



Contribution à l'analyse de performances des Systèmes à Événements Discrets non linéaires dans l'algèbre ($\min, +$)

Abderrahim Benfekir

► To cite this version:

Abderrahim Benfekir. Contribution à l'analyse de performances des Systèmes à Événements Discrets non linéaires dans l'algèbre ($\min, +$). Autre. Université de Cergy Pontoise, 2013. Français. <NNT : 2013CERG0657>. <tel-01128058>

HAL Id: tel-01128058

<https://tel.archives-ouvertes.fr/tel-01128058>

Submitted on 9 Mar 2015

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

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

ÉCOLE DOCTORALE DE L'UNIVERSITÉ DE CERGY

2013

Thèse de DOCTORAT

Spécialité : Génie Electrique

Présentée et soutenue publiquement par

Abderrahim BENFEKIR

le 19/12/2013

Université de Cergy

**Contribution à l'analyse de performances
des Systèmes à Événements Discrets non
linéaires dans l'algèbre $(\min, +)$**

Jury

Président : Abdelhafid OMARI, Professeur

Univ USTO, Algérie

Rapporteurs : Lionel AMODEO, Professeur

UTT, Troyes, France

Zaki SARI, Professeur

Univ de Tlemcen, Algérie

Examineurs :

Moumen DARCHERIF, Enseignant-Chercheur/Co-encadrant

EPMI, Cergy, France

Samir HAMACI, Enseignant-Chercheur/Co-encadrant

EPMI, Cergy, France

Directeurs de thèse : M. BOUHAMIDA & L. LAVAL

**CONTRIBUTION À L'ANALYSE DE
PERFORMANCES DES SYSTÈMES À
ÉVÉNEMENTS DISCRETS NON LINÉAIRES
DANS L'ALGÈBRE $(\min, +)$**

Abderrahim BENFEKIR



EPMI-USTO

Abderrahim BENFEKIR

*Contribution à l'analyse de performances des Systèmes à Événements Dis-
crets non linéaires dans l'algèbre $(\min, +)$*

xviii+

Résumé

Cette thèse s'inscrit dans le cadre de la théorie des systèmes linéaires dans les dioïdes. Cette théorie concerne la sous-classe des systèmes à événements discrets modélisables par les Graphes d'Événements Temporisés (GET). La dynamique de ces graphes peut être représentée par des équations récurrentes linéaires sur des structures algébriques particulières telles que l'algèbre $(\max, +)$ ou l'algèbre $(\min, +)$.

Ce mémoire est consacré à l'analyse de performances des systèmes dynamiques qui peuvent être modélisés graphiquement par des Graphes d'Événements Temporisés Généralisés (GETG). Ces derniers, contrairement au GET, n'admettent pas une représentation linéaire dans l'algèbre $(\min, +)$. Pour pallier à ce problème de non linéarité, nous avons utilisé une approche de modélisation définie sur un dioïde d'opérateurs muni de deux lois internes : loi additive (\oplus) correspondant à l'opération (\min) , et loi multiplicative (\otimes) équivalente à la loi de composition usuelle. Le modèle d'état obtenu, est utilisé pour évaluer les performances des GETG. Pour cela, nous avons proposé une nouvelle méthode qui a pour but de linéariser le modèle mathématique régissant l'évolution dynamique du modèle graphique, dans le but d'obtenir un modèle $(\min, +)$ linéaire. La deuxième partie de cette thèse est consacrée au problème qui consiste à déterminer les ressources à utiliser dans une ligne de production, en vue d'atteindre des performances souhaitée. Ceci est équivalent à déterminer le marquage initial de la partie commande du GETG.

Mots-clés : Réseaux de Petri, graphes d'événements temporisés généralisés, graphes d'événements temporisés, dioïde, algèbre $(\min, +)$, linéarisation, temps de cycle, optimisation.

Abstract

This thesis is part of the theory of linear systems over dioids. This theory concerns the subclass of discrete event dynamic systems modeled by Timed Event Graphs (TEG). The dynamics of these graphs can be represented by linear recurrence equations over specific algebraic structures such as $(\max, +)$ algebra or $(\min, +)$ algebra.

This report is devoted to the performance analysis of dynamic systems which can be represented graphically by Generalized Timed Event Graphs (GTEG). These type of graphs, unlike TEG, do not admit a linear representation in $(\min, +)$ algebra. To mitigate the problem of nonlinearity, we used a modeling approach defined on a dioid operators endowed with \min as (\oplus) usual composition as (\otimes) . The obtained state model is used to evaluate the performance of GTEG. For this, we proposed a new method to linearize the mathematical model governing the dynamic evolution of the graphical model in order to obtain a linear model in $(\min, +)$ algebra. The second part of this work is devoted to the problem of determining the resources to use in a production line, in order to achieve desired performance. These is equivalent to determining the initial marking of the control part of the GTEG.

Keywords: Petri Nets, generalized timed event graphs, timed event graphs, dioid, $(\min, +)$ algebra, linearization, cycle time, optimization.

Remerciements

Les travaux présentés dans ce mémoire ont été réalisés dans le cadre d'une thèse en cotutelle entre Ecole d'Électricité, de Productique et Méthodes Industrielle (EPMI), Université des Sciences et de la Technologie d'Oran, Algérie (USTO) et le Laboratoire ECS-Lab de l'Université de Cergy-Pontoise. Je remercie Moumen DARCHIRIF, directeur de l'EPMI et co-encadrant de cette thèse, de m'y avoir accueilli. Je tiens également à lui exprimer ma profonde gratitude pour ses conseils et son aide qui ont permis d'effectuer cette thèse dans les meilleures conditions possibles.

J'adresse mes sincères remerciements à mes directeurs de thèse, Mohamed BOUHAMIDA, Professeur à l'USTO et Laurent LAVAL, Maître de Conférences (HDR) au Laboratoire ECS-Lab, qui m'ont initié à la recherche. Leurs conseils et encouragements, aussi bien que la liberté d'initiative dont il m'ont laissé jouir, m'ont été très précieux.

Je suis particulièrement reconnaissant à Lionel AMODEO, Professeur à l'Université Technologique de Troyes et Zaki SARI, Professeur à l'Université de Tlemcen (Algérie), pour l'honneur qu'ils m'ont fait en acceptant de rapporter ce travail.

Je voudrais également remercier tous les membres de Jury, pour l'intérêt qu'ils ont porté à ces travaux en acceptant de les examiner.

Je tiens à exprimer particulièrement ma gratitude envers Samir HAMACI, Enseignant-Chercheur à l'EPMI et co-encadrant de cette thèse, pour sa disponibilité, son intuition scientifique et la pédagogie dont il a fait preuve à mon égard depuis le début de cette thèse.

Mes remerciements s'adressent également à Karim LABADI, Enseignant-Chercheur à l'EPMI, de l'intérêt qu'il a porté à ces travaux, et l'ambiance qu'il a fait régner dans le Laboratoire de Productique de l'EPMI, ont rendu m'y séjours très agréables.

Je voudrais également saluer mes collègues doctorants, particulièrement Taha BENARBIA pour l'ambiance et l'émulation qu'ils ont fait régner, ont rendu les années de thèse très agréables.

Je remercie mes parents pour le soutien qu'ils m'ont apporté tout au long de mes études.

A toute ma famille

Table des matières

Notations	3
Introduction Générale	5
1 Outils Algébriques	9
1.1 Préliminaires algébriques	9
1.1.1 Algèbre des dioïdes	9
1.1.2 Résolution d'Equations dans les dioïdes	13
1.2 Eléments de la théorie des systèmes linéaires	14
1.2.1 Notions sur les systèmes	14
1.2.2 Systèmes élémentaires	16
1.2.3 Opérations sur les systèmes	16
1.3 Conclusion	18
2 Modélisation des GETG dans l'algèbre des dioïdes	19
2.1 Introduction	19
2.2 Réseaux de Petri	20
2.2.1 Définitions et notations	20
2.2.2 Invariants d'un réseau de Petri	25
2.3 Graphes d'Evénements Temporisés	28
2.3.1 Conditions initiales	29
2.3.2 Modélisation des GET dans l'algèbre $(\min, +)$	30
2.3.3 Forme récurrente explicite	34
2.4 Graphes d'Evénements Temporisés Généralisés	36
2.4.1 Présentation des GETG	36
2.4.2 Equations récurrentes des GETG	39
2.4.3 Modélisation à base d'opérateurs des GETG	40
2.5 Conclusion	45
3 Evaluation de Performances des GETG	47
3.1 Introduction et état de l'art	47
3.2 Notions élémentaires	53
3.3 Eléments de la théorie spectrale	57

3.4	Evaluation de performances des GET	61
3.5	Evaluation de performances des GETG	64
3.5.1	Méthode de linéarisation proposée	64
3.5.2	Exemple : Extension au cas des GETG avec entrée/sortie .	73
3.6	Conclusion	75
4	Allocation de ressources des GETG	77
4.1	Introduction et état de l’art	77
4.2	Recherche des équations récurrentes linéaires	79
4.2.1	Présentation de la méthode de linéarisation utilisée	79
4.3	Allocation de ressources	85
4.3.1	Approche développée	86
4.3.2	Exemple illustratif	87
4.3.3	Application à un Atelier Flexible	92
4.4	Conclusion	98
5	Conclusion et Perspectives	101
6	Publications Personnelles	103
7	Annexes	105
	Bibliographie	111

Liste des Notations

SED : Système à Evénements Discret.

RdP : Réseau de Petri.

\mathcal{P} : ensemble non vide de places d'un réseau de Petri.

\mathcal{T} : ensemble non vide de transitions d'un réseau de Petri.

m_p : nombre de jetons initialement contenus dans la place p .

τ_p : durée minimale de séjour d'un jeton dans la place p .

GET : Graphe d'Evénements Temporisé.

GETG : Graphe d'Evénements Temporisé Généralisé

θ : vecteur T-invariant associé au graphe d'événements temporisé généralisé.

M_{qp} : poids de l'arc allant de la place p vers la transition x_q .

$M_{pq'}$: poids de l'arc allant de la transition $x_{q'}$ vers la place p .

\mathcal{N}_p : marque valuée associée à la place p .

\mathcal{N}_ϱ : marque valuée associée au circuit ϱ .

\oplus : addition dans un dioïde.

\otimes : multiplication dans un dioïde.

\mathbb{Z}_{max} : algèbre $(\mathbb{Z} \cup \{-\infty\}, max, +)$.

$\overline{\mathbb{Z}}_{max}$: algèbre $(\mathbb{Z} \cup \{\pm\infty\}, max, +)$.

\mathbb{Z}_{min} : algèbre $(\mathbb{Z} \cup \{+\infty\}, min, +)$.

$\overline{\mathbb{Z}}_{min}$: algèbre $(\mathbb{Z} \cup \{\pm\infty\}, min, +)$.

\preceq : relation d'ordre liée à un dioïde.

a^* : étoile de kleene.

$c_q(t)$: variable compteur associée à la transition c_q d'un graphe d'événements temporisé.

$x_q(t)$: variable compteur associée à la transition x_q d'un graphe d'événements temporisé généralisé.

$\mathcal{O}_{\min}[\delta]$: dioïde d'opérateurs permettant de modéliser les graphes d'événements temporisés généralisé.

$\gamma^\nu, \nu \in \mathbb{Z} \cup \{\pm\infty\}$: opérateur de stock sur les compteurs.

$\delta^\tau, \tau \in \mathbb{Z} \cup \{\pm\infty\}$: opérateur de retard sur les compteurs.

$\mu_m, m \in \mathbb{Q}$: opérateur multiplieur sur les compteurs.

$\overline{\mathbb{Z}}_{min}^{\mathbb{Z}}$: ensemble de signaux de \mathbb{Z} dans $\overline{\mathbb{Z}}_{min}$.

TC : Temps de Cycle d'un graphe d'événements temporisé.

λ : Taux de production d'un graphe d'événements temporisé.

TC_m : Temps de Cycle d'un graphe d'événements temporisé généralisé.

TC_a : Temps de Cycle atteignable.

λ_{m_q} : fréquence de franchissement associée à la transition x_q d'un graphe d'événements temporisé généralisé.

$\lceil a \rceil$: partie entière supérieure de a .

$\lfloor a \rfloor$: partie entière inférieure de a .

C : matrice d'incidence.

K : P-invariant.

Θ : T-invariant.

M_0 : vecteur marquage initial.

u.t : unité de temps.

Introduction Générale

Dans la catégorie de Systèmes (Dynamiques) à Événements Discrets (SED) sont regroupés certains systèmes dont le comportement dynamique ne peut être décrit par des équations différentielles. Cette classe de systèmes regroupe aussi bien les systèmes de production, pour lesquels on étudie, par exemple, les flux de matières, que les systèmes de transport ou les systèmes informatiques. Pour étudier ces systèmes, il est nécessaire de faire appel à des modèles aptes à prendre en compte toutes leurs caractéristiques dynamiques souvent de natures complexes. Or, les phénomènes mis en jeu par les SED, et responsables de leur dynamique, sont nombreux et de natures diverses : tâches séquentielles ou simultanées, temporisées ou non, synchronisées ou concurrentes. De cette diversité de phénomènes provient l'incapacité de décrire l'ensemble des SED par un modèle unique qui soit à la fois fidèle à la réalité et exploitable mathématiquement.

Dans la littérature, de nombreuses classes de modèles sont proposées pour l'étude des systèmes dynamiques. Le choix d'un type de modèle est en fonction des objectifs recherchés et de la nature du système considéré. Dans le cas des SED, plusieurs concepts de modélisation ont été élaborés : par exemple, la théorie des Réseaux de File d'Attente, pour l'évaluation de performances des systèmes temporisés [Baynat 2000], ou les réseaux de Petri déterministes pour l'optimisation de ressources [Proth 1995]. Certaines sous-classes de SED bénéficient néanmoins de modèles bien adaptés pour aborder, par exemple, les problèmes d'évaluation de performances ou de commande. Il a été montré que les systèmes mettant uniquement en jeu des phénomènes de synchronisation et de saturation peuvent être modélisés par des réseaux de Petri particuliers, appelés Graphes d'Événements Temporisés (GET). Ces derniers admettent une représentation linéaire sur une structure algébrique particulière, connue sous le nom de l'algèbre des dioïdes [Baccelli 1992] (l'algèbre $(\min, +)$ étant un exemple de dioïde). Cependant, les techniques développées dans le cadre des SED atteignent leur limite lorsque la taille du système considéré est importante (du fait du nombre important d'entités). Il s'avère alors utile d'utiliser des GET à arcs pondérés, encore appelés GET Généralisés (GETG),

ce qui permet de réduire la taille du modèle. Ces graphes permettent également de modéliser de façon simple des opérations d'assemblage et de désassemblage de produits présentes dans certains systèmes de production.

Contrairement aux GET, les GETG n'admettent pas une représentation linéaire dans l'algèbre des dioïdes. Par conséquent, l'utilisation de la théorie développée pour l'étude et l'analyse des GET n'est pas applicable pour les modèles pondérés qui sont les GETG.

Dans cette thèse, nous nous sommes intéressés à deux problématiques associées aux GETG :

1. Evaluation de performances :

Pour évaluer les performances d'un système modélisé par un GETG, on s'intéresse au calcul de deux indicateurs de performances : temps de cycle et le taux de production. Si la théorie spectrale permet de calculer ces performances dans le cas des GET [Gaubert 1992], le problème est toujours ouvert pour les GETG, ceci à cause de la non linéarité du modèle mathématique modélisant la dynamique du système. Cette non linéarité, qui est due aux parties entières engendrées par la présence des poids sur les arcs, empêche l'utilisation de la théorie développée dans le cadre de l'analyse de performances des systèmes linéaires dans les dioïdes.

Pour palier à ce problème de non linéarité, nous proposons une nouvelle méthode de linéarisation des équations d'états régissant l'évolution dynamique du graphe. Le modèle mathématique obtenu est $(\min, +)$ linéaire, il permet d'utiliser les résultats développés dans le cadre de l'analyse de performances des GET ordinaires dans l'algèbre des dioïdes.

2. Allocation de ressources :

Dans cette partie nous nous sommes intéressés à l'allocation de ressources dans les GETG. Le problème consiste à déterminer une répartition minimale des ressources en vue d'atteindre des performances souhaitées. Ceci revient à déterminer un nombre minimum de jetons à utiliser permettant de saturer le "processus goulot" tout en minimisant une somme pondérée des marquages des places. D'un point de vue pratique, pour un système de production, ce problème consiste à déterminer, durant de *phase de conception*, les ressources de production (chariots filoguidés, palettes, kanbans, stocks,...)

aussi réduit que possible à utiliser en vue d'atteindre un temps de cycle ou un taux de production souhaité. Il est donc possible de dimensionner les zones de stockage ou de déterminer le nombre minimum de ressources de transport qui permet de saturer la machine goulot dit aussi pilote (c-à-d la machine menante), tout en minimisant le coût.

Pour atteindre cet objectif, nous proposons une démarche originale. La procédure consiste à déterminer des équations récurrentes linéaires équivalentes, en utilisant des travaux déjà existants. Ces équations linéaires permettent de déduire le GET équivalent sous certaines conditions sur le marquage initial. Le marquage recherché se déduit en utilisant les performances souhaitées, fixées préalablement lors de la phase de conception.

Les travaux présentés dans ce mémoire visent à étendre certains résultats établis sur l'évaluation de performances et allocation de ressources des Graphes d'Événements Temporisés (classiquement étudiés dans le cadre de l'algèbre des dioïdes) aux Graphes d'Événements Temporisés Généralisés.

Ce mémoire est structuré en quatre chapitres :

- Dans le premier chapitre, nous passons en revue l'ensemble des outils algébriques utiles aux développements des résultats présentés dans ce mémoire. Cette présentation, purement bibliographique, ne se veut, en aucun cas, exhaustive, seuls quelques éléments de la théorie des dioïdes, résolution d'équations au point fixe ainsi qu'un rappel sur les signaux et la théorie des systèmes seront présentés.
- Afin d'aborder la modélisation des GETG dans les dioïdes, nous présentons, dans le deuxième chapitre, le dioïde d'opérateurs $\mathcal{O}_{\min}[\delta]$ qui sera utilisé, par la suite, comme la structure algébrique pour l'étude et l'analyse des GETG. Des exemples de modélisation de la dynamique des GETG dans ce dioïde seront présentés.
- Le troisième chapitre aborde le problème de l'évaluation de performances des GETG. Pour évaluer les performances des systèmes modélisés par ce type de graphes, une nouvelle méthode de linéarisation est proposée. Son

principe consiste à linéariser le modèle mathématique qui représente l'évolution dynamique d'un graphe dans le dioïde d'opérateurs $\mathcal{O}_{\min}[\delta]$, en synchronisant l'évolution de l'état, qui est non linéaire, avec une entrée impulsionnelle. Cette dernière, de par son amplitude qui tend vers l'infini, présente un intérêt mathématique, elle permet de linéariser toute équation récurrente non linéaire modélisant la dynamique du GETG. Le modèle mathématique obtenu est $(\min, +)$ linéaire, il permet d'utiliser certains résultats de base de la théorie spectrale, utilisés pour évaluer les performances des GET ordinaires.

- Le quatrième chapitre aborde le problème d'allocation de ressources des GETG. Le problème consiste à déterminer une répartition minimale des ressources en vue d'atteindre des performances souhaitées. Pour atteindre cet objectif, nous avons procédé par la recherche des équations récurrentes linéaires équivalentes, en s'inspirant des travaux déjà existants. Ces équations linéaires permettent de déduire le GET équivalent. Le marquage de ce dernier, se déduit en utilisant la théorie spectrale et les performances souhaitées, fixées préalablement. Par la suite, en exploitant la condition de linéarisation, nous déterminons le marquage saturant le "processus goulot" et qui minimise une somme pondérée des marquages des places du GETG.
- En conclusion, nous faisons une synthèse sur le travail réalisé et proposons des perspectives de recherche.

Outils Algébriques

Sommaire

1.1	Préliminaires algébriques	9
1.1.1	Algèbre des dioïdes	9
1.1.2	Résolution d'Equations dans les dioïdes	13
1.2	Éléments de la théorie des systèmes linéaires	14
1.2.1	Notions sur les systèmes	14
1.2.2	Systèmes élémentaires	15
1.2.3	Opérations sur les systèmes	16
1.3	Conclusion	17

1.1 Préliminaires algébriques

Dans ce premier chapitre, nous faisons un bref passage sur les définitions, notations et les résultats relatifs à l'algèbre des dioïdes. Bien que la théorie des dioïdes trouve un écho croissant auprès de la communauté qui s'intéresse aux Systèmes à Événements Discrets (SED), il nous a semblé nécessaire d'explicitier les éléments que nous serons amenés à utiliser dans ce mémoire.

Nous donnons, par la suite, un ensemble de définitions et notations relatives aux signaux et systèmes linéaires après avoir présenté la résolution d'équations au point fixe dans les dioïdes.

1.1.1 Algèbre des dioïdes

Cette section rappelle certaines propriétés propres à la structure algébrique d'un dioïde. Pour des exposées plus complets, on renvoie le lecteur aux ouvrages [Baccelli 1992], [Gunawardena 1998] et à la thèse de S. Gaubert [Gaubert 1992].

Définition 1.1 (Dioïde). On appelle un dioïde (ou semi-anneau idempotent) un ensemble \mathcal{D} muni de deux lois internes \oplus et \otimes telles que $\forall a ; b ; c \in \mathcal{D}$:

- la loi additive \oplus est :
 - associative : $(a \oplus b) \oplus c = a \oplus (b \oplus c) = a \oplus b \oplus c$,
 - commutative : $a \oplus b = b \oplus a$,
 - idempotente : $a \oplus a = a$,
 - d'élément neutre noté : $a \oplus \varepsilon = \varepsilon \oplus a = a$.
- la loi multiplicative \otimes est :
 - associative : $(a \otimes b) \otimes c = a \otimes (b \otimes c) = a \otimes b \otimes c$,
 - distributive à gauche et à droite par rapport à l'addition : $a \otimes (b \oplus c) = (a \otimes b) \oplus (a \otimes c)$ et $(b \oplus c) \otimes a = (b \otimes a) \oplus (c \otimes a)$,
 - d'élément absorbant ε : $a \otimes \varepsilon = \varepsilon \otimes a = \varepsilon$,
 - d'élément neutre noté e : $a \otimes e = e \otimes a = a$.

De plus, un semi-anneau idempotent, couramment noté $(\mathcal{D} ; \oplus ; \otimes)$, est commutatif lorsque la loi multiplicative \otimes est commutatif, soit lorsque $a \otimes b = b \otimes a$.

Exemple 1.1 (Dioïdes \mathbb{Z}_{min}). $\mathbb{Z}_{min} = (\mathbb{Z} \cup \{+\infty\}, \min, +)$ est un dioïde commutatif, où la loi additive \oplus correspond à l'opération \min usuelle ; la loi multiplicative \otimes est équivalente à l'addition usuelle. L'élément zéro de \mathbb{Z}_{min} est $\varepsilon = +\infty$ et l'élément identité est $e = 0$. Ce dioïde est appelé *l'algèbre* $(\min, +)$.

- **Application** On s'intéresse à la production de bicyclettes (figure 1.1), on peut dire qu'une bicyclette est la "somme" d'une paire de roues et d'un cadre. Cette nouvelle "somme", notée \oplus pour la distinguer du $+$ habituel, correspond à l'opération d'assemblage. On notera que lon se permet alors d'"ajouter" des grandeurs ne s'exprimant pas dans les mêmes unités. Si on a 2 roues et 1 cadre, on ne peut fabriquer qu'une seule bicyclette, donc $1 \oplus 2 = 1$, ce qui est équivalent dans l'algèbre usuelle au $\min(2, 1) = 1$.

Exemple 1.2 (Dioïdes \mathbb{Z}_{max}). $\mathbb{Z}_{max} = (\mathbb{Z} \cup \{-\infty\}, \max, +)$ est un dioïde commutatif où la loi additive \oplus correspond à l'opération \max usuelle ; la loi multiplicative \otimes est équivalente à l'addition usuelle et pour lequel $\varepsilon = -\infty$ et $e = 0$. Ce dioïde est appelé *l'algèbre* $(\max, +)$.

