{R}^n \rightarrow \mathbb{R}$ )**

$$t(x) = x_1 \vee \dots \vee x_n$$

$$b(x) = x_1 \wedge \dots \wedge x_n$$

$t(\mu(A))$  est le poids moyen maximum sur tous les circuits.

**Définition 30 (Circuit critique, classe critique)**

- ♠ Un circuit critique est un circuit élémentaire de poids moyen  $t(\mu(A))$ .
- ♠ On appelle classe critique et on note  $C(A) \subseteq \{1 \dots n\}$  l'ensemble des noeuds de  $G(A)$  appartenant à un circuit critique quelconque.

**Exemple 16** La fonction max plus:

$$\begin{aligned} F_1(x_1, x_2, x_3) &= x_2 + 2 \vee x_3 + 5 \\ F_2(x_1, x_2, x_3) &= x_2 + 1 \\ F_3(x_1, x_2, x_3) &= x_1 - 1 \vee x_2 + 3 \end{aligned}$$

admet la matrice max-plus, le graphe de précédence et le vecteur  $\mu$  de la figure 3.1.

Le temps de cycle moyen maximal est 2 et  $1 \leftarrow 3 \leftarrow 1$  est l'unique circuit critique.

**Proposition 5** Si  $F \in MM(n, n)$  est une fonction  $(\max, +)$  et  $A$  la matrice qui lui est associée sur  $\mathbb{R}_{max}$  alors  $\chi(F)$  existe et  $\chi(F) = \mu(A)$ .

**Preuve:** voir [1].

Si  $F$  admet un point fixe, de sorte que  $F(\vec{x}) = \vec{x} + h$ , alors  $h = \mu(A)$ . En particulier,  $h = t \mu(A)$ , le temps de cycle moyen maximum dans tous les circuits dans  $G(A)$ . C'est la valeur propre associée à n'importe quel vecteur propre de  $A$  se situant dans  $\mathbb{R}^n$ . C'est l'analogue pour les max-plus matrices de la racine de Perron, ou rayon spectral pour les matrices non négatives.

Supposons que  $A$  est une matrice  $n \times n$  définie sur  $\mathbb{R}_{max}$ . Supposez plus loin que  $t \mu(A) = 0$ , de sorte que tous les circuits de  $G(A)$  aient le poids non-positifs. Puisque tout chemin  $p$  dans  $G(A)$ , avec  $|p|_l \geq n$ , doit contenir un circuit, ce n'est pas difficile de voir que

$$(A^s)_{ij} \leq A_{ij} \vee \dots \vee (A^n)_{ij} \quad \text{pour tout } s \geq n$$

Soit  $(A^+)_{ij} = \sup \{ (A^s)_{ij} \mid 1 \leq s \}$ , qui est bien défini comme élément de  $\mathbb{R}_{max}$  par l'observation précédente. Notez qu'il est encore possible que  $(A^+) = -\infty$ , à partir du moment où il n'existe pas de chemin de  $j$  vers  $i$ . Soit  $C(A) \subseteq \{1, \dots, n\}$  l'ensemble des noeuds de  $G(A)$  qui font partie d'un circuit critique. Soit  $P(A)$  la matrice dans  $\mathbb{R}_{max}$  définie comme suit :

$$(P_A)_{ij} = \bigvee_{u \in C(A)} (A^+)_{iu} + (A^+)_{uj}.$$

$P_A$  est appelé projecteur spectral. La partie 3 du lemme 3.4 montrera qu'il contient des informations sur les vecteurs propres de  $A$ . Le lemme suivant est un résultat standard dans la théorie des matrice max-plus.

**Lemme 3.3 :** supposons que  $A$  est une matrice  $n \times n$  dans  $\mathbb{R}_{max}$  tels que  $t \mu(A) = 0$ . Alors

$$(P_A)A = A(P_A) = P_A \text{ et } (P_A)^2 = P_A$$

Le prochain lemme rassemble un certain nombre d'observations utiles. Certains d'entre elles sont bien connus dans la théorie max-plus.

**Lemme 3.4 :** Soit  $A$  une matrice  $n \times n$  dans  $\mathbb{R}_{max}$  tels que  $t \mu(A) = 0$ . Supposons plus loin que  $\vec{x}, \vec{y} \in (\mathbb{R}_{max})^n$ . Les propositions suivantes sont équivalentes :

- ♠ Si  $A\vec{x} \leq \vec{x}$  alors  $P_A \vec{x} \leq \vec{x}$ .

- ♠  $A\vec{x} = \vec{x}$ , si et seulement si,  $P_A \vec{x} \leq$ .
- ♠ L'image de  $P_A : (\mathbb{R}_{max})^n \rightarrow (\mathbb{R}_{max})^n$  est l'espace propre de A correspondant à la valeur propre 0.
- ♠ Si  $A\vec{x} = \vec{x}$ ,  $A\vec{y} = \vec{y}$  et  $x_i = y_i \forall i \in C(A)$ .
- ♠ Si  $i \in C(A)$  alors  $(P_A)_{ii} = 0$ .
- ♠ Si  $A\vec{x} \leq \vec{x}$  alors  $(A\vec{x})_i = x_i \forall i \in C(A)$ .
- ♠ Si  $\mu(A) = 0$  alors  $P_A : \mathbb{R}^n \rightarrow \mathbb{R}^n$

**Preuve:** Voir [1].

**Proposition 6** soit  $F \in MM(n,n)$  une fonction max-plus. Le résultat de point fixe (1) se tient pour  $F$ .

**Preuve:** Voir [1].

### 3.3 Point fixe de fonctions (min, max, +) :

Le but principal de cette section est de présenter un théorème constructif de point fixe pour les fonctions (min, max, +). Nous présentons alors un algorithme pour un cas spécial important et discutons sa complexité. Nous commençons par citer les conditions qui entrent dans le résultat de point fixe.