- Opération dans l'algèbre $(\max, +)$

$$10 \oplus 8 = 10 \Leftrightarrow \max(10, 8) = 10$$

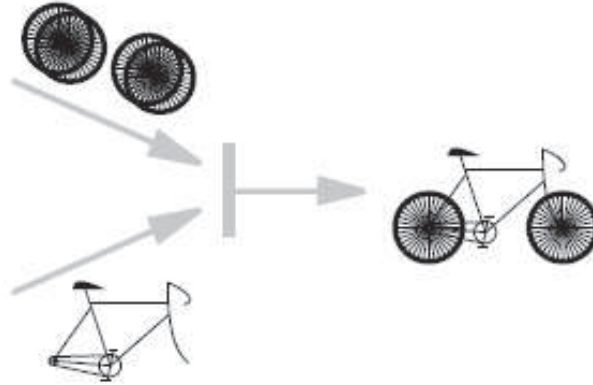


FIGURE 1.1 – Assemblage de bicyclettes.

$$2 \otimes 3 = 5 \Leftrightarrow 2 + 3 = 5$$

Définition 1.2 (Dioïde complet). Un dioïde $(\mathcal{D}, \oplus, \otimes)$ est *complet* si :

- il est fermé pour les sommes infinies,
- la loi \otimes distribue sur les sommes infinies, c'est-à-dire si pour tout $a \in \mathcal{D}$ et tout sous-ensemble $\mathcal{B} \subseteq \mathcal{D}$,

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

Un dioïde complet admet un élément maximum $\bigoplus_{a \in \mathcal{D}} a$, que l'on notera T pour *Top*. Il est absorbant pour l'addition, autrement dit, $\forall a \in \mathcal{D}, T \oplus a = T$.

Notons que, d'après la définition d'un dioïde, l'élément zéro ε est absorbant pour la multiplication pour tout élément de \mathcal{D} , aussi, on a : $T \otimes \varepsilon = \varepsilon \otimes T = \varepsilon$.

Exemple 1.3 On vérifie facilement que les dioïdes suivants sont complets :

- $(\mathbb{Z} \cup \{-\infty, +\infty\}, \min, +)$ avec $\varepsilon = +\infty, e = 0$ et $T = -\infty$ est un dioïde noté par la suite : $\overline{\mathbb{Z}}_{\min}$. On a dans $\overline{\mathbb{Z}}_{\min}$ la règle suivante :

$$(-\infty) + (+\infty) = (+\infty).$$

- $(\mathbb{Z} \cup \{-\infty, +\infty\}, \max, +)$ avec $\varepsilon = -\infty, e = 0$ et $T = +\infty$ est un dioïde noté par la suite : $\overline{\mathbb{Z}}_{\max}$. On a dans $\overline{\mathbb{Z}}_{\max}$ la règle suivante :

$$(-\infty) + (+\infty) = (-\infty).$$

Définition 1.3 (Dioïde matriciel). L'ensemble des matrices carrées de dimension n , à coefficients dans un dioïde $(\mathcal{D}, \oplus, \otimes)$, est un dioïde matriciel, noté $(\mathcal{D}^{n \times n}, \oplus, \otimes)$, où les opérations sont définies, à partir des opérations du dioïde $(\mathcal{D}, \oplus, \otimes)$ de manière analogue à l'algèbre classique, de la façon suivante :

$$\forall A, B \in \mathcal{D}^{n \times n};$$

$$A \oplus B : (A \oplus B)_{ij} = A_{ij} \oplus B_{ij}, \quad \forall i, j = 1, \dots, n,$$

$$A \otimes B : (A \otimes B)_{ij} = \bigoplus_{k=1}^n A_{ik} \otimes B_{kj}, \quad \forall i, j = 1, \dots, n.$$

L'élément identité de $\mathcal{D}^{n \times n}$ est la matrice, notée Id_n , composée de e sur la diagonale et de ε partout ailleurs. L'élément zéro est la matrice composée exclusivement de ε .

La somme et le produit de deux matrices de dimensions compatibles, pas nécessairement carrées, peuvent être définis de la façon suivante :

$$\bullet A \in \mathcal{D}^{n \times p}, \quad B \in \mathcal{D}^{n \times p};$$

$$A \oplus B : (A \oplus B)_{ij} = A_{ij} \oplus B_{ij}, \quad \forall i = 1, \dots, n, \quad \forall j = 1, \dots, p,$$

$$\bullet C \in \mathcal{D}^{n \times p}, \quad D \in \mathcal{D}^{p \times q};$$

$$C \otimes D : (C \otimes D)_{ij} = \bigoplus_{k=1}^p C_{ik} \otimes D_{kj}, \quad \forall i = 1, \dots, n, \quad \forall j = 1, \dots, q.$$

Pour que ces matrices puissent être manipulées comme des éléments d'un dioïde matriciel, il faut, en toute rigueur, considérer qu'elles appartiennent au dioïde de matrices carrées de dimension $\max(n, p, q) \times \max(n, p, q)$, en les complétant, si nécessaire, pour cela de lignes et/ou de colonnes constituées de l'élément ε .

Exemple 1.4 (Calcul Matriciel dans l'algèbre $(\max, +)$).

$$\begin{pmatrix} 3 & 10 \\ 0 & 5 \end{pmatrix} \oplus \begin{pmatrix} 8 & 1 \\ 4 & 5 \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

$$\bullet a = (3 \otimes 8) \oplus (10 \otimes 4) = \max(3 + 8, 10 + 4) = 14$$

- $b = (3 \otimes 1) \oplus (10 \otimes 5) = \max(3 + 1, 10 + 5) = 15$
- $c = (0 \otimes 8) \oplus (5 \otimes 4) = \max(0 + 8, 5 + 4) = 9$
- $d = (0 \otimes 1) \oplus (5 \otimes 5) = \max(0 + 1, 5 + 5) = 10$

Définition 1.4 (Dioïde de séries formelles). Soit $(\mathcal{D}, \oplus, \otimes)$ un dioïde, on définit une série formelle en q variables, notées p_1 à p_q , à coefficients dans \mathcal{D} comme une application Π de \mathbb{Z}^q dans \mathcal{D} : $\forall \kappa = (\kappa_1, \dots, \kappa_q) \in \mathbb{Z}^q$, $\Pi(\kappa)$ représente le coefficient de $p_1^{\kappa_1} \dots p_q^{\kappa_q}$. Une autre représentation équivalente de la série Π est :

$$\Pi = \bigoplus_{\kappa \in \mathbb{Z}^q} \Pi(\kappa_1, \dots, \kappa_q) p_1^{\kappa_1} \dots p_q^{\kappa_q}.$$

L'ensemble des séries formelles muni des opérations :

$$\Pi \oplus \Psi : (\Pi \oplus \Psi)(\kappa) = \Pi(\kappa) \oplus \Psi(\kappa),$$

$$\Pi \otimes \Psi : (\Pi \otimes \Psi)(\kappa) = \bigoplus_{i+j=\kappa} \Pi(i) \otimes \Psi(j),$$

est un dioïde noté $\mathcal{D}[[p_1, \dots, p_q]]$.

On appelle *support* de la série formelle Π l'ensemble :

$$\text{Supp}(\Pi) = \{\kappa \in \mathbb{Z}^q \mid \Pi(\kappa) \neq \varepsilon\}.$$

Une série formelle à support fini est appelée *polynôme*. Une série formelle dont le support est un singleton (de \mathbb{Z}^q) est appelée *monôme*.

- Des exemples sur les dioïdes de séries formelles sont exposés en Annexe.

1.1.2 Résolution d'Equations dans les dioïdes

Certaines équations définies dans des dioïdes complets admettent des solutions particulières extrêmes, c.-à-d. plus petite ou plus grande que toute autre solution. Des résultats généraux concernant l'étude d'équations au point fixe sur des dioïdes complets sont fournis dans [Baccelli 1992]. Nous rappelons ici simplement le résultat concernant la résolution de l'équation implicite $x = a \otimes x \oplus b$ qu'on va utiliser par la suite.

Théorème 1.1 Soit \mathcal{D} un dioïde complet, l'équation implicite

$$x = a \otimes x \oplus b$$

admet $x = a^*b = \bigoplus_{k \geq 0} a^k b$ comme plus petite solution. L'opérateur "*" est appelé *étoile de Kleene* dans la littérature.

Preuve 1 On vérifie que $x = a^*b$ est solution de $x = a \otimes x \oplus b$. On a

$$a(a^*b) \oplus b = a(e \oplus a \oplus a^2 \oplus \dots)b \oplus b = (e \oplus a \oplus a^2 \oplus a^3 \dots)b = a^*b.$$

1.2 Éléments de la théorie des systèmes linéaires

1.2.1 Notions sur les systèmes

Après cette présentation sur les dioïdes, la suite de ce chapitre est consacrée à la présentation des éléments de la théorie des systèmes linéaires dans les dioïdes.

Définition 1.5 (Signal). On appelle *signal* une application u de $\mathbb{Z} \rightarrow (\mathcal{D}, \oplus, \otimes)$, $t \mapsto u(t)$. On note $\mathcal{D}^{\mathbb{Z}}$ l'ensemble des signaux muni de deux opérations :

- Une loi additive interne, notée \oplus , définie par :

$$\forall u_1, u_2 \in \mathcal{D}^{\mathbb{Z}}, (u_1 \oplus u_2)(t) = u_1(t) \oplus u_2(t).$$

- Une loi externe, qui joue le rôle du produit d'un signal par un scalaire, notée " \cdot ", définie par :

$$\forall a \in \mathcal{D}, \forall u \in \mathcal{D}^{\mathbb{Z}}, (a \cdot u)(t) = a \otimes u(t).$$

L'ensemble $\mathcal{D}^{\mathbb{Z}}$ est composé de signaux. En effet :

- Un signal est croissant s'il satisfait

$$\forall t \in \mathbb{Z}, \forall \tau \in \mathbb{N}, u(t + \tau) \succeq u(t).$$

- Un signal est décroissant s'il satisfait

$$\forall t \in \mathbb{Z}, \forall \tau \in \mathbb{N}, u(t + \tau) \preceq u(t).$$

Exemple 1.5 L'ensemble $\overline{\mathbb{Z}}_{min}^{\mathbb{Z}}$ constitue un ensemble de signaux.

Définition 1.6 (Système linéaire). Un système, noté \mathbb{S} , est un opérateur défini de $\overline{\mathbb{Z}}_{min}^{\mathbb{Z}}$ vers $\overline{\mathbb{Z}}_{min}^{\mathbb{Z}}$. Il est linéaire si :

$$\forall u_1, u_2 \in \mathcal{D}^{\mathbb{Z}}, \quad \mathbb{S}(u_1 \oplus u_2) = \mathbb{S}(u_1) \oplus \mathbb{S}(u_2) \quad (\text{propriété d'additivité}), \quad (1.1)$$

$$\forall u_1 \in \mathcal{D}^{\mathbb{Z}}, \forall a \in \mathcal{D}, \quad \mathbb{S}(a \otimes u_1) = a \otimes \mathbb{S}(u_1) \quad (\text{propriété d'homogénéité}). \quad (1.2)$$

Définition 1.7 (Produit de convolution). Le *produit de convolution* de deux signaux est noté \otimes et est défini par :

$$\forall u_1, u_2 \in \mathcal{D}^{\mathbb{Z}}, \quad (u_1 \otimes u_2)(t) = \bigoplus_{\tau \in \mathbb{Z}} u_1(\tau) u_2(t - \tau) = \bigoplus_{\tau \in \mathbb{Z}} u_1(t - \tau) u_2(\tau),$$

Exemple 1.6 Le produit de convolution de deux signaux de $\overline{\mathbb{Z}}_{min}^{\mathbb{Z}}$ satisfait la relation suivante :

$$\forall u_1, u_2 \in \overline{\mathbb{Z}}_{min}^{\mathbb{Z}}, \forall t \in \mathbb{Z}, \quad (u_1 \otimes u_2)(t) = \min_{\tau \in \mathbb{Z}} \{u_1(\tau) + u_2(t - \tau)\},$$

on parle de produit d'*inf-convolution*.

De manière duale, le produit de convolution de deux signaux de $\overline{\mathbb{Z}}_{max}^{\mathbb{Z}}$ donne

$$\forall u_1, u_2 \in \overline{\mathbb{Z}}_{max}^{\mathbb{Z}}, \forall t \in \mathbb{Z}, \quad (u_1 \otimes u_2)(t) = \max_{\tau \in \mathbb{Z}} \{u_1(\tau) + u_2(t - \tau)\}.$$

on parle de produit de *sup-convolution*.

Définition 1.8 (Système continu). Un système \mathbb{S} est *continu* si pour tout sous-ensemble fini ou infini $(u_i)_{i \in I}$ d'éléments de $\mathcal{D}^{\mathbb{Z}}$, on a :

$$\mathbb{S}\left(\bigoplus_{i \in I} u_i\right) = \bigoplus_{i \in I} \mathbb{S}(u_i),$$

sachant que $(\mathcal{D}, \oplus, \otimes)$ est un dioïde complet.

Définition 1.9 (Système causal). Un système \mathbb{S} est *causal* si pour toutes entrées u_1 et u_2 , on a :

$$\forall \tau \in \mathbb{Z}, \quad u_1(t) = u_2(t) \text{ pour } t \leq \tau \Rightarrow [\mathbb{S}(u_1)](t) = [\mathbb{S}(u_2)](t) \text{ pour } t \leq \tau.$$

1.2.2 Systèmes élémentaires

On présente dans cette partie deux systèmes élémentaires linéaires :

- *Système gain* Γ^ν défini de $\overline{\mathbb{Z}}_{min}^{\mathbb{Z}} \rightarrow \overline{\mathbb{Z}}_{min}^{\mathbb{Z}}$ par :

$$\forall t \in \mathbb{Z}, \forall u, y \in \overline{\mathbb{Z}}_{min}^{\mathbb{Z}}, \nu \in \overline{\mathbb{Z}}, y(t) = [\Gamma^\nu(u)](t) = \nu \otimes u(t).$$



FIGURE 1.2 – Système Γ^ν .

- *Système retard* Δ^τ défini de $\overline{\mathbb{Z}}_{min}^{\mathbb{Z}} \rightarrow \overline{\mathbb{Z}}_{min}^{\mathbb{Z}}$ par :

$$\forall t \in \mathbb{Z}, \forall u, y \in \overline{\mathbb{Z}}_{min}^{\mathbb{Z}}, \tau \in \overline{\mathbb{Z}}, y(t) = [\Delta^\tau(u)](t) = u(t - \tau).$$



FIGURE 1.3 – Système Δ^τ .

1.2.3 Opérations sur les systèmes

Il existe trois opérations élémentaires permettant de mettre en relation des systèmes : la composition série, la composition parallèle et la mise en feedback.

- **Composition série** $\mathbb{S} = \mathbb{S}_1 \otimes \mathbb{S}_2$ est définie par :

$$[\mathbb{S}(u)](t) = [\mathbb{S}_2(\mathbb{S}_1(u))](t).$$



FIGURE 1.4 – Composition série.

- **Composition parallèle** $\mathbb{S} = \mathbb{S}_1 \oplus \mathbb{S}_2$ est définie par :

$$[\mathbb{S}(u)](t) = [\mathbb{S}_1(u)](t) \oplus [\mathbb{S}_2(u)](t).$$

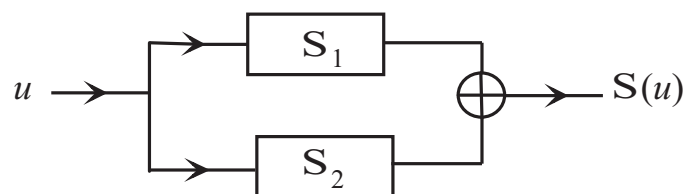


FIGURE 1.5 – Composition parallèle.

- **Mise en feedback** On considère le schéma suivant, où \mathbb{S}_1 et \mathbb{S}_2 sont deux opérateurs linéaires.

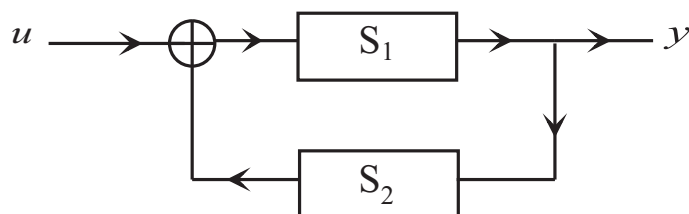


FIGURE 1.6 – Mise en feedback.

La relation entre la sortie y et l'entrée u satisfait :

$$y = \mathbb{S}_1(\mathbb{S}_2(y) \oplus u) = \mathbb{S}_1(\mathbb{S}_2(y)) \oplus \mathbb{S}_1(u).$$

D'après le théorème 1.1, la plus petite solution existe et est donnée par :

$$(\mathbb{S}_1\mathbb{S}_2)^*\mathbb{S}_1(u) = \left(\bigoplus_{n \in \mathbb{N}} (\mathbb{S}_1\mathbb{S}_2)^n\right)\mathbb{S}_1(u).$$

1.3 Conclusion

Ce chapitre introductif a permis de dresser la liste des outils mathématiques qui seront utilisés par la suite. Il s'agit d'un préalable à l'analyse de performances des GETG, présentée dans les chapitres suivants. Nous avons donné un ensemble de définitions sur la théorie des systèmes linéaires dans les dioïdes, pour aborder la représentation de certains systèmes dynamique non linéaires, modélisables par des GETG, dans un dioïde d'opérateurs.

Modélisation des GETG dans l'algèbre des dioïdes

Sommaire

2.1	Introduction	19
2.2	Réseaux de Petri	19
2.2.1	Définitions et notations	20
2.2.2	Invariants d'un réseau de Petri	25
2.3	Graphes d'Événements Temporisés	27
2.3.1	Conditions initiales	29
2.3.2	Modélisation des GET dans l'algèbre $(\min, +)$	30
2.3.3	Forme récurrente explicite	34
2.4	Graphes d'Événements Temporisés Généralisés	36
2.4.1	Présentation des GETG	36
2.4.2	Equations récurrentes des GETG	38
2.4.3	Modélisation à base d'opérateurs des GETG	40
2.5	Conclusion	45

2.1 Introduction

Ce chapitre a trait à la modélisation des Graphes d'Événements Temporisés Généralisés (GETG) dans l'algèbre de dioïdes. On présente sommairement les réseaux de Petri, qui sont un langage graphique permettant de décrire les Systèmes à Evènements Discrets (SED). L'emploi des réseaux de Petri pour la spécification, la simulation et l'analyse de ces systèmes remonte aux années 60 et s'est maintenant généralisé. La littérature sur les réseaux de Petri généraux est très abondante. On s'est limité aux préliminaires indispensables dans la première section

de ce chapitre. Les Graphes d'Événements Temporisés Généralisés, qui sont une sous classe des RdP, sont introduits par la suite, après avoir présenté les Graphes d'Événements Temporisés (GET) dans la section 2. Il est montré, en section 3, que la présence des poids sur les arcs fait qu'un GETG n'admet pas une représentation $(\min, +)$ linéaire. Comme alternative, en vue d'analyser ces graphes, nous utilisons une approche à base d'opérateurs, afin d'établir le modèle mathématique régissant l'évolution dynamique de ces modèles dans un dioïde d'opérateurs [Cohen 1998b]. Le modèle d'état obtenu sera utilisé, dans le chapitre suivant, pour évaluer certains indicateurs de performances des systèmes modélisables par ce types de graphes.

2.2 Réseaux de Petri

Les Réseaux de Petri (RdP) constituent un formalisme graphique propre à la modélisation des Systèmes à Événements Discrets (SED) introduit en 1962 par Carl Adam Petri [Petri 1962]. Ils sont particulièrement adaptés à l'étude des processus complexes mettant en jeu des propriétés de synchronisme et de partage de ressources. Leur support mathématique a permis en outre d'établir de nombreux résultats analytiques. Le lecteur intéressé par plus amples développements trouvera une présentation générale et des référence bibliographiques dans [Murata 1989], [Proth 1995] et [David 1992]. Pour l'étude des SED dans l'algèbre des dioïdes, les RdP sont souvent utilisés comme un outil de modélisation intermédiaire. La démarche consiste en effet à modéliser le système étudié en premier lieu par un réseau de Petri, puis à établir, à partir du graphe obtenu, les équations dans le dioïde propre à l'analyse de ce système.

2.2.1 Définitions et notations

Définition 2.10 Un réseau de Petri est un graphe biparti constitué de deux types de sommets : *places* (représentées par des cercles) et *transitions* (représentées par des barres). Des *arcs* orientés relient certaines places à certaines transitions, ou certaines transitions à certaines places. A chaque arc on associe un *poids* (entier positif). Le nombre de places (resp. de transitions) est fini, et non nul. Chaque place, peut contenir un ou plusieurs *jetons* (représentés par des points) qui modélisent la dynamique du système. A chaque place on associe une *temporisation*. Cette dernière correspond au temps de séjour des jetons dans une place.

D'une façon plus formelle, un RdP est un 5-tuplet $\mathcal{R} = (\mathcal{P}, \mathcal{T}, \text{Pré}, \text{Post}, \tau)$, où

- \mathcal{P} (resp., \mathcal{T}) est un ensemble non vide et fini de places (resp., transitions) ;
- $\text{Pré} : \mathcal{P} \times \mathcal{T} \longrightarrow \mathbb{N}$ une application d'incidence avant : $\text{Pré}(p; c_q)$ contient la valeur entière M_{qp} associée à l'arc allant de la place p à la transition c_q ;
- $\text{Post} : \mathcal{P} \times \mathcal{T} \longrightarrow \mathbb{N}$ une application d'incidence arrière : $\text{Post}(p; c_q)$ contient la valeur entière M_{pq} associée à l'arc allant de la transition c_q à la place p ;
- $\tau \in \mathbb{N}^{\mathcal{P}}$ définit les temporisations associées aux places : τ_p désigne la durée minimale de séjour d'un jeton dans la place p .

Le marquage \mathcal{M} d'un réseau de Petri est une application $\mathcal{M} : \mathcal{P} \longrightarrow \mathbb{N}$. La quantité $\mathcal{M}(p)$ détermine le marquage de la place p . Un réseau marqué est déterminé par le couple $\mathcal{N} = (\mathcal{R}, \mathcal{M}_0)$ formé d'un réseau de Petri \mathcal{R} et d'un marquage initial \mathcal{M}_0 .

Une transition sans place en amont est dite *transition source*, une transition sans place en aval est dite *transition puit*. La figure 4.2 représente un RdP ordinaire (tous les poids sont unitaires) dont les places sont représentées par des cercles et les transitions par des barres.

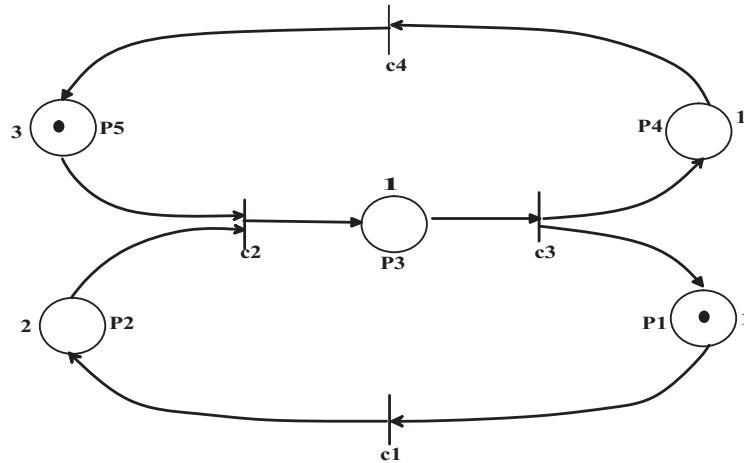


FIGURE 2.1 – Représentation graphique d'un RdP.

2.2.1.1 Equation fondamentale

L'évolution d'un RdP est décrit par le franchissement des transitions par des jetons séjournant dans les places du graphe.

Considérant un RdP et un marquage m , on dit que la transition $c_q \in \mathcal{T}$ est *franchissable* pour le marquage \mathcal{M} si l'on a :

$$\forall p \in \mathcal{P}, \mathcal{M}(p) \geq \text{Pré}(p; c_q).$$

Lorsque cette condition est validée, le *franchissement* de la transition c_q conduit à nouveau marquage \mathcal{M}' , qui est nouvel état du modèle, défini par :

$$\mathcal{M}'(p) = \mathcal{M}(p) - \text{Pré}(p; c_q) + \text{Post}(p; c_q).$$

Exemple 2.7 Considérons le RdP de la figure 4.2. En partant du marquage $\mathcal{M}_0^t = (1, 0, 0, 0, 1)$ ($\mathcal{M}(p_1) = 1, \mathcal{M}(p_2) = 0, \mathcal{M}(p_3) = 0, \mathcal{M}(p_4) = 0, \mathcal{M}(p_5) = 1$), on atteint le marquage $\mathcal{M}_1^t = (0, 1, 0, 0, 1)$, par franchissement de la transition c_1 .

Plus généralement, on note $[\mathcal{C}^+] = [\text{Post}(p; x_q)]$ (élément de la matrice d'incidence arrière) et $[\mathcal{C}^-] = [\text{Pré}(p; t_q)]$ (élément de la matrice d'incidence avant), $\mathcal{C} = \mathcal{C}^+ - \mathcal{C}^-$ (la matrice d'incidence) et en considérant \mathcal{F} une séquence de franchissement réalisable à partir d'un marquage \mathcal{M}_i , on a l'équation fondamentale suivante, qui traduit la dynamique de fonctionnement du RdP :

$$\mathcal{M}_k = \mathcal{M}_i + \mathcal{C} \times \bar{\mathcal{F}}, \quad (2.1)$$

$\bar{\mathcal{F}}$ est le vecteur caractéristique de la séquence de franchissement \mathcal{F} , c'est-à-dire un vecteur de dimension égale à $|\mathcal{T}|$; $\bar{\mathcal{F}}_q$ correspond au nombre de franchissement de la transition t_q dans la séquence \mathcal{F} . Dans la figure 4.2 où la séquence de tir était $\mathcal{F} = \{c_1\}$, le vecteur caractéristique est égal à $\bar{\mathcal{F}}^t = (1, 0, 0, 0)$.

2.2.1.2 Marquages accessibles

Soit $\mathcal{N} = (\mathcal{R}; \mathcal{M}_0)$ un réseau de Petri. L'ensemble des marquages accessibles $\mathcal{A}(\mathcal{R}; \mathcal{M}_0)$ d'un réseau de Petri marqué est l'ensemble des marquages que l'on peut atteindre à partir du marquage initial \mathcal{M}_0 par une séquence de franchissement, après un temps τ , c.-à-d.

$$\mathcal{A}(\mathcal{R}; \mathcal{M}_0) = \{\mathcal{M}_i, \exists c_q, \mathcal{M}_0 \xrightarrow{c_q/\tau} \mathcal{M}_i\}$$

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 $\mathcal{A}(\mathcal{R}; \mathcal{M}_0)$. Un arc orienté relie deux sommets \mathcal{M}_i et \mathcal{M}_j s'il existe une transition c_q franchissable permettant de passer d'un marquage à un autre après un temps $\tau : \mathcal{M}_i \xrightarrow{c_q} \mathcal{M}_j$. La figure 2.2 représente le graphe des marquages accessibles pour le réseau de Petri de la figure 4.2 avec un marquage initial $\mathcal{M}_0 = (1, 0, 0, 0, 1)$.

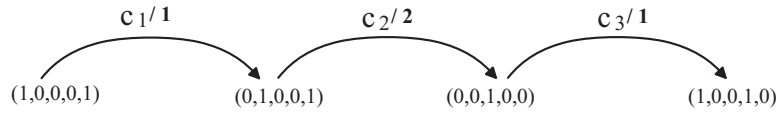


FIGURE 2.2 – Graphe des marquages atteignables du réseau de Petri de la figure 4.2.

2.2.1.3 Concurrency et synchronisation dans les RdP

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 2.3, nous avons représenté quatre structures de réseaux de Petri permettant de décrire ces phénomènes.

- la figure 2.3 (a) représente une structure de choix. La place p_1 a deux transitions en aval, notées c_1 et c_2 . Le tir de c_1 ou de c_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 2.3(b) permet de représenter une concurrence à l'apport de jetons dans une place, ici p_1 ;
- la configuration de la figure 2.3(c) permet de modéliser un phénomène de synchronisation. La transition c_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 2.3(d), l'apport de jetons dans les places p_1 et p_2 est synchronisé par le tir de la transition c_1 et permet, par exemple, de modéliser l'initiation simultanée de différentes tâches dans un système à événements discrets.

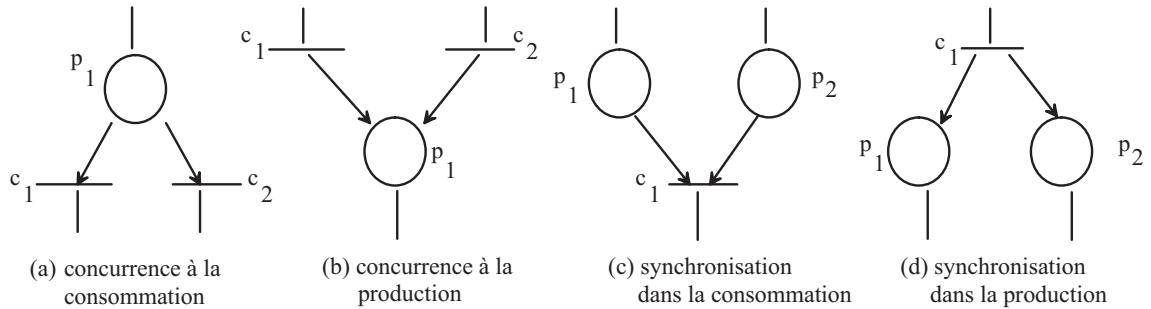


FIGURE 2.3 – Concurrency et synchronisation dans les RdP.

2.2.1.4 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, qu'il présente ou non des circuits, il reflète certaines des propriétés caractéristiques du système modélisé.

Définition 2.11 (Bornitude). Une place $p \in \mathcal{P}$ d'un réseau de Petri marqué $(\mathcal{R}; \mathcal{M}_0)$ est k -bornée ($k \in \mathcal{N}$) si pour tout marquage accessible $\mathcal{M} \in \mathcal{A}(\mathcal{R}; \mathcal{M}_0)$, le marquage de cette place vérifie $\mathcal{M}(p) \leq k$. Dans le cas contraire, nous dirons que p est non-bornée. Enfin si p est 1-bornée, on dit que p est binaire. Si on considère le réseau de Petri de la figure 4.2, on voit que pour le marquage initial \mathcal{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 stockages 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 2.12 (Vivacité). Étant donné un réseau de Petri \mathcal{R} et un marquage initial \mathcal{M}_0 , une transition t est vivante pour le réseau marqué $(\mathcal{R}; \mathcal{M}_0)$ si pour tout marquage \mathcal{M} accessible depuis \mathcal{M}_0 , il existe une suite de transitions \mathcal{F} , comportant au moins une fois la transition t , et telle que \mathcal{F} soit franchissable pour \mathcal{M} .

Un réseau de Petri marqué $(\mathcal{R}; \mathcal{M}_0)$ est dit vivant si toutes ses transitions sont vivantes.

Remarque 2.1 (L'intérêt d'étude de la vivacité d'un modèle). L'évolution du marquage d'un RdP se fait par franchissement de transitions. Lorsqu'au cours de son évolution, certaines transitions ne sont jamais franchies, cela indique que l'événement associé à la transition ne se produit pas et que le marquage d'une partie du RdP n'évolue pas. Cela indique que le sous système modélisé par cette partie-là ne fonctionnera pas. Il y a donc un problème au niveau de la conception du système. L'idée est d'être capable de détecter systématiquement ce phénomène par l'analyse de propriétés du modèle RdP, afin de disposer d'un outil d'aide à la conception des systèmes.

2.2.2 Invariants d'un réseau de Petri

Il existe deux types d'invariants (appelés semiflôts) dans un RdP [David 1992] : les *invariants de marquage* (notés également P-invariant) et les *invariants de franchissement* (notés également T-invariant).

Définition 2.13 Les invariants de marquage, appelés *P-invariant*, ou encore *P-semiflôts*, illustrent la conservation du nombre de jetons dans un sous ensemble de places du RdP. Un vecteur, noté \mathcal{K} , de dimension égale au nombre de places du RdP est un P-invariant si, et seulement si, il vérifie l'équation suivante :

$$\mathcal{K}^t \times \mathcal{C} = \vec{0}, \quad \mathcal{K} \neq \vec{0}, \quad (2.2)$$

où \mathcal{C} correspond à la matrice d'incidence du RdP.

D'après l'équation (2.1), on déduit que si \mathcal{K} est un P-invariant, alors pour un marquage donné, noté m_i , obtenu à partir d'un marquage initial m_0 , on a :

$$\mathcal{K}^t \times \mathcal{M}_i = \mathcal{K}^t \times \mathcal{M}_0 = n, \quad n \in \mathbb{N}^*.$$

Cette équation représente un invariant de marquage, elle signifie que si \mathcal{K} est un P-invariant du RdP alors la transposée du vecteur \mathcal{K} pondéré par le vecteur marquage m_i du RdP est une constante entière quel que soit le marquage m_i atteignable à partir du marquage initial m_0 . L'ensemble des places pour lesquelles la composante associée dans le P-invariant est non nulle, est appelée la *composante conservatrice* du RdP.

Définition 2.14 Un vecteur non nul d'entiers Θ de dimension $|\mathcal{T}| \times 1$ est un *T-invariant*, ou encore *T-semiflows*, du RdP si, et seulement s'il vérifie l'équation suivante :

$$\mathcal{C} \times \Theta = \vec{0}. \quad (2.3)$$

D'après l'équation (2.1), le franchissement à partir d'un marquage m_i d'une séquence dont le vecteur caractéristique est Θ ramène le graphe au même marquage $m_k = m_i$.

Un T-invariant correspondant à une séquence de franchissement réalisable est appelé *composante répétitive*. L'ensemble des transitions pour lesquelles la composante associée dans le T-invariant est non nulle est appelé le *support* du T-invariant.

Définition 2.15 Un RdP est dit *consistant* s'il possède un T-invariant Θ couvrant toutes les transitions du réseau. Un RdP qui possède cette propriété est dit *répétitif* ou *réinitialisable*. Le réseau atteint un régime périodique dès lors qu'il existe une séquence de franchissement réalisable avec Θ , comme vecteur caractéristique.

Définition 2.16 Un RdP est dit conservatif si toutes les places du graphe forment une composante conservatrice.

Remarque 2.2 • Un réseau de Petri consistant est dit fortement connexe si, et seulement s'il est conservatif.

- Si toutes les places d'un RdP forment une composante conservatrice, le graphe est dit *conservatif*.
- Toute combinaison de T-invariants (resp., P-invariants) est un T-invariant (resp., P-invariant).
- Un réseau de Petri consistant a un unique T-invariant élémentaire.
- Le produit des poids d'un circuit d'un réseau de Petri conservatif est égal à 1.

Exemple 2.8 Considérons le RdP de la figure 2.4 qui suit, un P-invariant est le vecteur \mathcal{K} vérifiant :

$$\mathcal{K}^t \times \mathcal{C} = (k_1, k_2, k_3, k_4, k_5) \times \begin{pmatrix} -1 & 0 & 2 \\ 1 & -3 & 0 \\ 0 & 3 & -2 \\ 0 & -3 & 2 \\ 0 & 0 & 0 \end{pmatrix} = \vec{0}.$$

Il existe un P-invariant : $\mathcal{K}^t = (0, 0, 1, 1, 1)$. L'invariant de marquage est : $1 \times m(p_3) + 1 \times m(p_4) + 1 \times m(p_5) = 6$. Ceci signifie que les places p_1 et p_2 forment une composante conservative du graphe.

Le RdP de la figure 2.4 admet un T-invariant, $\Theta^t = (6, 2, 3)$; vérifiant :

$$\mathcal{C} \times \Theta = \begin{pmatrix} -1 & 0 & 2 \\ 1 & -3 & 0 \\ 0 & 3 & -2 \\ 0 & -3 & 2 \\ 0 & 0 & 0 \end{pmatrix} \times \begin{pmatrix} \Theta_1 \\ \Theta_2 \\ \Theta_3 \end{pmatrix} = \vec{0}.$$

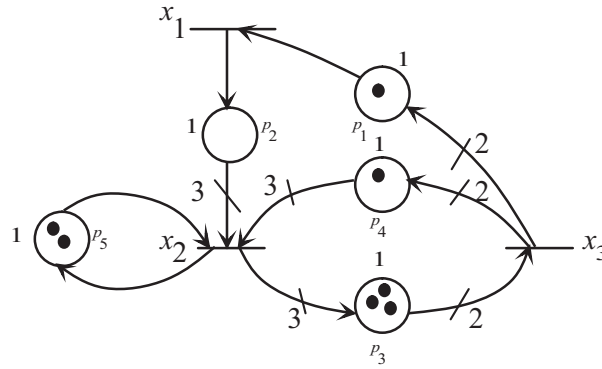


FIGURE 2.4 – Réseau de Petri à arcs pondérés.

Définition 2.17 Un RdP est dit k -serveur si, pour chaque transition, il ne peut y avoir au maximum que k franchissements simultanés. Une boucle de ré-entrance

implicite contenant k jetons est alors associée à chaque transition du graphe.

- Dans le cas où $k = 1$, le RdP est dit *mono-seveur*.
- Dans le cas où le nombre de franchissement de transitions n'est pas limité, le RdP est dit *infinité-serveur*.

Dans la suite, on note $\bullet q$ (resp., $q\bullet$) l'ensemble des places situées immédiatement en amont (resp., en aval) de la transition t_q . Similairement, $\bullet p$ (resp., $p\bullet$) représente l'ensemble des transitions situées immédiatement en amont (resp., en aval) de la place p .

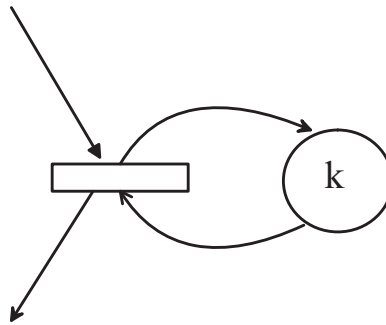


FIGURE 2.5 – Boucle de ré-entrée.

2.3 Graphes d'Événements Temporisés

Suite à cette présentation succincte des réseaux de Petri, notre intérêt va maintenant se porter sur une sous-classe des réseaux de Petri qui permet de modéliser les systèmes où interviennent des phénomènes de synchronisation et de délai (les phénomènes de conflits étant supposés résolus au préalable).

Définition 2.18 Un Graphe d'Événements Temporisé (GET) est un RdP tel que toute place a exactement une transition en amont et une transition en aval.

Pour illustrer ce type de modèles temporisés, nous proposons l'exemple d'un graphe d'événements temporisé représenté par la figure 2.6.

Exemple 2.9 Le GET 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 deux 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 (u.t). 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.

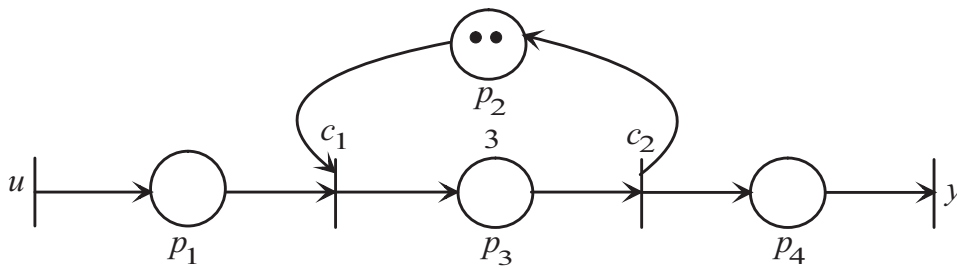


FIGURE 2.6 – Modèle GET d'une cellule de production.

2.3.1 Conditions initiales

Définition 2.19 (Conditions initiales canoniques). On dira que les conditions initiales d'un GET sont *canoniques* si tous les jetons du marquage initial sont considérés comme disponibles depuis l'instant $-\infty$.

Considérer des conditions initiales canoniques, implique de prendre pour hypothèse que le marquage initial du graphe n'est pas nécessairement égal au marquage à l'instant initial d'observation, (égal à 0 par convention). En effet, si les jetons initiaux sont supposés disponibles depuis $-\infty$, ils sont susceptibles d'avoir validé et tiré des transitions également à $-\infty$, aussi le marquage peut avoir évolué avant l'instant initial d'observation.

Exemple 2.10 La figure 2.7 représente un GET. Si on considère des conditions initiales canoniques, le marquage à l'instant initial d'observation est différent du marquage initial du graphe. En effet, le jeton contenu dans la place p_3 a validé et tiré la transition c_3 à l'instant $t = -\infty$. Un jeton a alors été ajouté dans les places

p_4 et p_5 . Finalement, le marquage, après qu'il ait évolué librement depuis l'instant $t = -\infty$, à l'instant d'observation est celui représenté sur la figure 2.8.

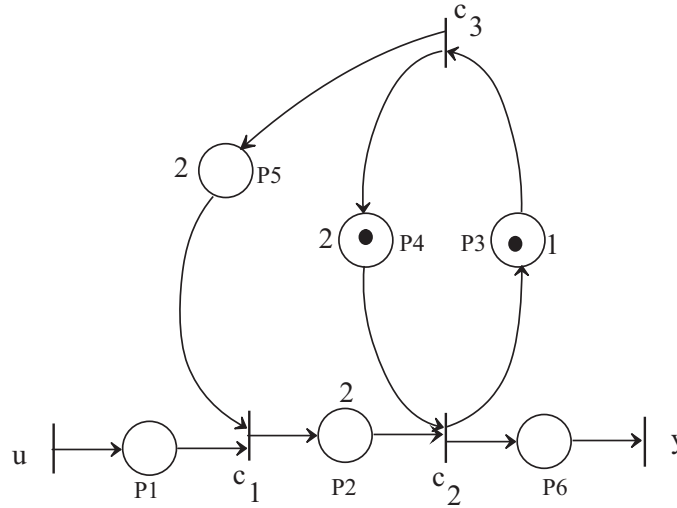


FIGURE 2.7 – Marquage initial ($t = -\infty$).

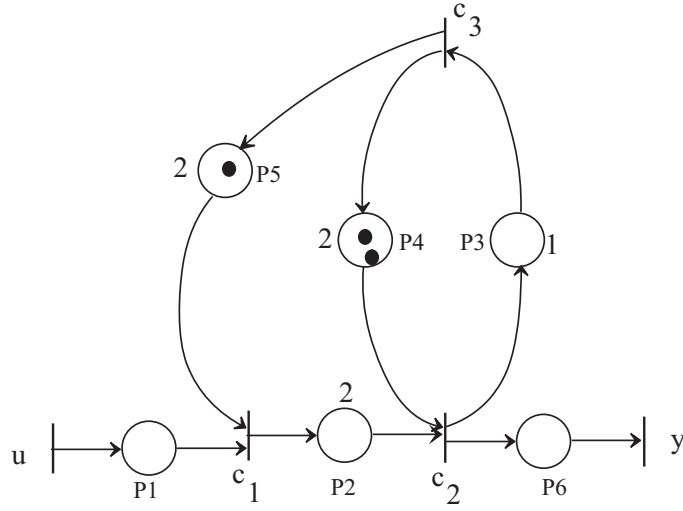
Définition 2.20 (Conditions initiales faiblement compatibles non canoniques). On dira que les conditions initiales d'un GET sont faiblement compatibles si tout jeton du marquage initial est disponible qu'à partir de à l'instant $t = 0$. Pour $t < 0$, le graphe est considéré comme étant "gelé".

Les jetons visibles à l'instant initial d'observation $t = 0$ sont alors ceux du marquage initial. Ces jetons sont supposés avoir été introduits dans le graphe avant l'instant 0, de telle sorte qu'à l'instant $t = 0$ un jeton du marquage initial est :

- soit en train de subir la temporisation de la place dans laquelle il a été introduit,
- soit tout juste disponible pour le tir de la transition aval.

2.3.2 Modélisation des GET dans l'algèbre $(\min, +)$

Dans le cas des RdP, l'état modélisant l'évolution du modèle est associé aux marquages des places. Si l'on se réfère à la figure 2.6, la seule caractéristique

FIGURE 2.8 – Marquage à l'instant initial d'observation ($t=0$).

que traduit une telle représentation concerne l'état de la machine M (au sens logique) : machine libre ou occupée. Pour pouvoir discuter des performances du système temporisé, c'est-à-dire déterminer son régime transitoire et/ou son régime permanent, l'équipe $(\max, +)$ de l'INRIA a proposé dans [Cohen 1985] une modélisation considérant non pas l'état du marquage mais les compteurs et les dates de franchissements des transitions.

Pour étudier les GET dans l'algèbre $(\min, +)$, Les variables d'état considérées sont de type compteurs d'événements. De telles variables permettent d'aboutir à une représentation linéaire de ces modèles dans les dioïdes. On a privilégié la représentation compteur à celle de dateur (*cf.* remarque 2.4) du fait des dioïdes inspirés de l'algèbre $(\min, +)$ qui seront utilisés par la suite pour représenter des Graphes d'Événements Temporisés Généralisés.

Définition 2.21 Un *compteur* associé à une transition c_j est une application croissante, notée $c_j(t)$, de $\mathbb{Z} \rightarrow \mathbb{Z} \cup \{\pm\infty\}$, $t \mapsto c_j(t)$, où $c_j(t) \in \mathbb{Z} \cup \{\pm\infty\}$ correspond au nombre de tirs de la transition c_j ayant lieu jusqu'à la date t .

Le vecteur d'état pour un GET, est défini par $C(t) = (c_1(t) \dots c_{|\mathcal{T}|}(t))^t$. Contrai-

remment aux marquages, ces compteurs ont une propriété de monotonie :

$$\forall t \in \mathbb{R}, \forall \beta \geq 0, \forall c_j \in \mathcal{T}, c_j(t + \beta) \geq c_j(t).$$

Remarque 2.3 Comme le franchissement des transitions est ici supposé être instantané, il n'y a pas à formuler d'hypothèses sur les tirs des transitions, leur fonctionnement est évidemment de type FIFO¹ (First-In, First-Out).

Pour illustrer l'évolution d'un compteur associé à la transition c_j d'un GET, considérons le GET élémentaire suivant :

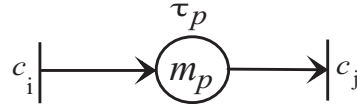


FIGURE 2.9 – GET élémentaire.

Sous l'hypothèse de fonctionnement au plus tôt, qui est un comportement limite du GET, l'évolution du compteur $c_j(t)$, associé à la transition c_j du GET élémentaire de la figure 2.9, est modélisé par l'équation suivante :

$$c_j(t) = m_p + c_i(t - \tau_p). \quad (2.4)$$

Dans un GET, le phénomène de synchronisation se produit lorsque plusieurs arcs convergent sur une même transition (cf. figure 2.10).

En général, l'évolution d'un compteur associé à une transition de synchronisation d'un GET est donnée par l'équation suivante :

$$c_j(t) = \min_{p \in \bullet_j, i \in \bullet_p} (m_p + c_i(t - \tau_p)). \quad (2.5)$$

On remarque que cette équation est non linéaire dans l'algèbre usuelle de par la présence de l'opérateur min qui modélise le phénomène de synchronisation au

1. Le n -ième jeton rentré dans une place sera également le n -ième jeton à contribuer au franchissement de la transition en aval à cette place.

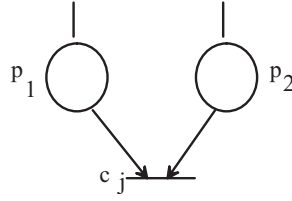


FIGURE 2.10 – Phénomène de synchronisation dans un GET.

niveau de la transition c_j . Par contre, elle se décrit de façon linéaire dans le dioïde $\overline{\mathbb{Z}}_{min}$. Le compteur $c_j(t)$ vérifie l'équation récurrente suivante :

$$c_j(t) = \bigoplus_{p \in \bullet j, i \in \bullet p} (m_p \otimes c_i(t - \tau_p)). \quad (2.6)$$

L'opération \bigoplus du dioïde $\overline{\mathbb{Z}}_{min}$ représente un phénomène de synchronisation au niveau de la transition c_j , et l'opération \otimes représente un décalage *événementiel* du compteur $c_j(t)$ par rapport à tout compteur associé à une transition en amont de la transition c_j .

D'une manière générale, les équations modélisant le fonctionnement au plus tôt des GET ont la forme suivante dans le dioïde $\overline{\mathbb{Z}}_{min}$:

$$\begin{cases} C(t) &= \bigoplus_{a=0}^T C_a X(t-a) \bigoplus \bigoplus_{a=0}^T B_a U(t-a), \\ Y(t) &= \bigoplus_{a=0}^T C_a C(t-a), \end{cases} \quad (2.7)$$

avec $T = \max_{p_a \in \mathcal{P}}(\tau(p_a))$. $C(t)$ est le vecteur d'état, composé des compteurs associés aux transitions internes du graphe. $U(t)$ et $Y(t)$ sont respectivement les vecteurs d'entrées (commandes) et de sorties, composés des transitions sources, respectivement puits, du graphe.