#### 3.3.1 Rectangularité :

**Définition 31** Soit  $F \in MM(n,m)$ . Un sous-ensemble  $S \subseteq MM(n,m)$  est dit une max-représentation de  $F$  si  $S$  est un ensemble fini de fonctions max-plus, tels que  $F = \bigwedge_{H \in S} H$ .

Chaque fonction (min, max, +) a une max-représentation et une min-représentation (par dualité). Puisque nous savons les vecteurs de durée de cycle des fonctions max-plus, nous pouvons estimer celui de  $F$ , quand il existe.

Supposons que  $\chi(F)$  existe. Pour tout  $H \in S$ ,  $F \leq H$ . avec la propriété M, on a :  $\chi(F) \leq \chi(H)$ . Par conséquent,

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

Une représentation max-plus donne donc une estimation majorée pour la durée de cycle. Cette évaluation peut être employée pour développer un état alternatif pour les points fixes. La première difficulté est qu'il existe généralement plus qu'une max-représentations pour une seule fonction (min, max, +) donnée et les évaluations correspondantes peuvent différer.

**Exemple 17** La fonction (min, max, +) suivante

$$\begin{aligned} F_1(x_1, x_2, x_3) &= (x_2 + 2 \vee x_3 + 5) \wedge x_1 \\ F_2(x_1, x_2, x_3) &= x_2 + 1 \wedge x_3 + 2 \\ F_3(x_1, x_2, x_3) &= x_1 - 1 \vee x_2 + 3 \end{aligned}$$

admet les deux max représentations suivantes :

$$\begin{pmatrix} x_2 + 2 \vee x_3 + 5 \\ x_2 + 1 \\ x_1 - 1 \vee x_2 + 3 \end{pmatrix} \begin{pmatrix} x_1 \\ x_3 - 2 \\ x_1 - 1 \vee x_2 + 3 \end{pmatrix}$$

et

$$\begin{pmatrix} x_2 + 2 \vee x_3 + 5 \\ x_3 - 2 \\ x_1 - 1 \vee x_2 + 3 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 + 1 \\ x_1 - 1 \vee x_2 + 3 \end{pmatrix}$$

Les vecteurs temps de cycle calculés des fonctions max-plus sont respectivement :

$$\begin{pmatrix} 2 \\ 1 \\ 2 \end{pmatrix}, \begin{pmatrix} 0 \\ 2.5 \\ 2.5 \end{pmatrix} \text{ et } \begin{pmatrix} 2 \\ 2.5 \\ 2.5 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix}$$

Ce qui fait que l'évaluation de l'approximation donne, pour la première max-représentation,  $(0, 1, 2)^T$ , tandis que pour la seconde,  $(0, 1, 1)^T$ .

Pour obtenir la meilleure estimation, l'information dans toutes les max-représentations de  $F$  doit être utilisée. On observe que l'ensemble de fonctions  $(\min, \max, +)$  de  $MM(n, m)$  a une représentation en tant qu'un produit cartésien:  $MM(n, m) = MM(n, 1) \times \dots \times MM(n, 1)$ . Si  $S \subseteq A_1 \times \dots \times A_m$  est un sous-ensemble d'un tel produit cartésien, soit  $\pi_i(S) \subseteq A_i$  qui dénote sa projection sur son  $i^{\text{me}}$  facteur.

### Définition 32 (Rectangularisation)

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

$$Rec(S) = \pi_1(S) \times \dots \times \pi_m(S)$$

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

C'est, naturellement, toujours le cas que  $S \subseteq Rec(S)$ . Il est également clair que  $\pi_i(S) = \pi_i(Rec(S))$ . Suite à ça, si  $S \subseteq MM(n, m)$  est fini, alors :

$$\bigwedge_{H \in S} H = \bigwedge_{H \in Rec(S)} H \quad \text{et} \quad \bigvee_{H \in S} H = \bigvee_{H \in Rec(S)} H,$$

Fait en outre, si  $S$  contient seulement les fonctions  $(\max, +)$ , alors ainsi est  $Rec(S)$ . Il vaut la peine d'observer que ni l'un ni l'autre des max-représentations utilisées ci-dessus n'était rectangulaire.

Supposons que  $S \subseteq P$ , où  $(P, \leq)$  est un ensemble partiellement commandé. Notons par  $Min(S)$  le sous-ensemble de plus petits éléments de  $S$ ,

$$Min(S) = \{x \in S \mid y \in S, y \leq x \Rightarrow y = x\}$$

et par  $Max(S)$  l'ensemble correspondant des plus grands éléments. Si  $S$  est fini et  $x \in S$ , alors là existent  $u \in Min(S)$  et  $v \in Max(S)$  tels que  $u \leq x \leq v$ .



Supposons maintenant que  $P$  est un ordre partiel de produit:  $P = A_1 \times \dots \times A_m$ , avec l'ordre partiel sur  $P$ .

**lemme 3.5** Soit  $S_i \subseteq A_i$  le sous-ensemble fini pour  $1 \leq i \leq m$ . on a

$$\text{Min}(S_1 \times \dots \times S_m) = \text{Min}(S_1) \times \dots \times \text{Min}(S_m).$$

**Preuve:** Voir [1].

**Théorème 3** Soit  $F \in \text{MM}(n,m)$  et supposons que  $S, T \subseteq \text{MM}(n,m)$  sont deux max-représentations rectangulaires de  $F$ . on a donc  $\text{Min}(S) = \text{Min}(T)$ .

**Corollaire 3.1** Soit  $F \in \text{MM}(n,n)$  et supposons que  $S, T \subseteq \text{MM}(n,m)$  sont deux max-représentations rectangulaires de  $F$ . Alors

$$\bigwedge_{H \in S} \chi(H) = \bigwedge_{G \in T} \chi(G).$$

Les max-représentations utilisées par exemple (17) ont des rectangularisation identiques, obtenus en prenant l'union des deux représentations. Le meilleur bilan estimatif pour la durée de cycle de  $F$ , sur la base du corollaire 3.1, est donc  $\chi(F) \leq (0,1,1)$ .