Exemple 2.11 Dans le dioïde $\overline{\mathbb{Z}}_{min}$, l'évolution des compteurs du GET de la figure 2.11 sont régies par les relations suivantes :

$$\begin{cases} c_1(t) &= 1 \otimes c_1(t-1) \bigoplus 2 \otimes c_2(t) \bigoplus u(t), \\ c_2(t) &= c_1(t-2), \\ y(t) &= c_2(t). \end{cases}$$

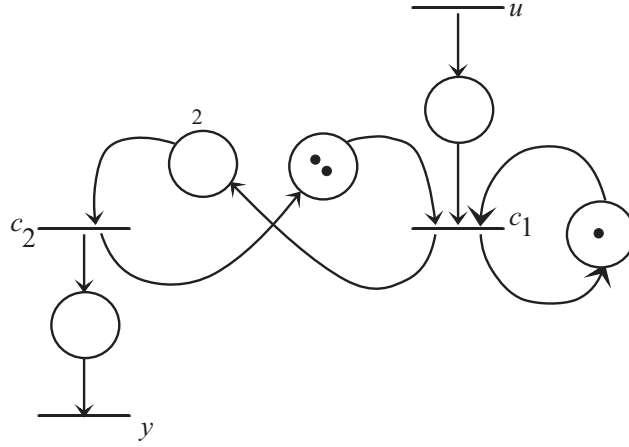


FIGURE 2.11 – Exemple d'un GET.

Il en résulte la représentation d'état suivante :

$$\begin{cases} C(t) = \begin{pmatrix} \varepsilon & 2 \\ \varepsilon & \varepsilon \end{pmatrix} C(t) \oplus \begin{pmatrix} 1 & \varepsilon \\ \varepsilon & \varepsilon \end{pmatrix} C(t-1) \oplus \begin{pmatrix} \varepsilon & \varepsilon \\ e & \varepsilon \end{pmatrix} C(t-2) \oplus \begin{pmatrix} e \\ \varepsilon \end{pmatrix} U(t), \\ Y(t) = \begin{pmatrix} \varepsilon & e \end{pmatrix} C(t), \end{cases}$$

avec $C = (c_1 \ c_2)^t$, $U = u$ et $Y = y$.

2.3.3 Forme récurrente explicite

La représentation d'état sous une forme ARMA (Auto Regressive Moving Average) donnée par la représentation (2.7) peut se ramener, moyennant une extension de la dimension du vecteur d'état, à une forme récurrente markovienne du type :

$$\begin{cases} C(t) &= AC(t-1) \oplus BU(t), \\ Y(t) &= CX(t). \end{cases} \quad (2.8)$$

Le GET équivalent auquel va correspondre une représentation de la forme (2.8) est tel que :

- la temporisation d'une place située entre deux transitions internes doit être égale à un ;

- la temporisation d'une place située entre une transition source et une transition interne, ou entre une transition interne et une transition puits, doit être nulle.

Exemple 2.12 La figure 2.12 représente un GET dont le comportement entrée-sortie est équivalent à celui de la figure 2.11. En introduisant un nouvel état c'_1 , on aboutit à la représentation d'état suivante sur le dioïde $\overline{\mathbb{Z}}_{\min}$:

$$\begin{cases} C(t) = \begin{pmatrix} \varepsilon & \varepsilon & 2 \\ \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon \end{pmatrix} C(t) \oplus \begin{pmatrix} 1 & \varepsilon & \varepsilon \\ e & \varepsilon & \varepsilon \\ \varepsilon & e & \varepsilon \end{pmatrix} C(t-1) \oplus \begin{pmatrix} e \\ \varepsilon \\ \varepsilon \end{pmatrix} U(t), \\ Y(t) = \begin{pmatrix} \varepsilon & \varepsilon & e \end{pmatrix} C(t), \end{cases}$$

avec $C = (c_1 \ c'_1 \ c_2)^t$, $U = u$ et $Y = y$.

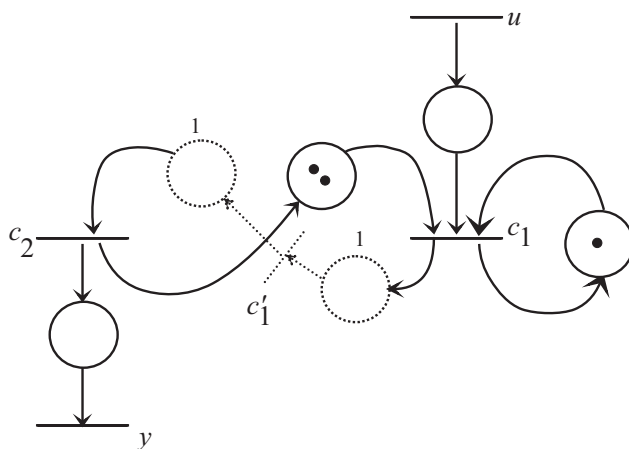
On obtient, après utilisation du théorème 1.1 pour supprimer la partie implicite de l'équation d'état, la forme récurrente markovienne suivante :

$$\begin{cases} C(t) = \begin{pmatrix} 1 & 2 & \varepsilon \\ e & \varepsilon & \varepsilon \\ \varepsilon & e & \varepsilon \end{pmatrix} C(t-1) \oplus \begin{pmatrix} e \\ \varepsilon \\ \varepsilon \end{pmatrix} U(t), \\ Y(t) = \begin{pmatrix} \varepsilon & \varepsilon & e \end{pmatrix} C(t), \end{cases}$$

Remarque 2.4 (Equations aux dateurs). Au lieu de considérer les nombres cumulés de tirs d'une transition d'un GET à chaque instant, on s'intéresse aux différentes dates de tirs de cette transition. A chaque transition c_i , on associe l'application $d_i : \mathbb{Z} \rightarrow \overline{\mathbb{Z}}_{\max}$, $k \mapsto_i (k)$ où $d_i(k)$ désigne la date du k -ième tir de la transition d_i . Cette application est non décroissante et est appelée *dateur*.

Par un raisonnement analogue à celui effectué pour les compteurs, on montre que la description du comportement d'un GET peut se représenter *via* des équations $(\max, +)$ linéaires.

Le graphe de la figure 2.11 peut être modélisé par les équations $(\max, +)$ linéaires suivantes :


$$\begin{cases} d_1(k) &= 1 \otimes d_1(k-1) \oplus d_2(k-2) \oplus u(k), \\ d_2(k) &= 2 \otimes d_1(k), \\ y(k) &= d_2(k). \end{cases}$$

2.4.1 Présentation des GETG

Définition 2.22 Un GETG est un GET dans lequel des poids (nombres entiers strictement positifs) sont associés aux arcs.

Dans ce type de graphe, dont un exemple élémentaire est donné par la figure 2.13, le poids $M_{pq'}$ rattaché à l'arc $x_{q'} \rightarrow p$ est tel que lors d'un franchissement de la transition $x_{q'}$, $M_{pq'}$ jetons sont ajoutés à la place p . Le poids M_{qp} associé à l'arc $p \rightarrow x_q$ implique que la transition x_q ne sera validée que si la place p contient au

moins M_{qp} jetons. Lors du franchissement de cette transition, après un séjour de τ_p unités de temps dans la place p , M_{qp} jetons y seront retirés.

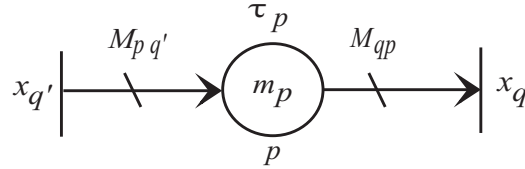


FIGURE 2.13 – GETG élémentaire.

La figure 2.14 représente un GETG tel que l'arc $p_1 \rightarrow x_1$ a un poids de 2, et l'arc $x_1 \rightarrow p_4$ a un poids de 3. Les arcs dont le poids n'est pas explicitement spécifié ont un poids égal à 1. La transition x_1 est validée puisque la place p_1 contient 2 jetons et la place p_2 contient un jeton. Après un séjour de deux unités de temps des jetons dans la place p_1 , le franchissement de x_1 consiste à retirer deux jetons de p_1 et un jeton de p_2 , et à ajouter un jeton à p_3 et trois jetons à p_4 . On obtient le marquage de la figure 2.14.b.

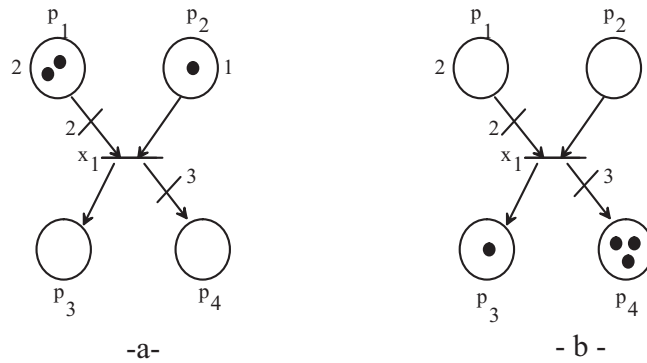


FIGURE 2.14 – Evolution du marquage.

Définition 2.23 (Gain d'un circuit [Teruel 1992]). Le gain d'un circuit ϱ , noté $G(\varrho)$, est le produit du quotient des poids des arcs entrants par le produit des poids des arcs sortants de toutes les places du circuit ϱ . Formellement, ce gain est défini

comme suit :

$$G(\varrho) = \prod_{P_i \in \varrho} \frac{a_i}{b_i},$$

où :

- a_i est égal au poids de l'arc entrant dans la place p_i ,
- b_i est égal au poids de l'arc sortant de la place p_i .

Selon la valeur de son gain, un circuit ϱ est classé dans l'une des trois catégories suivantes :

- $G(\varrho) > 1$: circuit générateur,
- $G(\varrho) = 1$: circuit neutre,
- $G(\varrho) < 1$: circuit absorbant.

La valeur de $G(\varrho)$ informe sur la conservation des marques dans le circuit ϱ . Si un circuit est générateur, alors le nombre de marques dans le circuit augmente avec l'évolution du marquage. Si le circuit est neutre, ce gain reste constant au cours de l'évolution du marquage. Enfin, si le circuit est absorbant, le nombre de marque diminue.

Dans le cas des GET ordinaires, le nombre total de jetons dans un circuit est constant. De par la valuation des arcs dans \mathbb{N}^* , ceci n'est plus vrai dans le cas des GETG. Chao et al. [Chao 1993] ont utilisé un principe similaire en introduisant la notion de marques valuées des places d'un circuit. La *marque valuée*, notée \mathcal{N}_p , d'une place p d'un circuit peut se calculer comme suit :

$$\mathcal{N}_p = \frac{m_p}{M_{qp}\theta_q}, \quad (2.9)$$

où :

- m_p correspond au marquage initial de la place p ,
- M_{qp} correspond au poids de l'arc sortant de la place p ,
- θ_q correspond à la composante de T-invariant associée à la transition en aval de la place p .

La *marque valuée totale* d'un circuit neutre, ϱ , est donnée par la somme des marques valuées des places de ce circuit. Elle ne dépend pas du temps et est constante quelque soit l'évolution du circuit. Par conséquent, elle est généralement calculé pour le marquage initial M_0 et est donnée par :

$$\mathcal{N}(\varrho) = \sum_{p \in \varrho} \mathcal{N}_p. \quad (2.10)$$

L'étude de la vivacité des GETG est difficile. Des résultats ont été obtenus dans ce domaine par D.T. Chao [Chao 1993] : un circuit d'un GETG est vivant s'il contient initialement au moins une place marquée dont le marquage valué est supérieur à un ($N(\varrho) \geq 1$).

Contrairement aux GET ordinaires, il n'existe pas de condition nécessaire et suffisante pour vérifier la vivacité des GETG.

2.4.2 Equations récurrentes des GETG

Comme dans le cas des GET ordinaires, on se propose d'étudier les GETG à travers leurs représentations d'état de type compteur.

Assertion 1 L'évolution du compteur associé à la transition x_q du GETG élémentaire de la figure 2.13 est donnée par l'équation suivante :

$$x_q(t) = \lfloor M_{qp}^{-1}(m_p + M_{pq'}x_{q'}(t - \tau_p)) \rfloor. \quad (2.11)$$

La présence de la partie entière inférieure dans l'équation (2.11) assure la valeur discrète du compteur indépendamment de la valeur M_{qp} . En général, une transition x_q peut avoir plusieurs transitions en amont $\{x_{q'} \in \bullet\bullet q\}$, ce qui implique que l'équation compteur associée fasse intervenir le min entre les compteurs associés aux transitions en amont.

Exemple 2.13 La variable compteur associée à la transition x_q du GETG représenté par la figure 2.15 satisfait l'équation suivante :

$$x_q(t) = \lfloor \frac{2 + 3x_{q'}(t - 1)}{2} \rfloor.$$

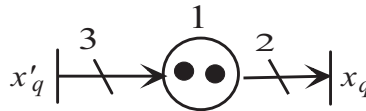
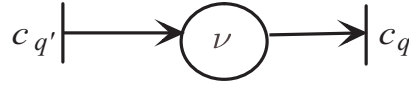


FIGURE 2.15 – Exemple de GETG élémentaire.

1. **Opérateur de stock γ^ν** : Il représente un décalage événementiel de ν événements ($\nu \in \overline{\mathbb{Z}}$), et modélise un système gain élémentaire.

FIGURE 2.17 – Opérateur γ^ν .

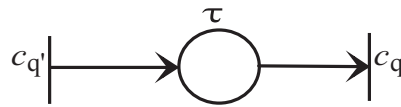
Il est défini comme suit :

$$\forall t \in \mathbb{Z}, \forall c_{q'} \in \overline{\mathbb{Z}}, \quad c_q(t) = \gamma^\nu c_{q'}(t) = c_{q'}(t) + \nu.$$

L'opérateur γ^ν satisfait les règles suivantes :

- $(\gamma^\nu \oplus \gamma^{\nu'})c_q(t) = \gamma^{\min(\nu, \nu')}c_q(t)$.
En effet, on a $(\gamma^\nu \oplus \gamma^{\nu'})c_q(t) = \min(c_q(t) + \nu, c_q(t) + \nu') = c_q(t) + \min(\nu, \nu') = \gamma^{\min(\nu, \nu')}c_q(t)$.
- $(\gamma^\nu \otimes \gamma^{\nu'})c_q(t) = \gamma^{\nu+\nu'}c_q(t)$.
En effet, on a $(\gamma^\nu \otimes \gamma^{\nu'})c_q(t) = \gamma^\nu(c_q(t) + \nu') = c_q(t) + \nu' + \nu = \gamma^{\nu+\nu'}c_q(t)$.

2. **Opérateur de retard δ^τ** : Il représente un retard temporel de τ unités de temps ($\tau \in \overline{\mathbb{Z}}$), et modélise un système retard élémentaire.

FIGURE 2.18 – Opérateur δ^τ .

Il est défini comme suit :

$$\forall t \in \mathbb{Z}, \forall c_{q'} \in \overline{\mathbb{Z}}, \quad c_q(t) = \delta^\tau c_{q'}(t) = c_{q'}(t - \tau).$$

L'opérateur δ^τ satisfait les règles suivantes :

- $(\delta^\tau \oplus \delta^{\tau'})c_q(t) = \delta^{\max(\tau, \tau')}c_q(t)$.
Du fait que le signal $c_q(t)$ soit non décroissant, on a : $(\delta^\tau \oplus \delta^{\tau'})c_q(t) = \min(c_q(t - \tau), c_q(t - \tau')) = c_q(t - \max(\tau, \tau')) = \delta^{\max(\tau, \tau')}c_q(t)$.

- $(\delta^\tau \otimes \delta^{\tau'})c_q(t) = \delta^{\tau+\tau'}c_q(t)$.

En effet, on a $(\delta^\tau \otimes \delta^{\tau'})c_q(t) = \delta^\tau c_q(t - \tau') = c_q(t - \tau' - \tau) = \delta^{\tau'+\tau}c_q(t)$.

3. **Opérateur multiplieur** μ_α : Il représente un facteur de mise à l'échelle m .

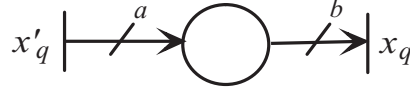


FIGURE 2.19 – Opérateur μ_α avec $m = \frac{a}{b}$.

Il est défini comme suit :

$\forall t \in \mathbb{Z}, \forall x_{q'} \in \overline{\mathbb{Z}}, \quad x_q(t) = \mu_m x_{q'}(t) = \lfloor m \times x_{q'}(t) \rfloor$, avec $m \in \mathbb{Q}^+$ (α est égal au quotient de deux nombres naturels strictement positifs).

Remarque 2.5 Contrairement aux opérateurs γ^ν et δ^τ , l'opérateur μ_α n'admet pas de règles de simplification pour l'opération \otimes :

$\forall \alpha, \alpha' \in \mathbb{Q}^+, \forall x_q \in \overline{\mathbb{Z}}, (\mu_\alpha \otimes \mu_{\alpha'})x_q(t)$ peut être différent de $(\mu_{\alpha \times \alpha'})x_q(t)$.

L'opérateur μ_α satisfait les règles suivantes lorsqu'il est composé avec les opérateurs δ^τ et γ^ν :

- $(\mu_\alpha \otimes \delta^\tau)x(t) = (\delta^\tau \otimes \mu_m)x(t)$,

En effet, on a : $(\mu_\alpha \otimes \delta^\tau)x(t) = \lfloor \alpha \times x(t - \tau) \rfloor = (\delta^\tau \otimes \mu_\alpha)x(t)$.

- $(\mu_\alpha \otimes \gamma^\nu)x(t) = (\gamma^{\nu \times \alpha} \otimes \mu_\alpha)x(t)$, pour $\nu \in \alpha^{-1} \times \mathbb{N}$.

En effet, on a : $\forall \nu \in \alpha^{-1} \times \mathbb{N}, (\mu_\alpha \otimes \gamma^\nu)x(t) = \lfloor \alpha \times \nu + \alpha \times x(t) \rfloor = \alpha \times \nu + \lfloor \alpha \times x(t) \rfloor = (\gamma^{\nu \times \alpha} \otimes \mu_\alpha)x(t)$, puisque $\nu \times m \in \mathbb{N}$.

Si les opérateurs γ^ν et δ^τ , habituellement utilisés dans le cas des GET ordinaires, sont linéaires, l'opérateur μ_α est, par contre, seulement additif. En effet, on a :

$\forall t \in \mathbb{Z}, \forall x_1, x_2 \in \overline{\mathbb{Z}},$

$$\mu_\alpha(x_1(t) \oplus x_2(t)) = \lfloor \alpha \times (\min(x_1(t), x_2(t))) \rfloor = \min(\lfloor \alpha \times x_1(t) \rfloor, \lfloor \alpha \times x_2(t) \rfloor) = \mu_\alpha x_1(t) \oplus \mu_\alpha x_2(t).$$

La non linéarité de l'opérateur μ_α est due à sa non homogénéité, en effet :

$$\forall \lambda \in \mathbb{N}, \quad \lambda \otimes (\mu_\alpha x_1(t)) = \lambda + \lfloor \alpha \times x_1(t) \rfloor$$

est différent de

$$\mu_\alpha(\lambda \otimes x_1(t)) = \lfloor \alpha \times (\lambda + x_1(t)) \rfloor.$$

Présentons à présent le dioïde d'opérateurs permettant de modéliser les GETG.

Pour cela, on note \mathcal{O}_{\min} , non commutatif de sommes finies de composition d'opérateurs μ, γ , muni des deux lois suivantes : \min pour l'opération (\oplus) et la composition pour l'opération (\otimes) , avec pour éléments neutres $\varepsilon = \mu_{+\infty} \gamma^{+\infty}$ et $e = \mu_1 \gamma^0$.

Un élément de \mathcal{O}_{\min} est une application, notée $g = \bigoplus_{i=1}^k \mu_{\alpha_i} \gamma^{\nu_i}$, définie de $\overline{\mathbb{Z}} \rightarrow \overline{\mathbb{Z}}$, telle que :

$$\forall t \in \mathbb{Z}, \quad g(x(t)) = \min_{1 \leq i \leq k} (\lfloor \alpha_i (\nu_i + x(t)) \rfloor).$$

Utilisons l'opérateur δ pour définir un dioïde de séries formelles. Un élément de \mathcal{O}_{\min} est associé à chaque valeur de τ . Pour cela, on définit l'application h telle que :

$$h : \mathbb{Z} \rightarrow \mathcal{O}_{\min}$$

$$\tau \mapsto h(\tau) = \bigoplus_{i=1}^{k_\tau} \mu_{\alpha_i^\tau} \gamma^{\nu_i^\tau}.$$

Une telle application peut être représentée par une série formelle à coefficients dans \mathcal{O}_{\min} et dont l'indéterminée est δ . Elle est définie comme suit :

$$H(\delta) = \bigoplus_{\tau \in \mathbb{Z}} h(\tau) \delta^\tau,$$

où l'opérateur \bigoplus correspond à la somme point à point pour $\tau \in \mathbb{Z}$.

L'ensemble de ces séries formelles, muni des deux opérations suivantes :

- $P(\delta) \oplus H(\delta) : (p \oplus h)(\tau) = p(\tau) \oplus h(\tau) :$

$$\overline{\mathbb{Z}} \rightarrow \overline{\mathbb{Z}}, \{x(t)\}_{t \in \mathbb{Z}} \mapsto \{\min(p(\tau)(x(t)), h(\tau)(x(t)))\}_{t \in \mathbb{Z}},$$

- $P(\delta) \otimes H(\delta) : (p \otimes h)(\tau) = \bigoplus_{i \in \mathbb{Z}} p(i) \otimes h(\tau - i) :$

$$\overline{\mathbb{Z}} \rightarrow \overline{\mathbb{Z}}, \{x(t)\}_{t \in \mathbb{Z}} \mapsto \{\inf_{i \in \mathbb{Z}} (p(i) + g(\tau - i))(x(t))\}_{t \in \mathbb{Z}},$$

est un dioïde, noté $\mathcal{O}_{\min}[\![\delta]\!]$. Ce dioïde a pour éléments neutres $\varepsilon = \mu_{+\infty}\gamma^{+\infty}\delta^{-\infty}$ et $e = \mu_1\gamma^0\delta^0$.

Un élément de $\mathcal{O}_{\min}[\![\delta]\!]$ permet de coder un signal $x(t)$ sous forme d'une série formelle en δ , et il permet également de modéliser la relation liant deux transitions d'un GETG.

Exemple 2.15 Dans le dioïde $\mathcal{O}_{\min}[\![\delta]\!]$, le compteur $x_q(t)$, associé à la transition x_q du GETG élémentaire de la figure 2.13, satisfait l'équation suivante :

$$X_q(\delta) = \mu_{M_{qp}^{-1}}\gamma^{m_p}\delta^{\tau_p}\mu_{M_{pq'}}X_{q'}(\delta). \quad (2.12)$$

Physiquement, le compteur $X_q(\delta)$ est égal au compteur $X_{q'}(\delta)$ composé avec les systèmes élémentaires décrits dans la figure 2.20.

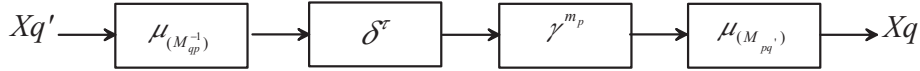


FIGURE 2.20 – Composition en série de systèmes élémentaires.

Pour $X_{q'}$ correspondant à une impulsion, notée $E(\delta)$, et égale à $e = \mu_1\gamma^0\delta^0$, on obtient la relation liant la transition x_q et $x_{q'}$ sous forme d'une composition d'opérateurs donnée par $\mu_{M_{qp}^{-1}}\gamma^{m_p}\delta^{\tau_p}\mu_{M_{pq'}}$.

Assertion 2 L'évolution de la dynamique d'un GETG est donnée par la représentation d'état suivante :

$$\begin{cases} X(\delta) = A X(\delta) \oplus B U(\delta), \\ Y(\delta) = C X(\delta) \oplus D U(\delta), \end{cases} \quad (2.13)$$

où $X(\delta)$ est le vecteur d'état composé des compteurs associés aux transitions internes du graphe, codés dans le dioïde $\mathcal{O}_{\min}[\![\delta]\!]$. $U(\delta)$ et $Y(\delta)$ sont les vecteurs d'entrées et de sorties codés dans $\mathcal{O}_{\min}[\![\delta]\!]$.

Les matrices A , B , C et D sont composées de monômes de séries formelles en δ , chaque monôme modélise une relation liant deux transitions du graphe dans $\mathcal{O}_{\min}[[\delta]]$.

Exemple 2.16 Dans le dioïde $\mathcal{O}_{\min}[[\delta]]$, le GETG de la figure 2.16 admet la représentation d'état suivante :

$$\left\{ \begin{array}{l} \begin{pmatrix} X_1(\delta) \\ X_2(\delta) \\ X_3(\delta) \end{pmatrix} = \begin{pmatrix} \varepsilon & \varepsilon & \gamma^3 \delta^2 \\ \mu_{\frac{1}{3}} \delta^2 \mu_2 & \varepsilon & \mu_{\frac{1}{3}} \gamma^3 \delta^2 \mu_2 \\ \varepsilon & \gamma^1 \delta^1 \mu_3 & \varepsilon \end{pmatrix} \begin{pmatrix} X_1(\delta) \\ X_2(\delta) \\ X_3(\delta) \end{pmatrix} \oplus \begin{pmatrix} e \\ \varepsilon \\ \varepsilon \end{pmatrix} U(\delta), \\ \\ Y(\delta) = \begin{pmatrix} \varepsilon & e & \varepsilon \end{pmatrix} \begin{pmatrix} X_1(\delta) \\ X_2(\delta) \\ X_3(\delta) \end{pmatrix} . \end{array} \right. \quad (2.14)$$

Une telle représentation est décrite dans l'algèbre usuelle dans l'exemple 3.17.

2.5 Conclusion

Contrairement aux GET ordinaires, qui admettent une représentation linéaire dans l'algèbre $(\min, +)$ [Baccelli 1992], les GETG, de par leurs poids sur les arcs, se modélisent d'une façon non linéaire dans cette algèbre. Pour palier à ce problème de non linéarité et de pouvoir appliquer certains résultats développés dans le cadre de l'analyse de performances des systèmes linéaires dans l'algèbre des dioïdes, une approche de modélisation de ces graphes a été utilisée [Cohen 1998a]. Cette approche est à base d'opérateurs, elle permet de décrire la dynamique du modèle et les relations reliant les différentes transitions du graphe. Le modèle d'état obtenu sera utilisé par la suite pour évaluer les performances des GETG.

Evaluation de Performances des GETG

Sommaire

3.1	Introduction et état de l'art	47
3.2	Notions élémentaires	53
3.3	Eléments de la théorie spectrale	57
3.4	Evaluation de performances des GET	61
3.5	Evaluation de performances des GETG	64
3.5.1	Méthode de linéarisation proposée	64
3.5.2	Exemple : Extension au cas des GETG avec entrée/sortie . .	73
3.6	Conclusion	75

3.1 Introduction et état de l'art

L'évaluation des paramètres de performances d'un modèle, peut se faire par trois méthodes : la simulation, les méthodes numériques et les méthodes analytiques. La simulation permet d'imiter le comportement d'un système en traitant des modèles généralement plus proches de la réalité que les méthodes formelles, et elle sert souvent à valider les modèles de ces systèmes. La simulation a l'avantage d'être insensible à la taille de l'espace d'état, mais l'utilisation directe de la simulation est difficile. Sa complexité temporelle croît avec les détails investis dans le modèle et le degré de précision (ou l'intervalle de confiance) requis. Pour faciliter la mise en place d'une simulation, il sera plus judicieux de construire un modèle formel pour s'assurer du bon fonctionnement du modèle avant sa mise en oeuvre dans le simulateur.

Les méthodes numériques servent de modèles abstraits à partir desquels des résultats exacts peuvent être obtenus, tout en résolvant un ensemble d'équations linéaires dérivées du modèle. Ces méthodes sont beaucoup plus précises et dans certains cas beaucoup plus rapides que la simulation, mais moins rapide que les méthodes analytiques. Ces dernières permettent de dériver les paramètres de performances en fonction des paramètres temporels du modèle, tout en utilisant des solutions sous forme de formules mathématiques prédéfinies. Ces méthodes sont très efficaces et beaucoup plus rapides que la simulation, mais la solution n'existe que pour une classe très restreinte des systèmes.

Dans cette partie, on s'intéresse aux performances d'une classe de SED modélisables par les GETG. Pour cette catégorie de RdP, le problème d'évaluation de certains paramètres de performances, comme le temps de cycle, est toujours ouvert. Ce problème trouve son origine dans la présence des poids sur des arcs qui engendrent une non linéarité dans l'évolution du comportement dynamique de ces systèmes. Cette non linéarité empêche l'utilisation des formules mathématiques existantes (3.5).

L'étude bibliographique effectuée dans ce domaine, a montré l'absence de méthode analytique générale permettant d'évaluer ces performances directement à partir des GETG. Les méthodes existantes s'appliquent, soit en linéarisant le modèle mathématique décrivant la dynamique de ces graphes avec une expansion de leurs tailles, soit avec des conditions restrictives sur le marquage initial.

1. Méthode proposée par Joseph Sifakis

Dans le cas où un RdP Temporisé Fonctionnant à vitesse propre, c'est-à-dire qu'une marque ne reste dans une place que pendant sa durée d'indisponibilité, J. Sifakis propose dans [Sifakis 1980] une relation liant les temporisations, les fréquences de franchissement, et le marquage initial permet d'en déduire les performances du système. Cette équation s'écrit de la façon suivante :

$$X^T.D.W^+.F = X^T.M_0, \quad (3.1)$$

où X est un P-invariant, D une matrice diagonale telle que $D_{ii} = d_{ii}$ est la

temporisation associée à la place p_i , C^+ est la matrice d'incidence arrière, F est le vecteur des fréquences de franchissement, et M_0 le marquage initial.

Dans le cas où le RdP fonctionne à vitesse non propre, cette relation devient une inéquation, car les fréquences réelles peuvent être inférieure à celle qui correspondraient à un fonctionnement en vitesse propre.

Exemple 3.17 Considérons le GETG de la figure 3.1 admet un T-invariant $\Theta^t = (3, 2, 1)$ et un P-invariant $\mathcal{K}^t = (1, 1, 1, 1)$.

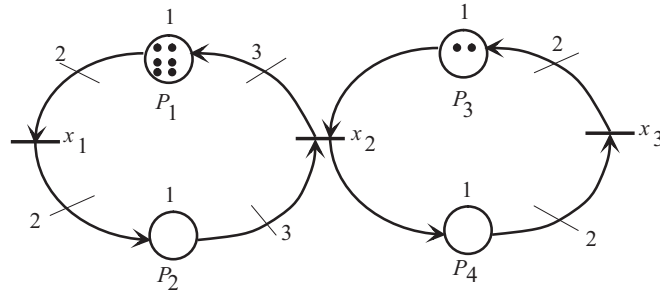


FIGURE 3.1 – Graphe d'Événements Temporisé Généralisé

- Dans le cas de fonctionnement à vitesse propre, pour un marquage initial $M_0 = (6, 0, 2, 0)^t$, Il existe deux invariants linéaires de place qui sont : $2f_1 + 3f_2 = 6$ et $f_2 + 2f_3 = 2$. On en déduit les fréquences de franchissement associées aux différentes transitions du graphe, $F = (f_1, f_2, f_3)^t = (\frac{3}{2}, 1, \frac{1}{2})^t$, ainsi que son temps de cycle $TC_m = 2$ unités de temps (u.t).
- Par contre, dans le cas où le marquage initial $M_0 = (4, 0, 2, 0)^t$ correspondant au fonctionnement à vitesse non propre, on obtient les deux inéquations suivantes : $2f_1 + 3f_2 \leq 4$ et $f_2 + 2f_3 \leq 2$. On en déduit les fréquences de franchissement qui deviennent $f_1 \leq 1$, $f_2 \leq \frac{2}{3}$ et $f_3 \leq \frac{1}{3}$, ainsi que le temps de cycle devient $TC \geq 3$ u.t.

2. Méthode proposée par D.T. Chao

D.T. Chao [Chao 1993] propose une méthode analytique pour calculer le temps de cycle des GETG avec des conditions restrictives sur le marquage initial. Cette méthode s'applique lorsque chaque circuit élémentaire

du graphe a une unique place p marquée d'un marquage valué $\mathcal{N}_p \geq 1$ (cf. équation (2.9)). Ce temps de cycle se calcule de la façon suivante :

$$TC_m = \max_{\varrho \text{ circuit élémentaire}} \frac{\mathcal{T}(\varrho)}{\mathcal{N}(\varrho)},$$

où $\mathcal{N}(\varrho)$ est le marquage valué du circuit ϱ défini par la relation (2.10). $\mathcal{T}(\varrho)$ correspond à la somme des temporisations dans le circuit ϱ .

Exemple 3.18 considérons le circuit $\rho_1 = (p_1, x_1, p_2, x_2, p_5, x_3)$ du GETG de la figure 3.2. Son T-invariant est donné par $\Theta = (2, 3, 3, 1)$. En utilisant la relation 2.10, la marque valuée de la place p_1 à l'instant initial est donnée par :

$$N_1 = \frac{6}{3 \cdot 2} = 1$$

La marque valuée du circuit ρ_1 est donnée par :

$$N(\rho) = \sum_{i=1,2,3} N_i = N_1 + N_2 + N_3 = 1 + 0 + 0 = 1$$

La marque valuée de la place p_4 à l'instant initial est donnée par :

$$N_4 = \frac{6}{1 \cdot 3} = 2$$

La marque valuée du circuit $\rho_2 = (p_3, x_4, p_4, x_2, p_5, x_3)$ est , quand à elle, donnée par :

$$N(\rho) = \sum_{i=3,4,5} N_i = N_3 + N_4 + N_5 = 0 + 2 + 0 = 2$$

Par conséquent, $TC_m = \max(\frac{4}{1}, \frac{5}{2}) = 4$ u.t.

Cette méthode permet le calcul du temps de cycle d'un GETG dans le cas multi-serveur (cf. définition 2.17). Dans le cas mono-serveur, les boucles de ré-entrance implicites contiennent un seul jeton et doivent être considérées comme des circuits. La condition $\mathcal{N}_p \geq 1$ est très rarement vérifiée pour ces places implicites. Par conséquent, cette méthode est peu applicable dans le cas des GETG mono-serveurs.

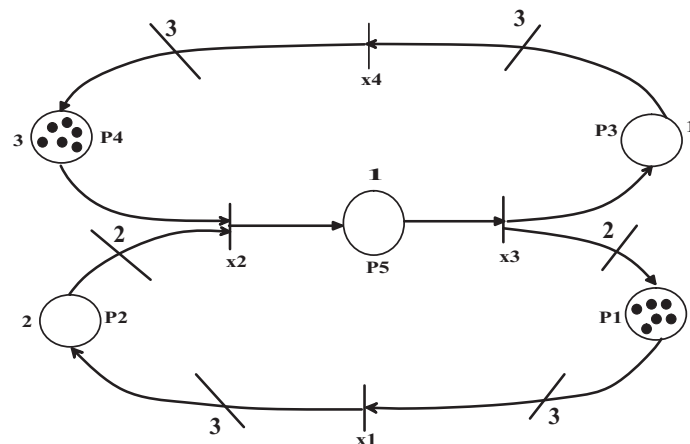


FIGURE 3.2 – Graphe d'Événements Temporisé Généralisé.

3. Méthode proposée par Alix Munier

Cette technique est une extension des travaux de P. Chrétienne [Chrétienne 1983] et C. Ramchandani [Ramchandani 1974], elle a pour objectif le calcul du temps de cycle moyen d'un graphe d'événements avec des poids sur les arcs. Dans le cas mono-serveur, A. Munier montre que le calcul du temps de cycle d'un graphe avec des poids sur les arcs peut se ramener à celui d'un graphe d'événements ordinaire. Pour cela, une méthode de linéarisation a été proposée. Cette méthode nécessite l'expansion du réseau initial. Cette expansion se fait en deux étapes : la première est la décomposition du graphe initial en un ensemble de composantes fortement connexes ; la seconde est la duplication des noeuds du graphe (places et transitions). Le lecteur intéressé par cette méthode peut trouver plus de détails dans [Munier 1993].

Dans le cas où les modèles sont multi-serveurs, une extension de cette méthode a été proposée par Nakamura et Silva [Nakamura 1999].

4. Approche de fluidification proposée par Guy Cohen

Cette méthode prend en compte le problème de pondération sur les arcs en transformant le modèle en un GET Continu. Elle utilise un dioïde d'opérateurs pour la modélisation du GET Continu avec Multiplieurs. Son principe consiste à linéariser le modèle mathématique qui reflète

l'évolution dynamique du modèle. Le modèle obtenu est $(\min, +)$ linéaire et permet d'utiliser certains éléments de la théorie spectrale pour l'analyse de performances des GET ordinaires. Cette méthode présente des avantages : elle préserve la structure et la taille du graphe initial. Il a été montré que sous certaines conditions sur le marquage initial, le comportement d'un GETG discret coïncide avec sa version continue. Ainsi, il est possible de linéariser un GETG en considérant sa version continue. Pour une présentation détaillée de cette méthode, nous renvoyons le lecteur à [Cohen 1998b].

5. Méthode proposée par Benoît Trouillet [Trouillet 2003]

A l'aide d'une algèbre spécifique, appelée (\min, \div) , une méthode est proposée afin de transformer un GETG en un GET ordinaire, indépendamment du marquage initial. Cette méthode considère que les GETG sont fortement connexes, et que chaque circuit du graphe admet au moins une composante du T-invariant égale à 1.

Une extension de cette méthode dans l'algèbre $(\min, +)$, a été proposé dans [Hamaci 2004], dans le cas où tout le graphe admet au moins une composante du T-invariant égale à 1.

6. Méthode proposée par S. Hamaci [Hamaci 2005]

Cette méthode est inspirée des travaux proposés par Guy Cohen dans [Cohen 1998b]. Elle a pour principe de linéariser les graphes d'événements temporisés avec multiplieurs sous réserve de vérifier une condition de linéarisation sur le marquage initial. Dans le cas où cette condition est satisfaite, les performances du GETG peuvent être déterminées d'une façon exacte. Dans le cas où cette condition n'est pas vérifiée, on procède à un ajout ou à un retrait de jetons dans le graphe, afin de satisfaire la condition de linéarisation pour toutes les places de ce dernier. Cette technique d'analyse permet d'encadrer les performances du GET avec multiplieurs entre deux bornes : une valeur supérieure obtenue par l'ajout d'un nombre minimal de jetons dans le graphe, et une valeur inférieure obtenue par le retrait d'un nombre minimal de jetons.

7. D'autres méthodes existent pour évaluer les performances des GETG.

Des bornes inférieures et supérieures du temps de cycle moyen dans les cas déterministe et stochastique ont été proposées par [Compos 1991] [Balbo 1998, §17] en utilisant des techniques de programmation linéaire.

Sont proposées également des méthodes de simulation, en faisant évoluer le graphe, pour déterminer le temps de cycle moyen d'un GETG [Sauer 2004].

Il résulte de cette étude bibliographique qu'une possibilité pour évaluer analytiquement les performances d'un GETG, sans hypothèse restrictive sur le marquage initial. Cette possibilité consiste à construire un GET ordinaire équivalent. Par contre, les méthodes décrites précédemment ont pour conséquence d'étendre le nombre de transitions du graphe. Une telle expansion peut alors rendre l'étude du GET ordinaire, obtenu après linéarisation, laborieuse dans le cas où la taille du GETG est déjà importante.

Dans ce contexte, pour aborder le problème d'évaluation de performances des GETG, nous proposons une méthode de linéarisation du modèle mathématique régissant l'évolution dynamique de ces graphes dans le but d'obtenir une représentations linéaires dans l'algèbre $(\min, +)$. Le modèle d'état obtenu permet d'appliquer certains résultats développés dans le cadre de l'analyse de performances des systèmes linéaires dans les dioïdes. Cette méthode est plus générale par rapport aux méthodes existantes, elle s'applique sans conditions restrictives et sans expansion du réseau.

Ce chapitre est organisé comme suit : après un rappel sur les éléments de la théorie des graphes, une application sur l'évaluation de performances des GET est présentée. Afin d'appliquer certains résultats de base de la théorie spectrale pour l'évaluation de performances des GETG, une nouvelles méthodes de linéarisation sera présentée. Cette méthode a pour but de linéariser le modèle mathématique associé au comportement d'un GETG n'admettant pas une représentation linéaire dans le dioïde $(\min, +)$, ceci afin d'obtenir un GET ordinaire équivalent.

3.2 Notions élémentaires

Cette section a pour vocation à rappeler brièvement un ensemble de définitions de la théorie des graphes utiles pour la compréhension des résultats présentés dans

de ce chapitre.

Définition 3.24 (Graphe orienté). Un *graphe orienté* qu'on note $\mathcal{G} = (\Omega, \Phi)$, est un graphe composé d'un ensemble de sommets (ou noeuds), noté Ω , et un ensemble d'arcs orientés, noté Φ (cf. figure 3.3).

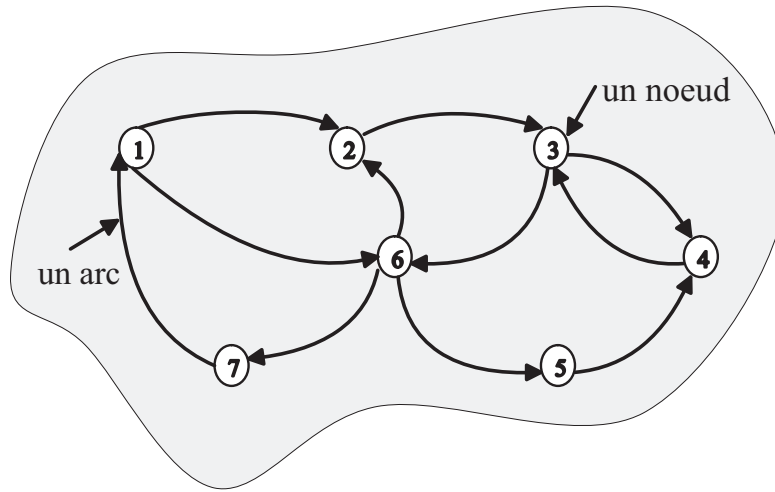


FIGURE 3.3 – Graphe orienté.

Définition 3.25 (Chemin, circuit). Un *chemin* est une suite de sommets que l'on peut parcourir successivement en utilisant les arcs de Φ . On note $\xrightarrow{a \ b}$ un chemin allant d'un sommet d'origine, noté a , vers un sommet d'extrémité, noté b .

Un chemin est dit :

- **simple** s'il ne passe pas deux fois par le même arc.
- **élémentaire** s'il ne passe pas deux fois par le même sommet.

Un *circuit* est un chemin dont le sommet de départ est confondu avec sommet d'arrivée (cf. figure 3.4).

Définition 3.26 (Composante fortement connexe). Un graphe orienté est *fortement connexe* si pour tout couple de sommets a et b , il existe un chemin allant de a à b (cf. figure 3.5).

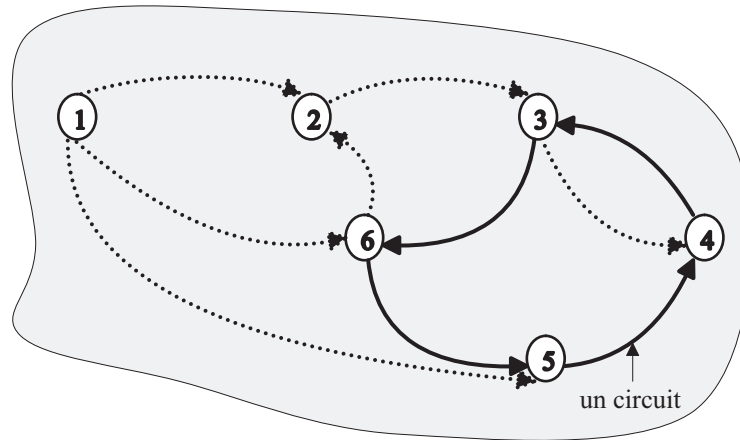


FIGURE 3.4 – Circuit dans un graphe orienté.

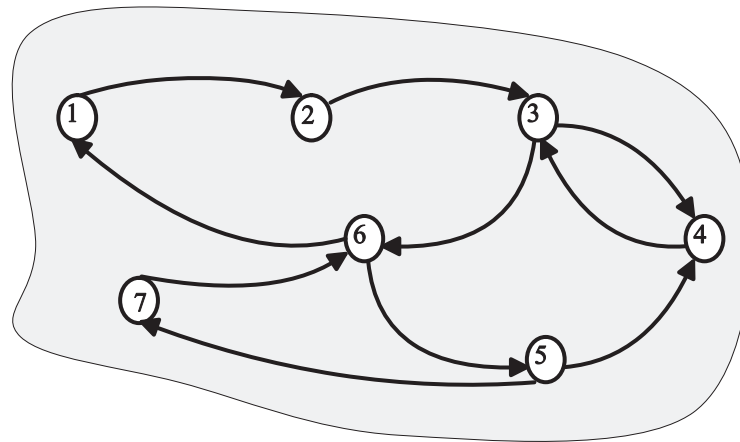


FIGURE 3.5 – Composante fortement connexe.

Définition 3.27 (Graphe valué). Un *graphe valué* est un graphe pour lequel tout arc $(a, b) \in \Phi$ est muni d'un poids μ_{ba} (l'indice du sommet d'arrivée b précède l'indice du sommet de départ a).

Soit $\Delta = (a_1, a_2, \dots, a_k)$ un chemin d'un graphe valué. On appelle :

- longueur de Δ , notée $|\Delta|_l$, le nombre d'arcs qui le composent ;

- poids de Δ , noté $|\Delta|_\mu$, la somme des poids des arcs qui le composent, qui est égal à $a_{i_k i_{k-1}} + \dots + a_{i_2 i_1}$.

Définition 3.28 (Matrice et graphe de précédence). Soit A une matrice de dimension $n \times n$ à coefficients dans un dioïde \mathcal{D} . La matrice A d'un graphe valué à n sommets, ayant pour élément A_{ab} le poids μ_{ab} de l'arc $\overset{b}{\underset{a}{\rightarrow}}$, est appelée *matrice de précédence*.

De tout graphe de précédence, on peut extraire une matrice de précédence.

La graphe représenté par la figure 3.6 admet la matrice de précédence suivante :

$$\begin{pmatrix} \varepsilon & \varepsilon & \varepsilon & 1 & 5 & \varepsilon \\ \varepsilon & 4 & \varepsilon & 3 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & 4 & 3 & \varepsilon \\ \varepsilon & 6 & \varepsilon & \varepsilon & \varepsilon & 3 \\ \varepsilon & 1 & \varepsilon & \varepsilon & \varepsilon & 5 \\ 5 & \varepsilon & 2 & \varepsilon & \varepsilon & \varepsilon \end{pmatrix}$$

ε correspond à un arc absent dans le graphe.

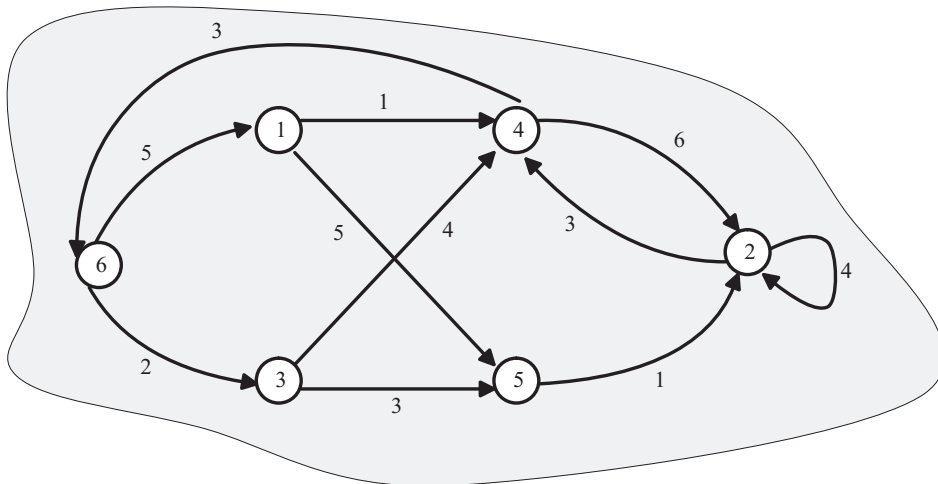


FIGURE 3.6 – Graphe orienté valué.

Pour toute matrice carrée A de dimension $n \times n$ à élément dans \mathbb{R}_{min} , l'éléments $(A^\eta)_{ab}$ correspond au *poids minimal* des chemins de longueur η allant du noeud b au noeud a dans son graphe de précédence. Par exemple, pour $\eta = 2$, on aura :

$$(A^2)_{ab} = \bigoplus_{l=1}^n A_{al} \otimes A_{lb},$$

Dans l'algèbre usuelle, ça donne :

$$(A^2)_{ab} = \min_{l=1, \dots, n} \{A_{al} + A_{lb}\}.$$

Définition 3.29 (Matrice irréductible). Une matrice *irréductible* $A \in \mathcal{D}^{n \times n}$ est une matrice pour laquelle, pour toute paire (a, b) , il existe un entier i tel que $(A^i)_{ab} \neq \varepsilon$.