Supposons que  $F \in \text{MM}(n,n)$ . soit  $S, T \in \text{MM}(n,n)$  deux représentations max et min rectangulaires, respectivement, de  $F$ . Si  $G \in T$  et  $H \in S$  on a alors clairement  $G \leq H$  et ainsi  $\chi(G) \leq \chi(H)$ . par le suite :

$$\bigvee_{G \in T} b\chi(G) \leq \bigvee_{G \in T} \chi(G) \leq \bigwedge_{H \in S} \chi(H) \leq \bigwedge_{H \in S} t\chi(H)$$

**Proposition 7** Supposons que  $S$  et  $T$  sont deux représentations max et min rectangulaires, respectivement, de  $F$  et supposons en outre, que  $F$  ait un point fixe, tels que  $F(\vec{x}) = \vec{x} + h$ . L'inégalité précédente s'effondre alors à une égalité.

Nous remarquons que l'égalité des termes est une condition nécessaire pour que  $F$  ait un point fixe. Il est aussi montré que c'est également une condition suffisante [19]. Il y a deux difficultés avec ce résultat. Premièrement, il est nonconstructif, ce qui est un handicap important dans les applications des fonctions (min, max, +). Deuxièmement, il exige l'information d'une représentation max et d'une représentation min de  $F$ , il est difficile d'obtenir un résultat pour n'importe quelle fonction donnée.

La proposition précédente montre également que la condition  $\bigwedge_{H \in S} (H) = h$ , et son dual, sont les deux nécessaires pour que  $F$  ait un point fixe. Ce résultat de point fixe présenté par le théorème ci-dessous, résous ces deux difficultés mentionnés.

### 3.3.2 Théorème de point fixe :

Il sera commode, dorénavant, de laisser tomber n'importe quelle distinction d'écriture entre les fonctions (max,+) ou (min,+) et leurs matrices associées. Si  $A$  est une fonction (max,+)

ou  $(\min, +)$ , nous emploierons le même symbole pour désigner sa matrice  $\mathbb{R}_{max}$  ou  $\mathbb{R}_{min}$  associée. En outre, nous emploierons  $\chi(A)$  au lieu de  $\mu(A)$ .

Si  $U$  et  $V$  sont deux ensembles, soit  $U \setminus V$  la notation du complément de  $V$  dans  $U$ :  $U \setminus V = \{i \in U \mid i \notin V\}$ . Le prochain résultat est le lemme technique principal de cette section.

**Lemme 3.6** supposons que  $F \in MM(n,n)$  et que  $S \subseteq MM(n,n)$  est une max-représentation rectangulaire de  $F$ . Pour n'importe quelle famille de  $n$  fonctions dans  $S$ :  $A_1, \dots, A_n \in S$ . Il existe une fonction  $K \in S$  tels que  $t\chi(k) = \bigvee_{1 \leq i \leq n} \chi_i(A_i)$ . **Preuve:** Voir [1].

Le lemme 2.6 a un certain nombre de conséquences utiles que nous rassemblons dans les lemmes suivants.

**Lemme 3.7** Dans les mêmes conditions que le lemme 2.6, la fonction  $\{1, \dots, n\} \times S \rightarrow \mathbb{R}$ :  $(i, H) \rightarrow \chi_i(H)$  admet un point de selle:

$$t\left(\bigwedge_{H \in S} \chi(H)\right) = \bigwedge_{H \in S} t\chi(H)$$

**Lemme 3.8** Dans les mêmes conditions que le lemme 2.6, si  $\bigwedge_{H \in S} \chi(H) = h$ , il existe  $K \in S$  tels que  $\chi(K) = h$ .

Le prochain résultat est le théorème principal de cette section. Il suit en détail un argument donné par Cochet-Terrasson et Gaubert [6]. L'ingrédient additionnel qui apparaît ici est le lemme 2.6, sous la couverture du lemme 2.4, qui permet à un résultat plus fort d'être dérivé [6].

La preuve est basée sur une analogie  $(\min, \max, +)$  de l'algorithme d'amélioration de la politique de Howard pour des problèmes stochastiques de contrôle [41, 35]. Typiquement, l'algorithme de Howard trouve un point fixe de  $F(\vec{x}) = \bigwedge_{u \in U} \vec{C}_u + P_u \vec{x}$  où  $U$  est un ensemble fini et, pour tout  $u \in U$ ,  $\vec{C}_u \in \mathbb{R}^n$  est un vecteur coût et  $P_u$  est une matrice stochastique. Nous remarquons, en utilisant la proposition 1.1 que les fonctions de cette forme sont en fait topicales. À chaque étape, l'algorithme de Howard choisit une fonction  $A(\vec{x}) = \vec{c} + P\vec{x}$  dans  $S = \text{Rec } \vec{u} + \vec{x} \mid u \in U$  et lui trouve un point fixe. Il est nécessaire de supposer qu'un tel point fixe peut être trouvé, ce qui est le cas où chaque matrice  $P_u$  est positive. Si ce point n'est pas également un point fixe de  $F$ , alors la fonction  $A$  est remplacée par  $A' \in S$  qui satisfait  $F(\vec{x}) = A'(\vec{x})$  et le processus est répété.

Dans des conditions appropriées on peut prouver que ceci mène, après un nombre fini d'itérations, à un point fixe de  $F$ .

**Théorème 4** Soit  $F \in MM(n,n)$  et supposons que  $S, T \in MM(n,n)$  sont rectangulaire et, respectivement, une max-représentation et une min-représentation de  $F$ . Les conditions suivantes sont équivalentes.

- ♠  $F$  admet un point fixe, avec :  $F(\vec{x}) = \vec{x} + h$
- ♠  $\bigwedge_{H \in S} \chi H = h$ .
- ♠  $\bigvee_{G \in T} \chi G = h$ .

**Preuve:** Voir [1]

**Corollaire 3.2** Si  $F \in \text{MM}(n,n)$  satisfait la conjoncture de la dualité, alors (1) est vérifiée.

**Corollaire 3.3** Supposons que  $F \in \text{MM}(n,n)$  admet une max-représentation  $S$  tel que chaque  $H \in S$  admette un point fixe, alors en  $F$  admet un aussi.

**Corollaire 3.4** Soit  $F \in \text{MM}(n,n)$  et  $S, T \in \text{MM}(n,n)$  sont rectangulaire et, respectivement, une max-représentation et une min-représentation de  $F$ .  $F$  admet un point fixe, avec  $F(\vec{x}) = \vec{x} + h$ , si, et seulement si,

$$\bigvee_{G \in T} b\chi(G) = h = \bigwedge_{H \in S} t\chi(H)$$

## 3.4 Problèmes Algorithmiques :

La recherche de points fixes des fonctions  $(\min, \max, +)$  est un problème d'application très important. Par exemple, le problème de vérification de programme d'horloge mentionné ci dessus est équivalent à trouver un point fixe d'une fonction  $(\min, \max, +)$  associée à un circuit numérique. La forme particulière des fonctions  $(\min, \max, +)$  qui surgissent dans cette application permet d'obtenir des algorithmes efficaces de recherche de points fixes. Bien que les théories de la section précédente soient constructives de nature, elles ne provoquent pas un algorithme général et efficace.

### 3.4.1 Technique de construction de point fixe :

Le problème provient du fait qu'une fonction  $(\min, \max, +)$  est typiquement présentée sous la forme  $F = \bigwedge_{H \in S} H$  où  $S$  est un sous-ensemble de fonctions max-plus qui n'est pas nécessairement rectangulaire. Afin de se servir de la méthode dans le théorème 2,2, il est nécessaire de trouver  $A \in \text{Rec}(S)$ , tel qu' $A$  admet un point fixe et  $\chi(A)$  soit minimal; c'est le point de départ pour l'itération. La recherche dans tout l'espace de  $\text{Rec}(S)$  pour trouver une telle fonction est prohibitivement chère. Cependant, c'est parfois le cas que toutes les fonctions  $H \in \text{Rec}(S)$  ont des points fixes.

Cependant, quand  $S$  se compose des fonctions pour lesquelles les matrices  $(\max, +)$  correspondantes n'ont aucune entrée égale à  $-\infty$ . Il est évident, par référence à la proposition 1.3 que chaque fonction  $H \in \text{Rec}(S)$  satisfasse  $\chi(H) = h$ . Par conséquent, à partir de la Proposition 1,4, chaque  $H$  admet un point fixe. Nous pouvons adapter la méthode de théorème 2,2 pour donner un algorithme réalisable dans ce cas-ci. Il sera commode de prolonger la notation de projecteur spectrale  $P_A$  aux matrices générales: si  $t\mu(A) \neq 0$ , soit  $\tilde{A} = -t\mu(A) + A$ , tels que  $t\mu(\tilde{A}) = 0$ .

Supposons qu'une fonction  $(\min, \max, +)$   $F$  est donnée sous la forme :

$$F_i(\vec{x}) = \bigwedge_{u \in U(i)} A_{iu}\vec{x},$$

où  $U(1), \dots, U(n)$  sont les ensembles finis et  $A_{iu}$  sont des vecteurs avec des entrées dans  $\mathbb{R}_{max}$ . Empruntant le vocabulaire de la commande optimale, nous disons qu'une politique est un

graphe  $\pi \{1, \dots, n\} \rightarrow \bigcup_{1 \leq i \leq n} U(i)$ , tel que  $\pi(i) \in U(i) \forall 1 \leq i \leq n$ . Par construction, l'ensemble de matrices de politique  $A[\pi]$  est rectangulaire.

L'algorithme de point fixe prend comme entrée une fonction (min, max, +) de la forme présentée précédemment, dans laquelle chacune des matrices de politique admet un point fixe. Autrement dit, par la proposition 1,4, pour chaque matrice de politique  $\pi$ , il existe  $h_\pi \in \mathbb{R}^n$  tels que  $\chi(A[\pi]) = h_\pi$ . L'algorithme produit comme sortie  $\vec{x} \in \mathbb{R}^n$  et  $h \in \mathbb{R}^n$  tels que  $F(\vec{x}) = \vec{x} + h$ .

### 3.4.2 Algorithme 3.1:

**1 - Initialisation.** Choisir une politique arbitraire  $\pi_1$ . Initialiser  $s=1$  et  $A_1 = A[\pi_1]$ . Trouver  $\vec{x}_1 \in \mathbb{R}^n$ , tels que  $A_1 \vec{x}_1 = \vec{x}_1 + h_1$

**2 -** Si  $F(\vec{x}_s) = \vec{x}_s + h_s$ , alors ARRET;

**3 -** Amélioration de la politique :

$$\forall 1 \leq i \leq n, \bigwedge_{u \in U(i)} A_{iu} \vec{x}_s = A_{i\pi_{s+1}(i)} \vec{x}_s$$

**4 -** Détermination de valeurs :

(a) Si  $\mu(A_{s+1}) < h_s$ , alors, prendre **un point fixe arbitraire**  $\vec{x}_{s+1}$  de  $A_{s+1}$ .

(b) Si  $\mu(A_{s+1}) = h_s$ , alors, prendre **le point fixe particulier**  $\vec{x}_{s+1} = P_{A_{s+1}} \vec{x}_s$ .