Ceci est équivalent à l'existence d'un chemin allant de b à a pour toute paire de sommets (a, b) .

3.3 Eléments de la théorie spectrale

Dans cette partie, on rappelle les principales propriétés spectrales des matrices à coefficients dans le dioïde $\mathbb{R}_{min}^{n \times n}$, dont le théorème de cyclicité des matrices irréductibles, et comment ces résultats peuvent être appliqués à l'évaluation de performances des GET.

Définition 3.30 (Valeurs propres et vecteurs propres). On appelle valeur propre et vecteur propre d'une matrice $A \in \mathcal{D}^n$, le scalaire $\lambda \in \mathbb{R}$ et le vecteur $v \in \mathbb{R}_{min}^n \setminus \{\varepsilon\}$ tels que :

$$A \otimes v = \lambda \otimes v. \quad (3.2)$$

Pour calculer cette valeur propre, on utilise le théorème suivant.

Théorème 3.2 [Gondran 1985] Une matrice $A \in \mathbb{R}_{min}^{n \times n}$ irréductible admet une unique valeur propre, notée λ , égale au rayon spectral de la matrice A , soit :

$$\lambda = \rho(A) = \bigoplus_{k=1}^n (tr A^k)^{\otimes 1/k},$$

où $tr A^k = \bigoplus_{i=1}^n (A^k)_{ii}$ correspond au poids minimal¹ de tous les circuits de longueur k du graphe $\mathcal{G}(A)$. En définissant le *poids moyen* d'un circuit ζ comme la valeur de son poids, noté $|\zeta|_w$, divisée par sa longueur, noté $|\zeta|_l$; le poids moyen minimal des circuits de longueur k s'obtient alors en divisant $tr A^k$ par k , ce qui est équivalent à l'expression $(tr A^k)^{\otimes 1/k}$ dans le dioïde $\mathbb{R}_{min}^{n \times n}$.

Cette valeur propre peut aussi être déduite directement à partir du graphe $\mathcal{G}(A)$ à travers le résultat donné dans le théorème suivant.

Théorème 3.3 [Gondran 1985] La valeur propre λ d'une matrice $A \in \mathbb{R}_{min}^{n \times n}$ irréductible est égale au poids moyen minimal des circuits de $\mathcal{G}(A)$, c'est-à-dire :

$$\lambda = \min_{\zeta \in \mathcal{C}} \frac{|\zeta|_w}{|\zeta|_l},$$

où \mathcal{C} est l'ensemble des circuits de $\mathcal{G}(A)$.

Une matrice irréductible peut admettre plusieurs vecteurs propres associés à la valeur propre λ . Afin de les calculer, on définit, au préalable, la matrice A_λ .

Définition 3.31 Soit $A \in \mathbb{R}_{min}^{n \times n}$ une matrice irréductible de valeur propre λ . On définit la matrice, notée A_λ , par :

$$A_\lambda = \lambda^{-1} \otimes A.$$

Le théorème suivant fournit une condition permettant de tester si une colonne de la matrice A_λ^+ est un vecteur propre de A .

Théorème 3.4 [Gondran 1985] Soit $A \in \mathcal{D}^{n \times n}$ une matrice irréductible de valeur propre λ . La j -ième colonne de la matrice A_λ^+ , notée $(A_\lambda^+)_j$, est un vecteur propre de A , si elle satisfait l'égalité :

$$(A_\lambda^+)_j = A_\lambda \otimes (A_\lambda^+)_j.$$

Remarque 3.6 Dans le cas où A est irréductible, toutes les coordonnées d'un vecteur propre sont différentes de ε .

1. $tr A^k = \bigoplus_{i=1}^n (A^k)_{ii}$ correspond au poids maximal de tous les circuits de longueur k du graphe $\mathcal{G}(A)$ dans le cas du dioïde $\mathbb{R}_{max}^{n \times n}$.

Définition 3.32 (Circuit critique, graphe critique). Soient $A \in \mathbb{R}_{min}^{n \times n}$ une matrice irréductible de valeur propre λ , et $\mathcal{G}(A)$ le graphe de précédence associé.

Un circuit ζ est dit *critique* si son poids moyen est minimum, c'est-à-dire si $\frac{|\zeta|_w}{|\zeta|_l} = \lambda$.

On appelle *graphe critique*, noté $\mathcal{G}^c(A)$, le sous graphe de $\mathcal{G}(A)$ formé des sommets et des arcs appartenant à un circuit critique.

Définition 3.33 (Cyclicité). On appelle :

- *cyclicité d'une composante fortement connexe* le plus grand commun diviseur (pgcd) des longueurs des circuits qui la composent ;
- *cyclicité d'un graphe* le plus petit commun multiple (ppcm) des cyclicités de ses composantes fortement connexes.

Exemple 3.19 Le graphe de précédence représenté par la figure 3.7 admet la matrice irréductible suivante dans le dioïde $\mathbb{R}_{min}^{3 \times 3}$:

$$A = \begin{pmatrix} \varepsilon & \varepsilon & 1 \\ 0 & 2 & 0 \\ \varepsilon & 2 & \varepsilon \end{pmatrix}.$$

Dans cet exemple, le graphe est composé de trois circuits élémentaires :

- le premier circuit : $noeud_1 \rightarrow noeud_2 \rightarrow noeud_3 \rightarrow noeud_1$ a un poids moyen égal à 1,
- le deuxième circuit : $noeud_2 \rightarrow noeud_3 \rightarrow noeud_2$ a un poids moyen égal à 1,
- le troisième circuit : $noeud_2 \rightarrow noeud_2$ a un poids moyen égal à 2.

Aussi, on a

$\lambda = \min(1, 1, 2) = 1$, ce qui correspond au poids moyen minimal du graphe.

Comme on a vu précédemment, cette valeur propre peut également être calculée en utilisant le théorème 3.2. On a :

$$A^2 = \begin{pmatrix} \varepsilon & 3 & \varepsilon \\ 2 & 2 & 1 \\ 2 & 4 & 2 \end{pmatrix}, \quad A^3 = \begin{pmatrix} 3 & 5 & 3 \\ 2 & 3 & 2 \\ 4 & 4 & 3 \end{pmatrix}.$$

D'où :

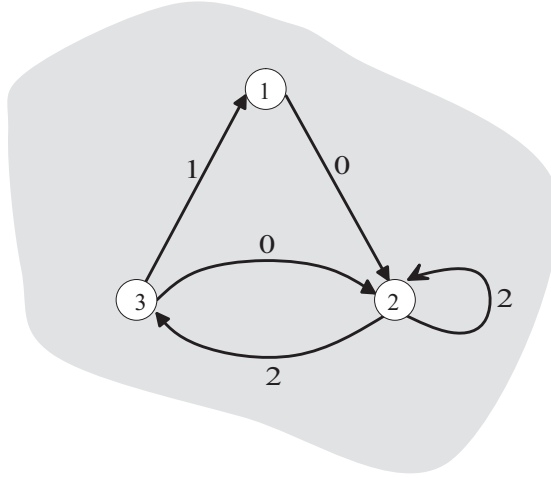


FIGURE 3.7 – Exemple d'un graphe de précedence fortement connexe.

- $tr(A) = \bigoplus_{i=1}^3 A_{ii} = 2,$
- $tr(A^2) = \bigoplus_{i=1}^3 A_{ii}^2 = 2,$
- $tr(A^3) = \bigoplus_{i=1}^3 A_{ii}^3 = 3,$

ainsi, on a :

$$\lambda = \bigoplus_{k=1}^3 (tr A^k)^{\otimes 1/k} = tr(A) \oplus tr(A^2)^{\otimes 1/2} \oplus tr(A^3)^{\otimes 1/3} = 1.$$

• Calcul des vecteurs propres associés à la matrice A

On a, d'après le définition 3.31 :

$$A_\lambda = \lambda^{-1} \otimes A = \begin{pmatrix} \varepsilon & \varepsilon & 0 \\ -1 & 1 & -1 \\ \varepsilon & 1 & \varepsilon \end{pmatrix} \quad (\lambda^{-1} \text{ correspond à } -\lambda \text{ dans l'al-}$$

gèbre usuelle),

$$\text{d'où } A_\lambda^+ = A_\lambda \oplus A_\lambda^2 \oplus \dots = \begin{pmatrix} 0 & 1 & 0 \\ -1 & 0 & -1 \\ 0 & 1 & 0 \end{pmatrix}.$$

En appliquant le théorème 3.4, on vérifie que les trois colonnes de la matrice A_λ^+ correspondent bien aux vecteurs propres associés à la valeur propre λ ($=1$) au sens où ils vérifient la relation (3.2).

3.4 Evaluation de performances des GET

Considérons un graphe d'événements temporisé fortement connexe. Son évolution dynamique se modélise par l'équation récurrente $C(t) = A \otimes C(t-1)$ dans l'algèbre $(\min, +)$. La valeur propre λ associée à la matrice d'état irréductible A , à coefficients dans le dioïde $\mathbb{R}_{min}^{n \times n}$, correspond à la *fréquence de franchissement* asymptotique du GET, défini comme le nombre moyen de franchissement, associée à chaque transition, par unité de temps. Cette fréquence de franchissement, qui représente aussi le taux de production dans le cas d'un GET modélisant un système de production, est identique pour toutes les transitions du graphe. Pour un système de production, il correspond aux nombres moyens de pièces produites par unité de temps.

Définition 3.34 (Taux de production asymptotique). [Gaubert 1995]

Soit $c(t)$ le compteur associé à la transition c . Son taux de production asymptotique, noté λ , est défini par :

$$\lambda = \lim_{t \rightarrow +\infty} \frac{c(t)}{t}.$$

Ce taux de production peut être calculé de deux façons : soit en appliquant une approche algébrique à travers le théorème 3.2, soit en utilisant le théorème suivant qui permet de calculer λ à partir du graphe de précédence.

Théorème 3.5 [Gaubert 1992] Le taux de production λ d'un GET fortement connexe est caractérisé par :

$$\lambda = \min_{\rho \text{ circuit élémentaire}} \frac{M(\rho)}{Temp(\rho)}, \quad (3.3)$$

où $M(\rho)$ correspond au nombre total de jetons du circuit ρ , équivalent au poids du circuit du graphe de précédence associé, et $Temp(\rho)$ correspond à la somme des temporisations des places du circuit ρ , équivalent à la longueur du circuit du graphe de précédence associé.

L'inverse de cette valeur propre est également utilisée pour analyser les performances d'un GET. Elle correspond au *temps de cycle moyen* du graphe, c'est-à-dire le temps moyen séparant deux tirs consécutifs d'une transition. Pour un système de production, il correspond au temps moyen séparant deux pièces produites. Ce temps de cycle, noté TC , peut être calculé de la façon suivante :

$$TC = \max_{\rho \text{ circuit élémentaire}} \frac{Temp(\rho)}{M(\rho)}. \quad (3.4)$$

Exemple 3.20 Les performances du GET de la figure 3.8 sont identiques à celles du graphe de précedence de la figure 3.7 calculées précédemment.

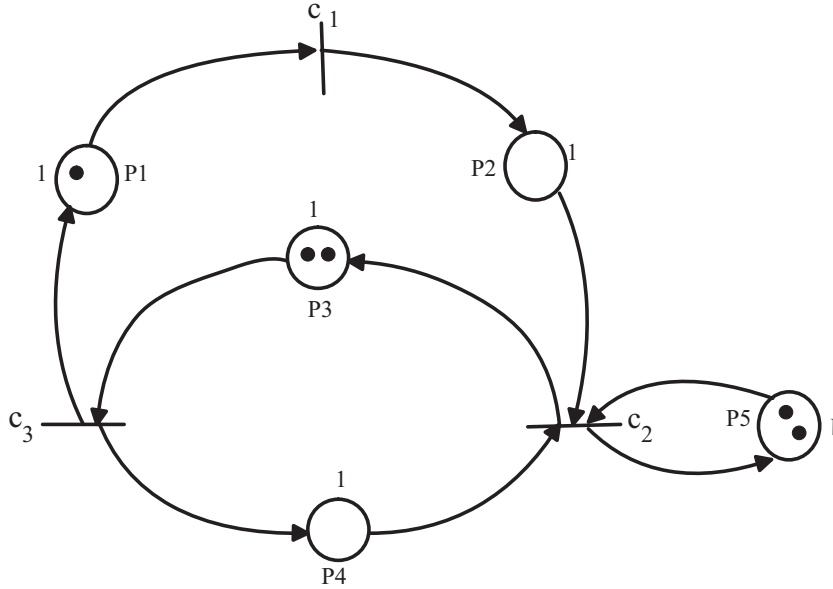


FIGURE 3.8 – GET équivalent au graphe de précedence de la figure 3.7.

Si λ est la valeur propre de A et si le vecteur d'état initial $C(0)$ est un vecteur propre de A , on a :

$$\begin{aligned} C(t) &= AC(t-1) \\ &= A^t C(0) \\ &= A^{t-1} \otimes A \otimes C(0) = A^{t-1} \otimes \lambda \otimes C(0) \\ &= \lambda^t \otimes C(0) = \lambda C(t-1), \end{aligned}$$

ce qui est équivalent dans l'algèbre classique à :

$$C(t) = \lambda + C(t - 1),$$

autrement dit, l'état du système est incrémenté de λ lorsque t est incrémenté de 1, ce qui signifie que, dès l'état initial, le GET fonctionne, en régime autonome, de façon périodique selon une période égale à 1.

L'établissement d'une valeur propre de A permet d'évaluer le comportement périodique d'un GET en régime autonome, lorsque la condition $C(0)$ correspond à un vecteur propre de A . Le théorème suivant fournit une réponse sur le comportement du graphe pour une condition initiale quelconque.

Théorème 3.6 [Baccelli 1992] Soit $A \in \mathbb{R}_{min}^{n \times n}$ une matrice irréductible. Il existe deux entiers N et c tels que :

$$n \geq N \Rightarrow A^{n+c} = \lambda^c A^n,$$

où c est appelé cyclicité de A .

La matrice A étant irréductible, une fois le régime périodique atteint, on a :

$$\begin{aligned} C(t) &= A^{n+c} C(t - n - c) \\ &= \lambda^c \otimes C(t - c), \end{aligned}$$

autrement dit, en régime autonome, pour t suffisamment grand, il existe $c \in \mathbb{N}$ tel que :

$$C(t) = c \times \lambda + C(t - c).$$

En résumé, on peut dire qu'un GET en régime autonome atteint, après un régime transitoire fini, un régime périodique de période égale à c .

Remarque 3.7 (Matrice réductible). Les propriétés spectrales et asymptotiques des matrices à éléments dans un dioïde ont également été étudiées dans le cas *réductible* (non irréductible) [Gaubert 1992, §4]. Pour une matrice $A \in \mathbb{R}_{min}^{n \times n}$ réductible, l'unicité de la valeur propre n'est pas assurée. Les résultats énoncés dans ce cas s'appuient sur une décomposition du graphe de précedence de A en composantes fortement connexes.

3.5 Evaluation de performances des GETG

Comme dans le cas des GET, les performances d'un GETG fortement connexe, sont obtenues dans le cas d'un fonctionnement au plus tôt et lorsque le régime permanent est établi [Munier 1993]. Par contre, les fréquences de franchissement ne sont pas identiques pour toutes les transitions, mais sont proportionnelles au T-invariant élémentaire (cf. définition 2.14). Pour chaque transition, la fréquence de franchissement, notée λ_{m_q} , peut être définie de la façon suivante :

$$\lambda_{m_q} = \frac{\theta_q}{TC_m}, \quad (3.5)$$

où :

- θ_q est la composante du T-invariant associée à la transition x_q ,
- TC_m est le temps de cycle du graphe.

Dans le cas d'un graphe neutre et fortement connexe, ce temps de cycle peut être défini comme suit.

Théorème 3.7 [Sauer 2003] Le *temps de cycle*, noté TC_m , d'un GETG neutre et fortement connexe, obtenu à partir d'un marquage initial M_0 , est le temps moyen nécessaire pour franchir une fois le T-invariant minimal en mode de fonctionnement au plus tôt à partir de M_0 .

Si le calcul de performances des GET a pu être réalisé *via* la théorie des graphes, le problème reste, à notre connaissance, ouvert pour les GETG. La difficulté de ces types de graphes est due à la présence des poids sur les arcs qui empêche l'utilisation de la relation (3.5). Comme alternative, nous proposons une méthode de linéarisation du modèle mathématique régissant l'évolution dynamique du graphe, dans le but d'obtenir un modèle linéaire dans l'algèbre $(\min, +)$ [Benfekir 2013a, Benfekir 2013b].

3.5.1 Méthode de linéarisation proposée

Cette méthode s'applique sur des GETG consistants (cf. définition 2.15) et conservatifs (cf. définition 2.16).

Pour l'étude des ces GETG, les conditions initiales considérées ici sont faiblement compatibles (2.20).

Pour obtenir ces conditions initiales, chaque compteur du GETG est synchronisé avec un compteur modélisant l'évolution dynamique d'une entrée impulsionnelle $e(t)$.

$$\begin{cases} e(t) = 0, & t < 0; \\ e(t) \rightarrow +\infty, & t = 0. \end{cases}$$

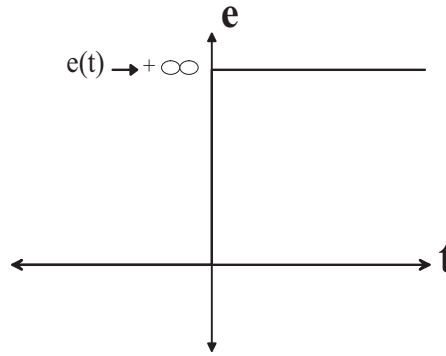


FIGURE 3.9 – Impulsion d'un point de vue compteur.

Cette entrée impulsionnelle $e(t)$, qui est un signal, se code dans le dioïde $\mathcal{O}_{\min}[\![\delta]\!]$ de la façon suivante :

$$E(\delta) = \gamma^0 \delta^0 \oplus \gamma^1 \delta^0 \oplus \gamma^2 \delta^0 \oplus \gamma^3 \delta^0 \oplus \gamma^4 \delta^0 \oplus \dots \oplus \gamma^{+\infty} \delta^0$$

Chaque compteur d'un GETG peut être codé par la composition des opérateurs : μ, γ, δ (cf. équation (2.12)).

En plus des règles de commutation entre ces opérateurs, cités précédemment dans le chapitre 2, la proposition suivante fournit d'autres règles utiles pour la méthode de linéarisation que nous proposons.

Proposition 1 Soient $a, b \in \mathbb{N}^*$, nous avons :

1. $\mu_{a^{-1}} \mu_b = \mu_{(a^{-1}b)}$.
2. $\forall t \in \mathbb{Z}$, $x(t)$ est un multiple de a , alors $\mu_{a^{-1}} \gamma^b X(\delta) = \gamma^{\lfloor a^{-1}b \rfloor} \mu_{a^{-1}} N(\delta)$.
3. $\gamma^b \mu_a = \mu_a \gamma^{a^{-1}b}$ est équivalent à $\mu_a \gamma^b = \gamma^{ab} \mu_a$.

Preuve 2

1. $\mu_{a^{-1}}\mu_b X(\delta)$ correspond à $\lfloor a^{-1} \lfloor b x(t) \rfloor \rfloor = \lfloor a^{-1} b x(t) \rfloor$, ce qui équivaut à $\mu_{(a^{-1}b)} X(\delta)$.
2. $\mu_{a^{-1}}\gamma^b X(\delta)$ correspond à $\lfloor a^{-1}(b + x(t)) \rfloor = \lfloor a^{-1}b \rfloor + a^{-1}x(t)$ sachant que $x(t) \in \mathbb{Z} \cup \{\pm\infty\}$ est un multiple de a , ce qui est équivalent à $\gamma^{\lfloor a^{-1}b \rfloor} \mu_{a^{-1}} X(\delta)$.
3. $\gamma^b \mu_a X(\delta)$ correspond à $b + \lfloor a x(t) \rfloor = \lfloor b + a x(t) \rfloor = \lfloor a(a^{-1}b + x(t)) \rfloor$, ce qui est équivalent à $\mu_a \gamma^{a^{-1}b} X(\delta)$.

Un GETG consistant possède un T-invariant dans lequel toutes ses composantes appartiennent à \mathbb{N}^* . Le principe de linéarisation consiste à exprimer chaque compteur en fonction d'une entrée impulsionnelle. Cette dernière, de par son amplitude qui tend vers l'infini, permet de linéariser toute équation récurrente non linéaire modélisant la dynamique de franchissement d'une transition.

Proposition 2 Soit E une entrée impulsionnelle, on a : $\forall a \in \mathbb{N}, \beta \in \mathbb{Q}^+, \forall q = 1, \dots, n$,

$$\mu_\beta \gamma^a \delta^\tau E_q(\delta) = \gamma^{\lfloor \beta a \rfloor} \delta^\tau E_q(\delta).$$

Preuve 3 Utilisons le point 3 de la proposition 1.

$\mu_\beta \gamma^a \delta^\tau E_q(\delta)$ correspond à $\lfloor \beta \times (a + e_q(t - \tau)) \rfloor = \lfloor \beta \times a \rfloor + e_q(t - \tau)$ puisque pour $t \geq 0, e_q(t) \mapsto +\infty$, donc $e_q(t)$ est un multiple de a , qui est équivalent à $\gamma^{\lfloor \beta a \rfloor} \delta^\tau E_q(\delta)$.

Dans la suite, nous donnons le modèle d'état modélisant la dynamique du GETG avec les entrées impulsionnelles.

Considérons le vecteur X composé des compteurs codés dans le dioïde d'opérateurs $\mathcal{O}_{\min}[\delta]$. Les compteurs E correspondent aux entrées impulsionnelles. Finalement, on obtient :

$$X(\delta) = A \otimes X(\delta) \bigoplus E(\delta). \quad (3.6)$$

D'après le théorème 1.1, ce type d'équations admet la plus petite solution suivante :

$$X(\delta) = A^* \otimes E(\delta), \quad (3.7)$$

avec $A^* = e \oplus A \oplus A^2 \oplus \dots$.

Proposition 3 Pour des conditions initiales faiblement compatibles, un GETG est linéarisable sans augmenter le nombre de ses transitions.

Preuve 4 Un GETG consistant et conservatif se représente par l'équation $X(\delta) = A \otimes X(\delta) \oplus E(\delta)$. En utilisant la proposition 2, on obtient une relation linéaire entre deux transitions du graphe (correspondant à un GET). Cette procédure de linéarisation peut être appliquée pour toutes les transitions du graphe, puisque pour toute transition, on peut associer une entrée impulsionnelle.

Dans la suite, nous donnons un exemple de GETG consistant et conservatif (fortement connexe) pour calculer son temps de cycle. Nous faisons remarquer qu' hormis la simulation, toutes les méthodes existantes ne permettent pas de donner la valeur exacte de ce GETG.

Exemple 3.21 L'évolution dynamique du GETG représenté par la figure 3.10 est décrite par les équations récurrentes suivantes :

$$\begin{cases} x_1(t) &= \min(3x_3(t-2)), \\ x_2(t) &= \min(\lfloor \frac{4+x_1(t-1)}{2} \rfloor, 3 + x_2(t-1), \lfloor \frac{4+3x_3(t-1)}{2} \rfloor), \\ x_3(t) &= \min(\lfloor \frac{2x_2(t-1)}{3} \rfloor). \end{cases}$$

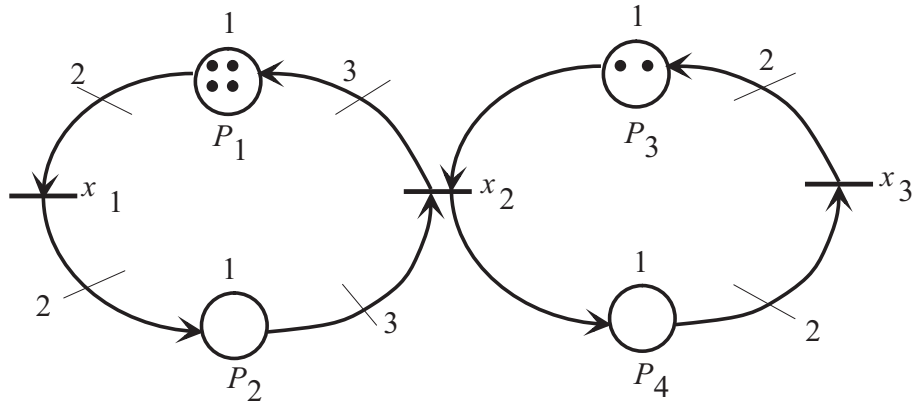


FIGURE 3.10 – Exemple de GETG.