**5 -** Incrémenter  $s$  par 1 et recommencer par la deuxième étape.

### 3.4.3 Script Scilab :

On donne dans la suite le script Scilab commenté de l'algorithme :

```
//-----
//DEBUT
//-----
test=%F;
//-----
// Saisie de la matrice représentative de la
//fonction (Min, Max, +), matrice générale,
//comportant les coefficients du système et
//composée par des sous-matrices carrées
// représentant les fonctions (max,+).
//-----

while(test==%F)
    MM=input('MM= ');
    MM=maxplus(MM);
    [l,c]=size(MM);
```

```

//-----
//Test de la saisie de la matrice générale,
//étant composée par des matrices carrées,
//elle doit avoir des dimensions proportionnelles
// de quotient égal au nombre de fonctions max plus
// qui composent la fonction Min-Max.
//-----

test = (int(c/l)==(c/l))
end t=c/l; b=0 ;

//-----
// Initialisation de la politique initiale.
//Arbitrairement ça sera la première matrice
//carrée dans l'ordre de la saisie de
//la matrice générale.
//-----

A=MM( : ,1:l) A=maxplus(A); MM=maxplus(MM);

//-----
// Calcul selon la politique de Howard de
//l'espace propre de la politique initiale.
//-----

[h,x]=howard(A)
x=maxplus(x);
h=karp(A);

//-----
//DEBUT DES ITERATION.
//-----

while( b == 0 )

//-----
// Calcul de l'image du vecteur propre de
// la politique en cours par la fonction Min-Max.
//-----

F= %top;
for i= 0:(t-1)
    d=int(i * l);
    f=int((i+1) * l);
    F= F & (MM( : ,d+1:f)*x);
end

```

```

//-----
//Tester si le vecteur propre de la politique en
//cours est aussi un vecteur propre de la fonction Min-Max.
//-----

if F == (x * h) then
    b=1;
    disp(x,'Un point fixe de la fonction est:')
    break
else

    //-----
    //Amélioration de la politique,
    //-----

    for j= 1:l
        for i= 0:(t-1)
            d=i+1;
            f=(i+1)*l;
            if F(j) == (MM(j:j ,d+1:f) * x ) then
                if A(j:j, : ) <> MM(j:j,d+1:f) then
                    A(j:j, : ) = MM(j:j,d+1:f)
                end
            end
        end
    end
end

//-----
//Nouvelle politique
//-----

A=maxplus(A);

//-----
//Calcul de la nouvelle valeur propre.
//-----

    hh=karp(A)

if hh <> h then
    //-----
    //Choix d un point fixe arbitraire de la
    //nouvelle politique et calcul du nouveau h
    //-----

    [h,x]=howard(A)
    h=karp(A)

```

```

else
//-----
//Calcul du point fixe particulier
//-----

    [p,n]=mprojspec(A);
    p
    xx=p * x
    x=xx
end
end
end
//-----
//FIN
//-----

```

### 3.4.4 Exemple d'application

Considérons la fonction (min, max, +) suivante:

$$F_1 = (x_1 + 2 \vee x_2 \vee x_3 + 1) \wedge (x_1 - 1 \vee x_2 + 1 \vee x_3 + 1) \wedge (x_1 + 1 \vee x_2 - 1 \vee x_3 - 1) \wedge (x_1 + 2 \vee x_2 + 1 \vee x_3)$$

$$F_2 = (x_1 + 1 \vee x_2 \vee x_3 + 2) \wedge (x_1 + 2 \vee x_2 + 1 \vee x_3) \wedge (x_1 + 2 \vee x_2 \vee x_3 + 1) \wedge (x_1 + 2 \vee x_2 + 3 \vee x_3 + 3)$$

$$F_3 = (x_1 + 1 \vee x_2 + 2 \vee x_3 - 1) \wedge (x_1 + 2 \vee x_2 + 1 \vee x_3) \wedge (x_1 + 2 \vee x_2 \vee x_3 + 2) \wedge (x_1 - 1 \vee x_2 \vee x_3 + 2)$$

Afin de rendre l'algorithme plus interactif on va rendre les entrées que doit fournir l'utilisateur plus simples. Au lieu d'entrer les composantes des fonctions max plus une par une; il va fournir une seule matrice qui représente la fonction (min, max, +) en totalité. Cette matrice sera de suite traitée pour séparer les fonctions max-plus et puis leurs composantes.

Dans notre cas la matrice qui représente l'exemple est la suivante, on l'a choisi de dimensions assez considérables afin d'avoir une résolution plus explicite de l'algorithme :

$$\begin{pmatrix} 2 & 0 & 1 & -1 & 1 & 1 & 1 & -1 & -1 & 2 & 1 & 0 \\ 1 & 0 & 2 & 2 & 1 & 0 & 2 & 0 & 1 & 2 & 3 & 3 \\ 1 & 2 & -1 & 2 & 1 & 0 & 2 & 0 & 2 & -1 & 0 & 2 \end{pmatrix}$$

Traitant le cas où aucune composante n'est égale à  $+\infty$ , les matrices représentant les fonctions max-plus sont donc des matrices carrées: les dimensions de la matrice représentant la fonction (min, max, +) doivent alors être proportionnels (nombre de colonnes =  $k \times$  nombre de lignes). Le programme sépare ensuite les fonctions max plus puis leurs composantes pour commencer la construction du point fixe.

Dans notre cas, le nombre de fonctions max-plus est 4.

Initialisation : choix par défaut  $\pi_1 (1) = 1, \pi_1 (2) = 1, \pi_1 (3) = 1$  tels que :

$$A_1 = \begin{pmatrix} 2 & 0 & 1 \\ 1 & 0 & 2 \\ 1 & 2 & -1 \end{pmatrix}$$

On a  $\mu(A_1) = 2$ , alors  $h_1 = 2$  et on choisit un point fixe arbitraire de  $A_1$ , par exemple :

$$\vec{x}_1 = (2 \ 2 \ 2)^T$$

Après ce choix il s'avère que ça ne peut pas être un point fixe de la fonction F ; on procède donc par amélioration de la politique.

Ici la meilleure politique est tels que :  $\pi_2 (1) = 3, \pi_2 (2) = 2, \pi_2 (3) = 4$  tels que :

$$A_2 = \begin{pmatrix} 1 & -1 & 1 \\ 2 & 1 & 0 \\ -1 & 0 & 2 \end{pmatrix}$$

On a  $\mu(A_2) = 2$ , alors  $h_2 = 2 = h_1$  on choisit donc le point fixe particulier :

$$\vec{x}_2 = (-1 \ 0 \ 2)^T = P_{A_2} \times \vec{x}_1$$

Où

$$P_{A_2} = \begin{pmatrix} -5 & -5 & -3 \\ -4 & -4 & -2 \\ -2 & -2 & 0 \end{pmatrix}$$

Ce vecteur n'est pas non plus un vecteur point fixe de F. On procède de nouveau donc par amélioration de politique. Ici la meilleure politique est tels que :  $\pi_3 (1) = 3, \pi_3 (2) = 2, \pi_3 (3) = 2$  tel que :

$$A_3 = \begin{pmatrix} 1 & -1 & 1 \\ 2 & 1 & 0 \\ 2 & 1 & 0 \end{pmatrix}$$

On a  $\mu(A_3) = 1$ , alors  $h_3 = 2 < h_2$ . On choisit donc un point fixe arbitraire correspondant à la nouvelle politique :

$$\vec{x}_3 = (1 \ 2 \ 2)^T$$

Par vérification  $\vec{x}_3$  est bien un point fixe de la fonction F tels que  $F(\vec{x}_3) = \vec{x}_3 + h_3$ .  
L'algorithme s'arrête.



## 3.5 Exemple de généralisation :

Nous avons présenté un algorithme de calcul d'éléments spectraux d'une classe spéciale de fonctions (Min, Max, +). Des fonctions représentatives de graphes d'événements discrets fortement connexes (autrement dit ne comportant pas d'éléments égaux à  $-\infty$ ). Cependant les applications réelles sont moins évidentes que le cas traité ; autrement dit, un graphe d'événements discrets peut ne pas être fortement connexe. Nous présentons un exemple d'extension de la technique spectrale pour une classe plus large de fonctions (Min, Max, +). Nous décrivons la spécialisation à l'algèbre (max, +) de l'amélioration de la politique de Howard, qui rapporte un algorithme de calcul des solutions des problèmes spectraux appelés généralisés dans (max, +).

Nous allons présenter cette technique comme titre d'exemple de généralisation de l'outil précédent. Nous envoyons le lecteur à [X] pour les détails de cet outil.

### 3.5.1 Algorithme de calcul d'mode propre généralisé :

#### Algorithme de détermination de valeurs :

Entrées : une bonne matrice polynômiale  $A$  et une politique  $\pi$ .

Sortie : un mode propre de  $A^\pi$

#### Algorithme 4.1

1. Choisir un circuit  $c$  dans le multigraphe de  $A^\pi$

2. Calculer :

$$\bar{\eta} = \frac{\sum_{e \in c} \omega(e)}{\sum_{e \in c} \tau(e)}. \quad (19)$$

3. Choisir un noeud arbitraire  $i$  dans  $c$ , affecter  $\eta_i = \bar{\eta}$  et  $x_i$  d'une valeur arbitraire. (on choisira 0)

4. En visitant chaque noeud  $j$  ayant accès à  $i$  dans une topologie de retour arrière, initialiser :

$$\eta_j = \bar{\eta} \quad (20)$$

$$x_j = \omega(\pi(j)) - \bar{\eta} \times \tau(\pi(j)) + x_{Out(\pi(j))} \quad (21)$$

5. S'il reste un ensemble non vide  $C$  de noeuds  $j$  qui n'a pas accès à  $i$ , répéter l'algorithme en utilisant  $C \times C$ , la sous-matrice de  $A$  et la restriction de  $\pi$  à  $C$ .

#### Commentaire :

Le choix du circuit initial peut être effectué en démarrant d'un noeud arbitraire  $i$ , visiter  $\pi(i)$  et ainsi de suite jusqu'à tomber sur un noeud déjà visité. Ce qui signifie qu'on a trouvé un circuit. Applique l'équation (21) par visite rétrograde à partir du noeud choisi. La dernière étape donne un aspect récursif à l'algorithme.

### 3.5.2 Algorithme d'amélioration de politique :

Entrées : une bonne matrice polynômiale  $A$ , une politique  $\pi$ , et un mode propre généralisé  $(\eta, x)$  de  $A^\pi$ .

Sortie : une politique  $\pi'$ , tels que  $\chi(A^{\pi'}) \geq \chi(A^\pi)$ .

#### Algorithme 4.2

1. Soit

$$J = \{ i / \max_{(i,t,j) \in \varepsilon} \eta_i > \eta_j \}$$

$$K(i) = \arg \max_{(i,t,j) \in \varepsilon} \eta_i, \text{ pour } i=1, \dots, n.$$

$$I = \{ i / \max_{e=(i,t,j) \in K(i)} (\omega(e) - \tau(e) \eta_j + x_j) > x_i \}$$

$$L(i) = \arg \max_{(i,t,j) \in K(i)} (\omega(e) - \tau(e) \eta_j + x_j),$$

2. Si  $I = J = \emptyset$ ,  $(\eta, x)$  est un mode propre généralisé de  $A$  alors *ARRÊT*.

3. (a) Si  $J \neq \emptyset$ , alors :

si  $i \in J$  alors  $\pi'(i) = \forall e \in K(i)$

sinon  $\pi'(i) = \pi(i)$ .

(b) Si  $J = \emptyset$  et  $I \neq \emptyset$ , alors :

si  $i \in I$  alors  $\pi'(i) = \forall e \in L(i)$

sinon  $\pi'(i) = \pi(i)$ .