Ces équations récurrentes sont non linéaires dans l'algèbre $(\min, +)$. Cette non linéarité - de par les poids sur les arcs - est due à la présence de parties entières dans le modèle $(\min, +)$ régissant l'évolution dynamique de ces graphes.

Pour pallier à ce problème de non linéarité et pouvoir, en suite, appliquer les résultats développés dans le cadre de l'analyse de performances des GET dans l'algèbre $(\min, +)$, on code chaque dynamique associée à l'état du GETG dans le dioïde d'opérateurs $\mathcal{O}_{\min}[\![\delta]\!]$, et de synchroniser l'évolution de chaque compteurs en fonction d'une impulsion (figure 3.11).

$$\begin{pmatrix} X_1 \\ X_2 \\ X_3 \end{pmatrix} = \begin{pmatrix} \varepsilon & \mu_{1/2}\gamma^4\delta^1\mu_3 & \varepsilon \\ \mu_{1/3}\delta^1\mu_2 & \varepsilon & \gamma^2\delta^1\mu_2 \\ \varepsilon & \mu_{1/2}\delta^1 & \varepsilon \end{pmatrix} \begin{pmatrix} X_1 \\ X_2 \\ X_3 \end{pmatrix} \oplus \begin{pmatrix} e \\ e \\ e \end{pmatrix} . E$$

Les entrées E correspondent aux entrées impulsionnelles $e(t)$ codées dans le dioïde $\mathcal{O}_{\min}[\![\delta]\!]$. Elles n'ont aucune influence sur l'évolution du graphe. En effet, $\forall x_q \in \mathcal{T}$, $\min(x_q(t), e_q(t)) = x_q(t)$, puisque $e(t) \rightarrow +\infty$.

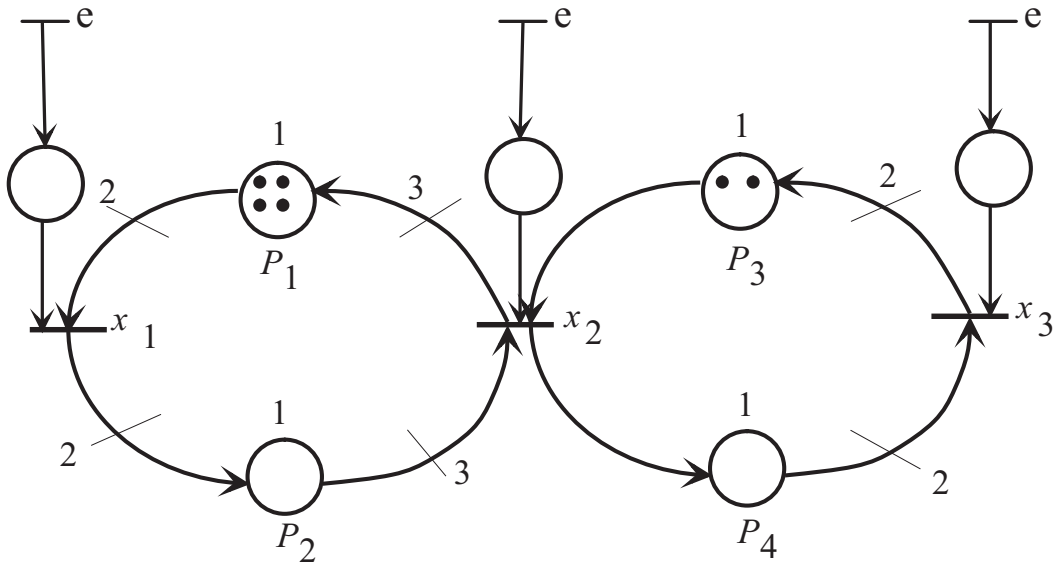


FIGURE 3.11 – GETG avec des entrées impulsionnelles associées aux transitions du graphe.

Utilisant l'équation (4.1), $X(\delta) = A^*E(\delta)$. La proposition (2) permet de calculer $A^*E(\delta)$:

$$\begin{aligned} A^*E(\delta) &= (e \oplus A \oplus A^2 \oplus A^3 \oplus \dots)E(\delta) \\ &= (E(\delta) \oplus AE(\delta) \oplus \underbrace{A \otimes AE(\delta)}_{A^2E(\delta)} \oplus \underbrace{A \otimes A^2E(\delta)}_{A^3E(\delta)} \oplus \dots). \end{aligned}$$

Dans cette partie, nous donnons les expressions des sept différents termes de $A^*E(\delta)$. Cette dernière converge vers un régime périodique après certaines itérations.

• **Etape 1**

$$AE(\delta) = \begin{pmatrix} \varepsilon & \mu_{1/2}\gamma^4\delta^1\mu_3 & \varepsilon \\ \mu_{1/3}\delta^1\mu_2 & \varepsilon & \gamma^2\delta^1\mu_2 \\ \varepsilon & \mu_{1/2}\delta^1 & \varepsilon \end{pmatrix} \begin{pmatrix} e \\ e \\ e \end{pmatrix} E(\delta) = \begin{pmatrix} \gamma^2\delta \\ \delta \\ \delta \end{pmatrix} \otimes E(\delta)$$

• **Etape 2**

$$A^2E(\delta) = AAE(\delta) = \begin{pmatrix} \varepsilon & \mu_{1/2}\gamma^4\delta^1\mu_3 & \varepsilon \\ \mu_{1/3}\delta^1\mu_2 & \varepsilon & \gamma^2\delta^1\mu_2 \\ \varepsilon & \mu_{1/2}\delta^1 & \varepsilon \end{pmatrix} \begin{pmatrix} \gamma^2\delta \\ \delta \\ \delta \end{pmatrix} E(\delta) = \begin{pmatrix} \gamma^2\delta^2 \\ \gamma^1\delta^2 \\ \delta^2 \end{pmatrix} E(\delta)$$

• **Etape 3**

$$A^3E(\delta) = AA^2E(\delta) = \begin{pmatrix} \varepsilon & \mu_{1/2}\gamma^4\delta^1\mu_3 & \varepsilon \\ \mu_{1/3}\delta^1\mu_2 & \varepsilon & \gamma^2\delta^1\mu_2 \\ \varepsilon & \mu_{1/2}\delta^1 & \varepsilon \end{pmatrix} \begin{pmatrix} \gamma^2\delta^2 \\ \gamma^1\delta^2 \\ \delta^2 \end{pmatrix} E(\delta) = \begin{pmatrix} \gamma^3\delta^3 \\ \gamma^1\delta^3 \\ \delta^3 \end{pmatrix} E(\delta)$$

• **Etape 4**

$$A^4 E(\delta) = AA^3 E(\delta) = \begin{pmatrix} \varepsilon & \mu_{1/2}\gamma^4\delta^1\mu_3 & \varepsilon \\ \mu_{1/3}\delta^1\mu_2 & \varepsilon & \gamma^2\delta^1\mu_2 \\ \varepsilon & \mu_{1/2}\delta^1 & \varepsilon \end{pmatrix} \begin{pmatrix} \gamma^3\delta^3 \\ \gamma^1\delta^3 \\ \delta^3 \end{pmatrix} E(\delta) = \begin{pmatrix} \gamma^3\delta^4 \\ \gamma^2\delta^4 \\ \delta^4 \end{pmatrix} E(\delta)$$

• **Etape 5**

$$A^5 E(\delta) = AA^4 E(\delta) = \begin{pmatrix} \varepsilon & \mu_{1/2}\gamma^4\delta^1\mu_3 & \varepsilon \\ \mu_{1/3}\delta^1\mu_2 & \varepsilon & \gamma^2\delta^1\mu_2 \\ \varepsilon & \mu_{1/2}\delta^1 & \varepsilon \end{pmatrix} \begin{pmatrix} \gamma^3\delta^4 \\ \gamma^2\delta^4 \\ \delta^4 \end{pmatrix} E(\delta) = \begin{pmatrix} \gamma^5\delta^5 \\ \gamma^2\delta^5 \\ \gamma^1\delta^5 \end{pmatrix} E(\delta)$$

• **Etape 6**

$$A^6 E(\delta) = AA^5 E(\delta) = \begin{pmatrix} \varepsilon & \mu_{1/2}\gamma^4\delta^1\mu_3 & \varepsilon \\ \mu_{1/3}\delta^1\mu_2 & \varepsilon & \gamma^2\delta^1\mu_2 \\ \varepsilon & \mu_{1/2}\delta^1 & \varepsilon \end{pmatrix} \begin{pmatrix} \gamma^5\delta^5 \\ \gamma^2\delta^5 \\ \gamma^1\delta^5 \end{pmatrix} E(\delta) = \begin{pmatrix} \gamma^5\delta^6 \\ \gamma^3\delta^6 \\ \gamma^1\delta^6 \end{pmatrix} E(\delta)$$

• **Etape 7**

$$A^7 E(\delta) = AA^6 E(\delta) = \begin{pmatrix} \varepsilon & \mu_{1/2}\gamma^4\delta^1\mu_3 & \varepsilon \\ \mu_{1/3}\delta^1\mu_2 & \varepsilon & \gamma^2\delta^1\mu_2 \\ \varepsilon & \mu_{1/2}\delta^1 & \varepsilon \end{pmatrix} \begin{pmatrix} \gamma^5\delta^6 \\ \gamma^3\delta^6 \\ \gamma^1\delta^6 \end{pmatrix} E(\delta) = \begin{pmatrix} \gamma^6\delta^7 \\ \gamma^3\delta^7 \\ \gamma^1\delta^7 \end{pmatrix} E(\delta)$$

Par conséquent, la dynamique de chaque compteur s'exprime sous forme d'une série formelle dans le dioïde $\mathcal{O}_{\min}[[\delta]]$ comme suit :

- $X_1(\delta) = \gamma^2\delta^1 \oplus \gamma^2\delta^2 \oplus \gamma^3\delta^3 \oplus \gamma^3\delta^4 \oplus \gamma^5\delta^5 \oplus \gamma^5\delta^6 \oplus \gamma^6\delta^7 \oplus \gamma^6\delta^8 \oplus \gamma^8\delta^9 \oplus \gamma^8\delta^{10} \oplus \gamma^9\delta^{11} \oplus \gamma^9\delta^{12} \oplus \gamma^{11}\delta^{13} \oplus \gamma^{11}\delta^{14} \oplus \gamma^{12}\delta^{15} \oplus \gamma^{12}\delta^{16} \oplus \gamma^{14}\delta^{17} \oplus \dots$

Après avoir appliqué les règles de simplification entre les opérateurs γ et δ , développées dans le chapitre 2, on obtient :

$$X_1(\delta) = \gamma^2\delta^2(\gamma^3\delta^4)^*$$

- $X_2(\delta) = \delta^1 \oplus \gamma^1\delta^2 \oplus \gamma^1\delta^3 \oplus \gamma^2\delta^4 \oplus \gamma^2\delta^5 \oplus \gamma^3\delta^6 \oplus \gamma^3\delta^7 \oplus \gamma^4\delta^8 \oplus \gamma^4\delta^9 \oplus \gamma^5\delta^{10} \oplus \gamma^5\delta^{11} \oplus \gamma^6\delta^{12} \oplus \gamma^6\delta^{13} \oplus \gamma^7\delta^{14} \oplus \gamma^7\delta^{15} \oplus \dots$

Après simplification,

$$X_2(\delta) = \delta^1(\gamma^1\delta^2)^*$$

- $X_3(\delta) = \delta^1 \oplus \delta^2 \oplus \delta^3 \oplus \delta^4 \oplus \gamma^1\delta^5 \oplus \gamma^1\delta^6 \oplus \gamma^1\delta^7 \oplus \gamma^1\delta^8 \oplus \gamma^2\delta^9 \oplus \gamma^2\delta^{10} \oplus \gamma^2\delta^{11} \oplus \gamma^2\delta^{12} \oplus \gamma^3\delta^{13} \oplus \gamma^3\delta^{14} \oplus \gamma^3\delta^{15} \oplus \gamma^3\delta^{16} \oplus \gamma^4\delta^{17} \dots$

Ce qui donne :

$$X_3(\delta) = \delta^4(\gamma^1\delta^4)^*$$

Par conséquent, on trouve :

$$A^* E(\delta) = \begin{pmatrix} (\gamma^2\delta^2)(\gamma^3\delta^4)^* \\ \delta^1(\gamma^1\delta^2)^* \\ \delta^4(\gamma^1\delta^4)^* \end{pmatrix} E(\delta),$$

laquelle est la plus petite solution du système d'équations implicite suivant :

$$\begin{pmatrix} X_1(\delta) \\ X_2(\delta) \\ X_3(\delta) \end{pmatrix} = \begin{pmatrix} \gamma^3\delta^4 \\ \gamma^1\delta^2 \\ \gamma^1\delta^4 \end{pmatrix} \begin{pmatrix} X_1(\delta) \\ X_2(\delta) \\ X_3(\delta) \end{pmatrix} \oplus \begin{pmatrix} \gamma^2\delta^2 \\ \delta^1 \\ \delta^4 \end{pmatrix} E(\delta).$$

Ces équations admettent une représentation $(\min, +)$ linéaire dans le dioïde $\overline{\mathbb{Z}}_{\min}$, nous avons, $\forall t \in \mathbb{Z}$:

$$\begin{cases} x_1(t) = 3 \otimes x_1(t-4) \oplus 2 \otimes e(t-2), \\ x_2(t) = 1 \otimes x_2(t-2) \oplus e(t-1), \\ x_3(t) = 1 \otimes x_3(t-4) \oplus e(t-4). \end{cases}$$

lesquelles peut se représenter par le GET de la figure 3.12.

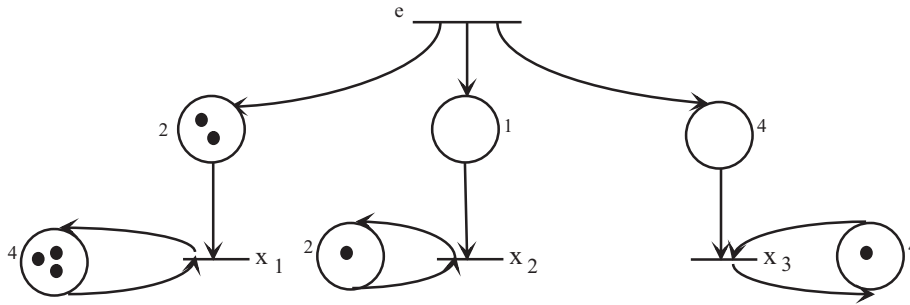


FIGURE 3.12 – GET obtenu après linéarisation.

Le GET de la figure 3.12, qui n'est pas fortement connexe, est composé de trois circuits : (x_1, x_1) , (x_2, x_2) et (x_3, x_3) . Chaque circuit admet un T-invariant composé d'une seule composante qui est égale à 1.

Utilisant l'équation (3.4), nous déduisons que chaque circuit, qui est un GET fortement connexe, admet le temps de cycle suivant :

- Circuit (x_1, x_1) , $TC = \frac{4}{3}$.
- Circuit (x_2, x_2) , $TC = \frac{2}{1}$.
- Circuit (x_3, x_3) , $TC = \frac{4}{1}$.

Le temps de cycle, TC_m , du GETG de la figure 3.10, correspond au temps moyen nécessaire pour franchir une fois le T-invariant minimal en mode de fonctionnement au plus tôt à partir de M_0 . Ce qui correspond dans la pratique au temps moyen pour fabriquer une pièce par un système de production par exemple.

$$TC_1 = 3 \times \frac{4}{3}, \quad TC_2 = 2 \times \frac{2}{1}, \quad TC_3 = 1 \times \frac{4}{1}.$$

Nous remarquons que ce temps de cycle est identique pour toutes les transitions du graphe. Il est égal à 4 u.t. Ceci signifie que chaque transition est franchie, en moyenne, une fois toutes les quatre unités de temps.

Utilisant la relation (3.5), nous déduisons la fréquence de franchissement associée à chaque transition du graphe :

$$\lambda_{m_1} = \frac{3}{4}, \quad \lambda_{m_2} = \frac{1}{2}, \quad \lambda_{m_3} = \frac{1}{4}.$$

3.5.2 Exemple : Extension au cas des GETG avec entrée/sortie

Considérons le GETG de la figure 3.13 avec une entrée et une sortie. Nous souhaitons calculer la fréquence de franchissement de la transition de sortie Y . Cette dernière correspond à la sortie des produits finis dans un système de production par exemple. Pour ce faire, hormis la transition Y , à chaque transition est associée une entrée impulsionnelle e .

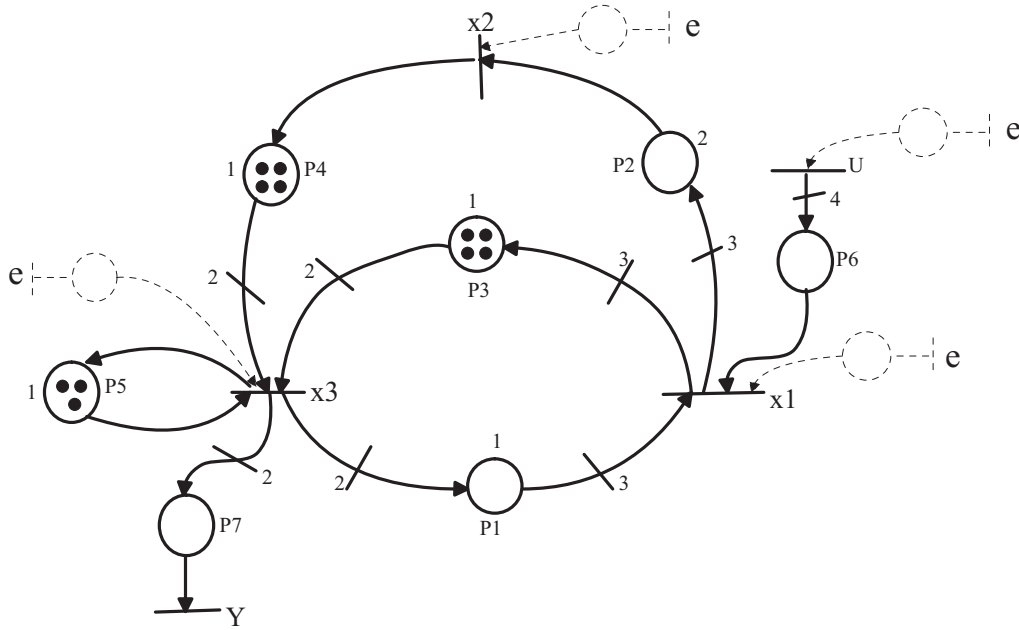


FIGURE 3.13 – GETG avec entrée/sortie.

L'évolution dynamique de chaque compteur dans le dioïde $\mathcal{O}_{\min}[\delta]$, est donnée

par la représentation d'état suivante :

$$\left\{ \begin{array}{l} \begin{pmatrix} X_1 \\ X_2 \\ X_3 \end{pmatrix} = \begin{pmatrix} \varepsilon & \varepsilon & \mu_{1/3}\delta\mu_2 \\ \mu_3\delta^2 & \varepsilon & \varepsilon \\ \mu_{1/2}\gamma^4\delta\mu_3 & \mu_{1/2}\gamma^4\delta & \gamma^3\delta \end{pmatrix} \begin{pmatrix} X_1 \\ X_2 \\ X_3 \end{pmatrix} \oplus \begin{pmatrix} e \\ e \\ e \end{pmatrix} E \\ Y = \begin{pmatrix} \varepsilon & \varepsilon & \mu_2 \end{pmatrix} \begin{pmatrix} X_1 \\ X_2 \\ X_3 \end{pmatrix} \end{array} \right. \quad (3.8)$$

• Notons que $\mu_4 U = \mu_4 E = E$

Ce système d'équation admet la solution suivante :

$$X = A^* \otimes E, \quad (3.9)$$

$A^* = e \oplus A \oplus A^2 \oplus \dots$. La proposition (2) permet de calculer A^*E :

$$A^*E = (e \oplus A \oplus A^2 \oplus A^3 \oplus \dots)E$$

$$= (E \oplus AE \oplus \underbrace{A \otimes AE}_{A^2E} \oplus \underbrace{A \otimes A^2E}_{A^3E} \oplus \dots).$$

Par conséquent, on a

$$\left\{ \begin{array}{l} \begin{pmatrix} X_1 \\ X_2 \\ X_3 \end{pmatrix} = A^*E = \begin{pmatrix} \delta(\gamma^1\delta^4)^* \\ (\gamma^3\delta^4)^* \\ \gamma\delta^4(\gamma^3\delta^8)^* \end{pmatrix} E, \\ Y = CA^*E = (\gamma^6\delta^8)^*\gamma^2\delta^4E. \end{array} \right. \quad (3.10)$$

En analysant le comportement dynamique dans le dioïde $\mathcal{O}_{\min}[\![\delta]\!]$ des compteurs associés aux transitions du graphe représenté par la figure 3.13, nous dédui-

sons la fréquence de franchissement associée à chaque transition.

- $X_1(\delta) = (\gamma\delta^4)^* \delta E = \delta(e \oplus \gamma\delta^4 \oplus \gamma^2\delta^8 \oplus \gamma^3\delta^{12} \oplus \gamma^4\delta^{16} \oplus \gamma^5\delta^{20} \oplus \gamma^6\delta^{24} \oplus \dots)$

L'évolution dynamique du compteur $X_1(\delta)$ montre que le franchissement de la transition X_1 est franchie une fois toutes les quatre unités de temps dès que le régime période est établi. Ce comportement est représenté par $(\gamma\delta^4)^*$. Par conséquent, $\lambda_1 = \frac{1}{4}$, ceci est équivalent, en moyenne, à 0.25 franchissement par unité de temps.

- $X_2(\delta) = (\gamma^3\delta^4)^* E = (e \oplus \gamma^3\delta^4 \oplus \gamma^6\delta^8 \oplus \gamma^9\delta^{12} \oplus \gamma^{12}\delta^{16} \oplus \gamma^{15}\delta^{20} \oplus \gamma^{18}\delta^{24} \oplus \dots)$

Concernant le comportement du compteur associé à la transition X_2 , nous remarquons que cette transition est franchie trois fois toutes les quatre unités de temps. Ce qui donne un $\lambda_2 = \frac{3}{4}$.

- $X_3(\delta) = (\gamma^3\delta^8)^* \gamma\delta^4 E(\delta) = (e \oplus \gamma^3\delta^8 \oplus \gamma^6\delta^{16} \oplus \gamma^9\delta^{24} \oplus \gamma^{12}\delta^{32} \oplus \gamma^{15}\delta^{40} \oplus \dots)$

A partir de $X_3(\delta)$, nous déduisons $\lambda_3 = \frac{3}{8}$.

Dans la suite, nous nous intéressons à la dynamique de la sortie Y décrite par la série formelle suivante dans le dioïde $\mathcal{O}_{\min}[[\delta]]$.

- $Y(\delta) = (\gamma^6\delta^8)^* \gamma^2\delta^4 E = \gamma^2\delta^4(e \oplus \gamma^6\delta^8 \oplus \gamma^{12}\delta^{16} \oplus \gamma^{18}\delta^{24} \oplus \gamma^{24}\delta^{32} \oplus \dots)$.

Notons que l'évolution dynamique du franchissement de la sortie Y atteint un régime périodique à $t=4$ u.t après deux franchissements. Ceci est modélisé par $\gamma^2\delta^4$ correspondant au régime transitoire. Le régime permanent est modélisé par $(\gamma^6\delta^8)^*$, ce qui donne $\lambda_Y = \frac{2}{3}$ par unité de temps quel que soit l'entrée U . Dans la pratique, ceci est équivalent à la production de deux pièces par trois unités de temps.

3.6 Conclusion

Dans ce chapitre, nous nous sommes intéressés à l'évaluation de performances des GETG. Pour pouvoir appliquer la théorie spectrale, développée dans le cadre d'analyse de performances des systèmes linéaires dans les dioïdes, une méthode de linéarisation a été proposée. Cette méthode a pour objectif d'obtenir un modèle admettant une représentation $(\min, +)$ linéaire. Contrairement aux méthodes existantes, cette méthode s'applique à tout GETG sans conditions restrictives sur le

marquage initial et sans expansion du réseau, pour obtenir un GET ordinaire.