#### Commentaire :

Les règles d'amélioration de la politique 3.(a) et 3.(b) veut dire simplement qu'on sélectionne l'arc qui maximisent les équations : (17) et (18). Ce maximum est réalisé hiérarchiquement; l'équation (17) a la priorité sur l'équation (18) dans chaque pas d'amélioration de politique. Seulement quand l'équation (17) est satisfaite que (18) est utilisée pour déterminer la nouvelle politique. L'autre condition signifient simplement qu'on garde le plus possible les valeurs de la politique courante.

### 3.5.3 Algorithme général :

Entrée : Une bonne matrice polynômiale  $A$ .

Sortie : Un mode propre généralisé de  $A$ .

1. *Initialisation* : choisir une politique arbitraire  $\pi_1$ . calculer le mode propre généralisé  $(\eta^1, x^1)$  de  $A^{\pi_1}$  utilisant l'algorithme 4.1. Initialiser  $k$  à 1.

2. *Amélioration de la politique* : Améliorer la politique  $\pi_k$  en utilisant l'algorithme 4.2 avec des entrees  $\pi = \pi_k$ ,  $\eta = \eta^k$ ,  $x = x^k$ .

Si la condition d'arrêt de cet algorithme est satisfaite,  $(\eta^k, x^k)$  est alors un mode propre généralisé de  $A$ . l'algorithme s'arrête.

Sinon, poser  $\pi_{k+1} = \pi'$  ( la sortie de l'algorithme 4.2).

**3. Détermination de valeurs :** Calculer un mode propre généralisé  $(\eta^{k+1}, x^{k+1})$  utilisant l'algorithme 4.1 en prenant la valeur spéciale  $x_i^{k+1} = x_i^k$  dans l'étape 3 de l'algorithme.

**4.** Incrémenter k par 1 et aller à l'étape deux.

### Commentaire :

L'algorithme construit une sequence de modes propres généralisés  $(\eta^k, x^k)$ , qui est strictement ordonnée croissante dans l'ordre lexicographique sur  $(\mathbb{R}^n)^2$ , défini par  $(x, y) <_{lex} (x', y')$  si  $x < x'$  ou  $x = x'$  et  $y < y'$ .

Une bonne matrice polynômiale  $A$  a un mode propre généralisé  $(\eta, x)$ . En particulier le temps de cycle  $\chi(A) = \eta$  existe.

L'algorithme de Howard n'est pas limité au problème spectral. Il est possible de concevoir des algorithmes d'itération de politique pour les équations de la forme  $x = Ax \oplus b$ , où  $A$  est une matrice carrée ayant une valeur propre strictement inférieure à 1, et  $b$  un vecteur colonne.

### 3.5.4 Exemple d'application :

On va appliquer cet algorithme sur l'exemple de la figure 4.1 pour calculer le mode propre généralisé de la matrice qui y est présentée. Cet exemple correspond au cas où  $A = A \gamma$ , et  $\tau \equiv 1$ . En particulier, le multigraphe de  $A$  sera identifié à partir du graphe de  $A$ .

On choisira comme politique initiale  $\pi_1: 1 \rightarrow 1, i \rightarrow 2$ , pour  $i = 2, 3, 4$ . En appliquant l'algorithme 4.1, on trouve le circuit  $c_1: 1 \rightarrow 1$ , avec  $\bar{\eta} = \omega(c_1)/\tau(c_1) = 1$ . on a donc  $\eta_1^1 = 1, x_1^1 = 0$ . Puisque 1 est le seul noeud qui a accès à 1, on appliquera l'algorithme 4.1 au sous-graphe  $G_A$  avec les noeuds 2, 3, 4. On trouve le circuit  $c_2: 2 \rightarrow 2$  et on a  $\bar{\eta} = \omega(c_2)/\tau(c_2) = 3, \eta_2^1 = 3$  et  $x_2^1 = 0$ . Puisque 3 et 4 ont accès à 2, on a  $\eta_i^1 = 3$  pour  $i = 3, 4$ . on applique donc (21) et on a :  $x_3^1 = 4 - 3 + x_2^1, x_4^1 = 2 - 3 + x_2^1$ . Pour récapituler,

$$\eta^1 = [1 \ 3 \ 3 \ 3]^T, x^1 = [0 \ 0 \ 1 \ -1]^T$$

Pour améliorer la politique on utilise l'algorithme 4.3. puisque  $J = 1 \neq \emptyset$ , on a l'amélioration du type 3.a. Ceci donne  $\pi_2: i \rightarrow 2$ , pour  $i = 1, 2, 3, 4$ . seule l'entrée  $x_1^1$  et  $\eta_1^1$  va être changée.

$$\eta^2 = [3 \ 3 \ 3 \ 3]^T, x^2 = [-1 \ 0 \ 1 \ -1]^T$$

Dans l'étape suivante on appliquera 4.2 type 3b pour améliorer la politique.  $\pi_3: 1 \rightarrow 4, 2 \rightarrow 3, 3 \rightarrow 2, 4 \rightarrow 3$ . Algorithme 4.1 circuit trouvé :  $c: 3 \rightarrow 2 \rightarrow 3, \bar{\eta} = \omega(c_3)/\tau(c_3) = \bar{\eta} = (\omega(2,3) + \omega(3,2))/2 = 9/2$ .

$$\eta^3 = [9/2 \ 9/2 \ 9/2 \ 9/2]^T, x^3 = [\frac{11}{2} \ 0 \ \frac{-1}{2} \ 3]^T$$



FIG. 3.2 – Exemple d'application

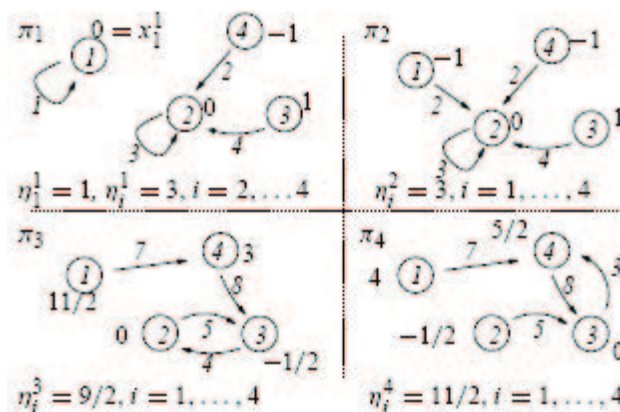


FIG. 3.3 – Suite des politique construites par itération sur les politiques

Dans la suite Algorithme 4.2 type 3b, le seul changement  $\pi_4(3) = 4$ . Algorithme 4.1 : circuit trouvé :  $c : 3 \rightarrow 4 \rightarrow 3$ ,  $\bar{\eta} = \omega(c_4)/\tau(c_4) = \bar{\eta} = (\omega(3,4) + \omega(4,3))/2 = 11/2$ .

$$\eta^4 = [11/2 \ 11/2 \ 11/2 \ 11/2]^T, \quad x^4 = [4 \ \frac{-1}{2} \ 0 \ \frac{5}{2}]^T$$

L'algorithme 4.3 s'arrête :  $11/2$  est une valeur propre de  $A$ , et  $x^4$  en est un vecteur propre.

### 3.6 Conclusion :

Si  $H$  est une fonction max-plus qui n'a pas un point fixe (de sorte que  $\chi(h) \neq h \ \forall h \in \mathbb{R}$ ), il est néanmoins possible de montrer qu'il existe  $\vec{x} \in \mathbb{R}^n$  tels que  $H^k(\vec{x}) = \vec{x} + k \chi(H) \ \forall 1 \leq k$ . Nous pouvons considérer ceci comme point fixe généralisé approprié pour la situation dans la quelle  $\chi(h) \neq h$ .

Les méthodes d'amélioration de politique de l'algorithme, peuvent être adaptées pour montrer que n'importe quelle fonction (min, max, +) a un point fixe généralisé [13]. Ceci rapporte immédiatement une preuve de la conjecture de dualité. Une preuve alternative, qui fonctionne pour une plus grande classe des fonctions topicales [12]. Ceci rappelle un problème non résolu qui consiste à prouver que l'algorithme donné ci dessus, pour une sous-classe des fonctions (min, max, +) peut être étendu à une classe plus large.

# CONCLUSION ET PERSPECTIVES

Un système à événements discrets est un système comprenant un ensemble fini d'événements qui se produisent de façon déterministe et répétée. Si  $n$  est le nombre d'événements, le système peut être modélisé par une certaine fonction  $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$  donnant la date de la prochaine occurrence de chaque événement  $i$ , par rapport à une date d'origine arbitraire. Dans ce cas, le comportement dynamique du système peut être modélisé par le système dynamique  $F$ .

Après avoir présenté la notion de treillis et l'algèbre des dioïdes, ce travail a étudié la théorie spectrale, la technique de calcul présentée ci-dessus, a mené à des algorithmes de calcul de point fixe ou de valeurs propres et de vecteurs propres.

Ce rapport s'est articulé sur deux directions. D'une part on a exposé une technique et un algorithme de calcul de points fixes d'une classe particulière de fonctions (Min, Max, +). Le développement de cet outil de calcul sous scilab est illustré par un exemple d'application.

D'une autre part, nous nous sommes attaché au cas général des fonctions (Min, Max, +), et nous avons exposé un outil plus efficace de calcul de point fixe généralisé.

Une généralisation vers une classe plus large a été présentée dans certains ouvrages tels que la thèse de Cochet-Terrasson 2001, donnant l'algorithme universel d'itération sur les politiques, ainsi que dans l'article de Costan, Gaubert, Goubault, Martel et Putot 2005, "A policy iteration algorithm for computing fixed points in static analysis of programs", qui présente un nouvel outil de calcul des point fixe encore plus efficace.

Les algorithmes d'itération de politique peuvent donner des résolutions précises des problèmes spectraux. Il existe des heuristiques dans le choix des politiques initiales qui pourraient être examinées, en utilisant par exemple un choix initial dynamique, en exigeant des dépendances des valeurs obtenues par le choix de la politique initiale. Un autre objectifs serait de généraliser l'algorithme d'itération de politique à des treillis de propriétés plus complexes, telles que celui d'octagons [44].

Ceci étant, nous pouvons affirmer qu'un algorithme d'itération de politique est meilleurs que celui des itérations de valeurs. En choisissant une iteration d'amélioration de politique, nous simplifions ainsi la résolution de l'ensemble des équations du système.

# BIBLIOGRAPHIE

Nous présentons ci dessous les bibliographies qui ont permis la réalisation de ce mémoire.

[Cochet-Terrasson, Gaubert, Gunawardena 1999] A constructive fixed point theorem for min-max functions.

Jean Cochet-Terrasson Guy Cohen . Stéphane Gaubert .;§ Michael Mc Gettrick . Jean-Pierre Quadrat .Numerical co

S. Gaubert and J. Gunawardena. A non-linear hierarchy for discrete event dynamical systems. Proceedings of WODES'98, IEE, Cagliari, Italy, August 1998.

S. Gaubert and J. Gunawardena. The duality theorem for  $(\min, \max, +)$  functions. Comptes Rendus Acad. Sci., 326:4348, 1998.

A. Costan, S. Gaubert, E. Goubault, M. Martel, S. Putot 2005. A policy iteration algorithm for computing fixed points in static analysis of programs

These de doctorat - Mehdi Lhommeau 2003. mputation of spectral elements in max plus algebra.

M. Gondran and M. Minoux. Graphs and Algorithms. Wiley-Interscience Series in Discrete Mathematics. John Wiley, 1984.

J. Gunawardena. Cycle times and fixed points of  $(\min, \max, +)$  functions. In G. Cohen and J.-P. Quadrat, editors, 11th International Conference on Analysis and Optimization of Systems, pages 266272. Springer LNCIS 199, 1994.

These de doctorat - J. Cochet Terrasson 2001.