Allocation de ressources des GETG

Sommaire

4.1	Introduction et état de l'art	77
4.2	Recherche des équations récurrentes linéaires	79
4.2.1	Présentation de la méthode de linéarisation utilisée	79
4.3	Allocation de ressources	85
4.3.1	Approche développée	86
4.3.2	Exemple illustratif	87
4.3.3	Application à un Atelier Flexible	92
4.4	Conclusion	99

4.1 Introduction et état de l'art

Une complexité croissante des systèmes de fabrication et de leur commande, ainsi que les coûts toujours plus élevés de leurs composants (machines à commande numérique, chariots filoguidés, robots, etc ...) ont rendu nécessaire le renforcement des phases d'évaluation et d'optimisation du comportement des systèmes durant leur phase de conception. Durant cette étape, il est nécessaire de définir les ressources ou bien les en-cours à utiliser (palettes de transport par exemple) afin d'être capable de réaliser la production voulue, de bonne qualité et correspondant surtout à un prix de revient le plus compétitif possible. Ce nombre d'en-cours reflète la productivité d'un système, puisqu'un nombre insuffisant ne permet pas d'atteindre la performance souhaitée. Cependant, un nombre trop important d'en-cours augmente les coûts de production, car ces pièces utilisent des stocks ou des supports de transport.

Dans cette partie, nous nous sommes intéressés à un problème qui a pour objectif de trouver le marquage à utiliser permettant de saturer le "processus goulot"

d'un GETG et qui minimise le coût des ressources.

Contrairement au cas des GET, peu de travaux ont été réalisés pour traiter ce type de problème dans le cas des GETG.

1. **Cohen et al.** présentent dans [Cohen 1983] une méthode donnant une répartition minimale des ressources dans un GET. Dans le cas où le taux de production visé est le taux du processus le plus lent, on parvient de de la sorte à saturer le "processus goulot".
2. **Hillion et Proth** dans [Hillion 1989] ont considéré un second problème analogue à celui présenté dans [Cohen 1983], qui consiste à minimiser un coût linéaire associé aux ressources, toujours sous la contrainte d'un taux de production donné. Ils ramènent ce problème à un problème de programmation linéaire en nombres entiers, dont les contraintes sont indicées par l'ensemble des circuits du graphe. Constatant que cette approche était rendue ineffective par l'impossibilité apparente d'énumérer les circuits (en trop grand nombre), ils formulent des heuristiques pour résoudre le problème.
3. L'équipe de **J.C Gentina** (Voir chapitre 7 [Lopez 2001]) a proposé une méthode qui permet d'ordonnancer les produits sur les machines en respectant le temps de cycle optimale, tout en optimisant un critère secondaire (le nombre ressource de transport).
4. Dans [Trouillet 2001] et [Trouillet 2003] à l'aide de l'algèbre spécifique (min, div), les auteurs transforment un GETG vérifiant certaines conditions en un GET ordinaire, indépendamment du marquage initial. Ils transforment donc le problème d'optimisation du marquage d'un GETG en un problème d'optimisation du marquage d'un GET.
5. Dans [Sauer 2003] l'approche proposée a été de travailler directement sur le GETG et d'exploiter les résultats obtenus dans le cas des GET ordinaires. Le principe de la méthode se base sur une heuristique. L'idée est d'enlever à chaque itération un jeton dans une place en répondant à deux objectifs fondamentaux : réduire au maximum le coût des ressources, et augmenter aussi

faiblement que possible le temps de cycle, c'est-à-dire diminuer au minimum la productivité.

Les méthodes existantes n'étaient pas satisfaisantes est surtout complexe à utiliser, car soit elles ne nous donnaient pas de résultats pour des graphes de taille importante, soit elles nous donnaient une solution approchée quelle que soit la taille du système, mais ne donnaient pas aucune indication sur la distance à la solution optimale.

Dans ce contexte, nous proposons une démarche simple est originale. Son principe consiste à déterminer le nombre minimum de jetons à utiliser dans un GETG permettant de saturer le "processus goulot" et d'atteindre les performances souhaitées, fixées préalablement. Pour cela, on procède par la recherche des équations récurrentes linéaires équivalentes, en utilisant des travaux déjà existants et certains résultats développés dans le cadre de l'analyse de performances permettant de déterminer le marquage qui minimise le coût des ressources à utiliser [Hamaci 2010, Benfekir 2012].

4.2 Recherche des équations récurrentes linéaires

4.2.1 Présentation de la méthode de linéarisation utilisée

Cette méthode développée dans [Hamaci 2005] s'applique sur des GETG consistants (cf. définition 2.15) et conservatifs (cf. définition 2.16). On note X le vecteur d'état du GETG. On définit un vecteur C de même dimension que X . Chaque composante de C de type compteur satisfait une équation récurrente $(\min, +)$ linéaire.

Cette méthode a pour principe d'exprimer chaque composante de X en fonction d'une composante de C . Le fait que le GETG considéré soit consistant implique l'existence d'un T-invariant sur les transitions. Comme on a vu au chapitre 2, un T-invariant est un vecteur dont les composantes ne sont pas forcément unitaires, contrairement à un GET où toutes les composantes du T-invariant sont unitaires. En effet, chaque compteur $x_q(t)$ associé à la transition x_q du GETG s'exprime comme suit :

$$\forall q \in \mathcal{T}, \forall t \in \mathbb{Z}, x_q(t) = \theta_q c_q(t), \quad (4.1)$$

où :

- $x_q(t)$ est le compteur associé à la transition x_q du GETG,
- θ_q est la composante ($\in \mathbb{N}^*$) du T-invariant associée à la transition x_q du GETG.

Contrairement à la méthode proposée par Alix Munier, où à chaque transition x_q du GETG correspond θ_q représentants dans le GET obtenu après linéarisation (ce qui provoque une expansion du graphe), le changement de variables (4.1) signifie que durant un cycle, chaque tir d'une transition c_q du GET est équivalent à θ_q tirs de la transition x_q dans le GETG, ce qui donne l'équivalence entre le comportement du GETG et le GET obtenu après linéarisation.

Pour que le changement de variable (4.1) soit possible, c'est-à-dire pour que le GETG soit linéarisable, chaque compteur $c_q(t)$ doit appartenir au dioïde \mathbb{Z}_{min} . La proposition suivante donne la condition pour que le GETG soit linéarisable.

Proposition 4 Un GETG est *linéarisable* si :

$$\forall q \in \mathcal{T}, \forall p \in \bullet q, \quad \frac{1}{\theta_q} \lfloor \frac{m_p}{M_{qp}} \rfloor \in \mathbb{N}. \quad (4.2)$$

Preuve 5 D'après l'assertion 1 (chapitre 2), chaque compteur $x_q(t)$ d'un GETG satisfait la relation suivante :

$$x_q(t) = \min_{p \in \bullet q, q' \in \bullet p} \lfloor M_{qp}^{-1}(m_p + M_{pq'}x_{q'}(t - \tau_p)) \rfloor.$$

En utilisant le changement de variable (4.1) et la distributivité de la multiplication par rapport à l'opérateur min, on obtient l'expression $c_q(t)$ suivante :

$$c_q(t) = \min_{p \in \bullet q, q' \in \bullet p} \frac{1}{\theta_q} \lfloor (\frac{m_p}{M_{qp}} + \frac{M_{pq'}}{M_{qp}} x_{q'}(t - \tau_p)) \rfloor.$$

Notons $\theta = (\theta_1 \ \theta_2 \dots \theta_n)^t$ le vecteur T-invariant. Ce vecteur satisfait l'expression $\mathcal{C} \times \theta = 0$, où \mathcal{C} est la matrice d'incidence associée au GETG. Il en résulte, sachant que les graphes sont consistants et conservatifs, que :

$$\forall q', q \in \mathcal{T} \quad p \in P, M_{qp} \theta_q - \theta_{q'} M_{pq'} = 0.$$

De cette relation, on déduit que :

$$\frac{\theta_q}{M_{pq'}} = \frac{\theta_{q'}}{M_{qp}},$$

par conséquent,

$$c_q(t) = \min_{p \in \bullet q, q' \in \bullet p} \frac{1}{\theta_q} \lfloor \left(\frac{m_p}{M_{qp}} + \frac{\theta_q}{\theta_{q'}} x_{q'}(t - \tau_p) \right) \rfloor,$$

c'est-à-dire,

$$c_q(t) = \min_{p \in \bullet q, q' \in \bullet p} \frac{1}{\theta_q} \lfloor \left(\frac{m_p}{M_{qp}} + \theta_q c_{q'}(t - \tau_p) \right) \rfloor.$$

Puisque $\theta_q c_{q'}(t - \tau_p)$ est un entier, nous obtenons :

$$c_q(t) = \min_{p \in \bullet q, q' \in \bullet p} \left(\frac{1}{\theta_q} \lfloor \frac{m_p}{M_{qp}} \rfloor + c_{q'}(t - \tau_p) \right), \quad (4.3)$$

ce qui correspond à une équation récurrente $(\min, +)$ linéaire dès lors que $\frac{1}{\theta_q} \lfloor \frac{m_p}{M_{qp}} \rfloor \in \mathbb{N}$. Cette équation se code dans le dioïde $\mathcal{O}_{\min}[\![\delta]\!]$ de la façon suivante :

$$C_q(\delta) = \bigoplus_{p \in \bullet q, q' \in \bullet p} \gamma^{\frac{1}{\theta_q} \lfloor \frac{m_p}{M_{qp}} \rfloor} \delta^{\tau_p} C_{q'}(\delta). \quad (4.4)$$

Remarque 4.8 Définissons une classe d'équivalence au niveau des marquages initiaux induite par la relation d'équivalence suivante :

$$m' \equiv m'' \Leftrightarrow \forall q \in \mathcal{T}, \forall p \in \bullet q, \quad \lfloor \frac{m'_p}{M_{qp}} \rfloor = \lfloor \frac{m''_p}{M_{qp}} \rfloor. \quad (4.5)$$

Tous les marquages initiaux appartenant à une même classe d'équivalence engendrent un même comportement au niveau des transitions du graphe. Aussi, pour ces mêmes marquages, le GETG admet, après linéarisation, un même modèle $(\min, +)$ linéaire.

Dans le cas où la condition de linéarisation (4.2) n'est pas satisfaite, une alternative consiste à utiliser la linéarisation approchée correspondant à un ajout/retrait de jetons dans le GETG.

Le principe consiste à approcher autant que possible, en utilisant la condition de linéarisation (4.2), le comportement du GETG initial. Il en résulte une approximation par valeur inférieure (resp., supérieure) obtenue par un retrait (resp., ajout) minimal de jetons dans les places ne satisfaisant pas la condition de linéarisation. Autrement dit, dans chaque place p pour laquelle $\lfloor \frac{m_p}{M_{qp}} \rfloor \notin \theta_q \mathbb{N}$, nous retirons autant

de jetons que nécessaire du marquage initial, afin que la condition de linéarisation soit vérifiée.

On note \underline{C} (resp., \overline{C}) le vecteur d'état du GET ordinaire obtenu à l'issue de la linéarisation approchée par retrait (resp., ajout) de jetons dans le GETG.

En reprenant le même raisonnement que celui permettant d'établir la proposition 4, nous obtenons :

$$\overline{C}_q(\delta) = \bigoplus_{p \in \bullet q, q' \in \bullet p} \gamma^{\frac{1}{\theta_q} \lfloor \frac{m_p + \overline{m}_p}{M_{qp}} \rfloor} \delta^{\tau_p} \overline{C}_{q'}(\delta). \quad (4.6)$$

où \overline{m}_p est le nombre minimum de jetons ajoutés dans la place p tel que $\lfloor \frac{m_p + \overline{m}_p}{M_{qp}} \rfloor \in \theta_q \mathbb{N}$.

De même, on a :

$$\underline{C}_q(\delta) = \bigoplus_{p \in \bullet q, q' \in \bullet p} \gamma^{\frac{1}{\theta_q} \lfloor \frac{m_p - \underline{m}_p}{M_{qp}} \rfloor} \delta^{\tau_p} \underline{C}_{q'}(\delta). \quad (4.7)$$

où \underline{m}_p est le nombre minimum de jetons retirés de la place p tel que $\lfloor \frac{m_p - \underline{m}_p}{M_{qp}} \rfloor \in \theta_q \mathbb{N}$.

Sachant que le retrait (resp., l'ajout) de jetons dans un GETG ralentit (resp., accélère) l'évolution du graphe, on a par conséquent :

$$\forall q, \quad \theta_q \underline{C}_q(t) = \underline{x}_q(t) \leq x_q(t) \leq \overline{x}_q(t) = \theta_q \overline{C}_q(t).$$

La fréquence de franchissement (3.5) d'un GETG linéarisable, peut être calculé directement du modèle $(\min, +)$ linéaire obtenu après linéarisation, en utilisant la relation $\lambda_{m_q} = \theta_q \lambda$, déduite directement du changement de variable. Par conséquent, $TC_m = TC$.

Exemple 4.22 Considérons le GETG représenté par la figure 4.1. Son évolution dynamique est donnée par les équations suivantes :

$$\begin{cases} X_1(\delta) = \mu_{\frac{1}{3}} \gamma^6 \delta^1 \mu_2 X_3(\delta), \\ X_2(\delta) = \mu_{\frac{1}{2}} \delta^2 \mu_3 X_1(\delta) \oplus \gamma^3 \delta^3 \mu_3 X_4(\delta), \\ X_3(\delta) = \delta^1 X_2(\delta), \\ X_4(\delta) = \mu_{\frac{1}{3}} \delta^1 X_3(\delta). \end{cases}$$

Comme nous l'avons constaté précédemment, le système ainsi obtenu n'est pas $(\min, +)$ linéaire en raison de la présence de parties entières rendant les équations non linéaires. Pour linéariser ces équations, on commence par rechercher l'invariant total de transitions. Cet invariant correspond à $\Theta^t = (2 \ 3 \ 3 \ 1)$. On vérifie

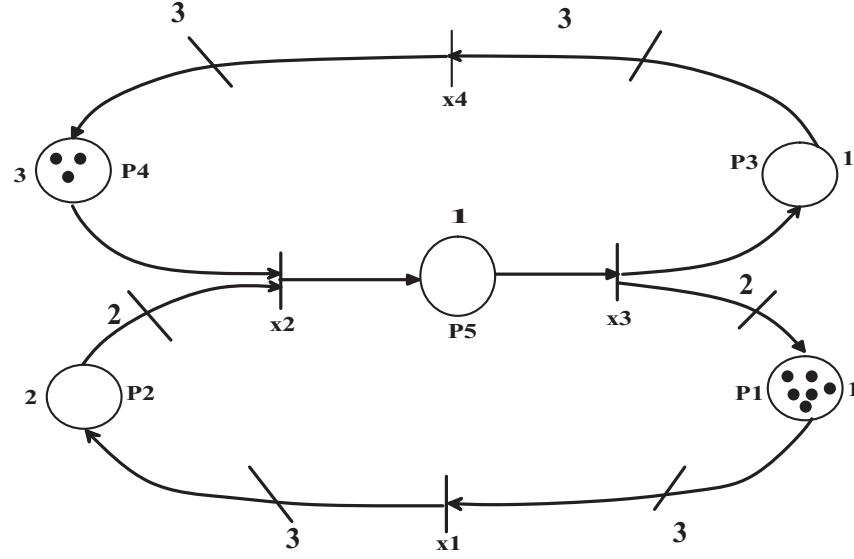


FIGURE 4.1 – Exemple de GETG à linéariser.

que le marquage de chaque place satisfait la condition (4.2), ce qui signifie que le GETG est linéarisable. En utilisant le changement de variable (4.1), on obtient le modèle suivant évoluant dans le dioïde $\mathcal{O}_{\min}[\delta]$, issu de l'équation (4.4) :

$$\begin{cases} C_1(\delta) = \gamma^1 \delta^1 C_3(\delta), \\ C_2(\delta) = \gamma^1 \delta^3 C_4(\delta) \oplus \delta^2 C_1(\delta), \\ C_3(\delta) = \delta^1 C_2(\delta), \\ C_4(\delta) = \delta^1 C_3(\delta). \end{cases}$$

Ces équations modélisent le comportement du GET représenté par la figure 4.2, elle admettent la représentation suivante dans l'algèbre $(\min, +)$.

$$\begin{cases} c_1(t) = 1 \otimes c_3(t-1), \\ c_2(t) = 1 \otimes c_4(t-3) \oplus c_1(t-2), \\ c_3(t) = c_2(t-1), \\ c_4(t) = c_3(t-1). \end{cases}$$

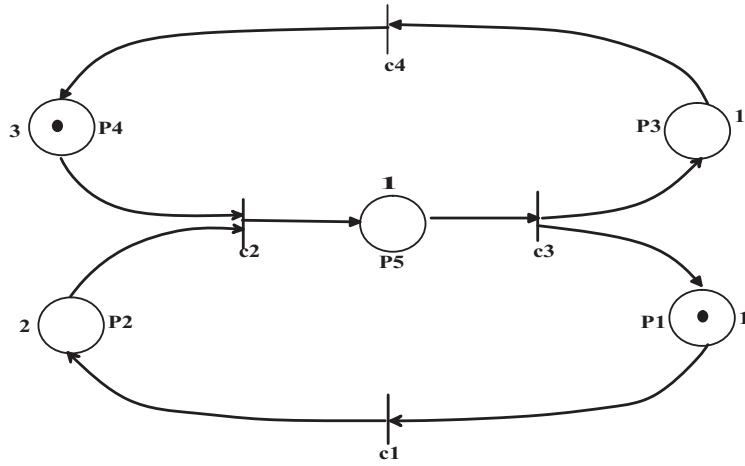


FIGURE 4.2 – GET obtenu après linéarisation.

Pour le marquage initial suivant : $M(0) = (6 \ 0 \ 0 \ 4 \ 0)$, on note que la place p_4 , marqué avec un jeton, ne satisfait pas la condition de linéarisation (4.2), ce qui motive l'utilisation de la méthode de linéarisation approchée par ajout/retrait de jetons dans le graphe. Le retrait d'un jeton dans la place p_4 permet de vérifier la condition (4.2), de même que l'ajout deux jetons dans cette place.

L'utilisation des équations (4.6) et (4.7) permet d'obtenir les modèles approchés suivants qui se codent facilement dans le dioïde $\mathcal{O}_{\min}[[\delta]]$:

$$\begin{cases} \overline{C}_1(\delta) = \gamma^1 \delta^1 \overline{C}_3(\delta), \\ \overline{C}_2(\delta) = \gamma^2 \delta^3 \overline{C}_4(\delta) \oplus \delta^2 \overline{C}_1(\delta), \\ \overline{C}_3(\delta) = \delta^1 \overline{C}_2(\delta), \\ \overline{C}_4(\delta) = \delta^1 \overline{C}_3(\delta). \end{cases}$$

et

$$\begin{cases} \underline{C}_1(\delta) = \gamma^1 \delta^1 \underline{C}_3(\delta), \\ \underline{C}_2(\delta) = \gamma^1 \delta^3 \underline{C}_4(\delta) \oplus \delta^2 \underline{C}_1(\delta), \\ \underline{C}_3(\delta) = \delta^1 \underline{C}_2(\delta), \\ \underline{C}_4(\delta) = \delta^1 \underline{C}_3(\delta). \end{cases}$$

Ces systèmes d'équations s'expriment linéairement dans l'algèbre $(\min, +)$:

$$\begin{cases} \bar{c}_1(t) = 1 \otimes \bar{c}_3(t-1), \\ \bar{c}_2(t) = 2 \otimes \bar{c}_4(t-3) \oplus \bar{c}_1(t-2), \\ \bar{c}_3(t) = \bar{c}_2(t-1), \\ \bar{c}_4(t) = \bar{c}_3(t-1). \end{cases}$$

et

$$\begin{cases} \underline{c}_1(t) = 1 \otimes \underline{c}_3(t-1), \\ \underline{c}_2(t) = 1 \otimes \underline{c}_4(t-3) \oplus \underline{c}_1(t-2), \\ \underline{c}_3(t) = \underline{c}_2(t-1), \\ \underline{c}_4(t) = \underline{c}_3(t-1). \end{cases}$$

4.3 Allocation de ressources

Dans cette section, nous présentons la démarche développée pour déterminer les ressources à utiliser pour atteindre des performances souhaitées.

Définition 4.35 (Temps de Cycle atteignable). On appelle un Temps de Cycle atteignable, TC_a , tout temps de cycle vérifiant les conditions suivantes :

$$\begin{cases} TC_a \leq T(c_g), \\ TC_a \in \left\{ \frac{T(c_g)}{n} \right\}, n \in \mathbb{N}^*. \end{cases}$$

- $T(c_g)$: Correspond à la somme des temporisations dans le circuit modélisant le processus goulot.

Ces deux conditions permettent de préserver l'aspect discret des jetons dans le circuit modélisant le processus goulot.

4.3.1 Approche développée

Dans cette partie, on expose le principe de l'approche développée, dans le but de déterminer la quantité de ressources à utiliser permettant d'atteindre les performances souhaitées.

•**Etape 1** Déterminer un temps de cycle souhaité et atteignable, TC_a .

•**Etape 2** Si la condition de linéarisation est vérifiée pour toutes les places du graphe :

Etape 2.1 Établir le modèle mathématique (min, +) linéaire

Etape 2.2 Construire le GET équivalent.

Etape 2.4 Déterminer le marquage global du GET obtenu après linéarisation en utilisant la relation 3.4.

Etape 2.5 Définir les différents marquages initiaux M_0 possibles du GETG permettant d'atteindre les performances souhaitées, en utilisant la condition de linéarisation 4.2.

Etape 2.6 Déterminer le meilleur marquage m_{op} qui soit solution du problème d'optimisation suivant :

$$\begin{cases} \text{Minimiser} & f(M_0) = \mathcal{K}.M_0, \\ \text{Sous les contraintes,} & \\ TC_m = TC_a, M_0 \in IN^+, \forall p \in P & \end{cases} \quad (4.8)$$

Où

- TC_a est le temps de cycle désiré, atteignable. Il peut aussi correspondre au temps de cycle du processus goulot ;
- TC_m est le temps de cycle obtenu à partir du marquage M_0 ;
- $f(M_0)$ représente le coût des en-courts du système de fabrication considéré. Comme $\mathcal{K} = (k_1, k_2, \dots, k_n)$ est un P-invariant, $f(M_0) = f(M_a)$ pour tout marquage accessible M_a à partir de M_0 .

•**Etape 3** Dans le cas où la condition de linéarisation n'est pas vérifiée, nous procédons par trois étapes :

Etape 3.1 Ajouter des jetons dans les places où la condition de linéarisation n'est pas satisfaite, sachant que les jetons ajoutés n'auront aucune influence sur la vitesse d'évolution du graphe. Une fois que la condition de linéarisation est

vérifiée pour toutes les places avec le nouveau marquage, nous reproduisons l'approche adoptée à l'étape 1 et à l'étape 2.

Etape 3.2 Enlever des jetons dans les places où la condition de linéarisation n'est pas satisfaite et de reproduire la même démarche que précédemment une fois la vivacité du graphe est assurée.

Etape 3.3 Vérifier que le temps de cycle TC_a , fixé initialement est atteignable dans les deux cas de la linéarisation approchée (Etape 3.1 et Etape 3.2).

4.3.2 Exemple illustratif

Le GETG représenté par la figure 4.3, modélise une partie de fabrication d'un produit, noté A . Ce dernier résulte de l'assemblage de deux types de produit, A_1 et A_2 , après avoir été traité avec deux machines différentes : Machine M_1 est modélisée par le circuit 1, composé de p_1 et p_2 ; Machine M_2 est modélisée par le circuit 2, composé des places p_3 et p_4 . La vitesse d'évolution de cet ensemble est contrôlée par la machine M_1 qui est menante.

Pour ce système, nous cherchons à définir le marquage des places p_1 et p_2 pour atteindre un TC_a égal à 2.5 u.t. Ce système admet un T-invariant $\Theta = (2, 3, 1)$ et un P-invariant $\mathcal{K} = (1, 1, 1, 1)$. Dans la suite, que la partie fortement connexe du GETG représenté par la figure 4.3 est considérée. Le modèle mathématique régissant l'évolution dynamique de ce graphe est donné par le système d'équations suivant :

$$\begin{cases} X_1(\delta) &= \mu_{\frac{1}{3}} \gamma^{m_1} \delta^4 \mu_2 X_2(\delta) \\ X_2(\delta) &= \mu_{\frac{1}{2}} \gamma^{m_2} \delta^1 \mu_3 X_1(\delta) \oplus \gamma^3 \delta^1 \mu_3 X_3(\delta) \\ X_3(\delta) &= \mu_{\frac{1}{3}} X_2(\delta). \end{cases}$$

Pour ce GETG, toutes les places vérifient la condition (4.2). Après linéarisation, on obtient le modèle linéaire suivant :

$$\begin{cases} C_1(\delta) &= \gamma^{\frac{1}{2} \lfloor \frac{m_1}{3} \rfloor} \delta^4 C_2(\delta), \\ C_2(\delta) &= \gamma^{\frac{1}{3} \lfloor \frac{m_2}{2} \rfloor} \delta^1 C_1(\delta) \oplus \gamma^1 \delta^1 C_3(\delta), \\ C_3(\delta) &= C_2(\delta). \end{cases}$$

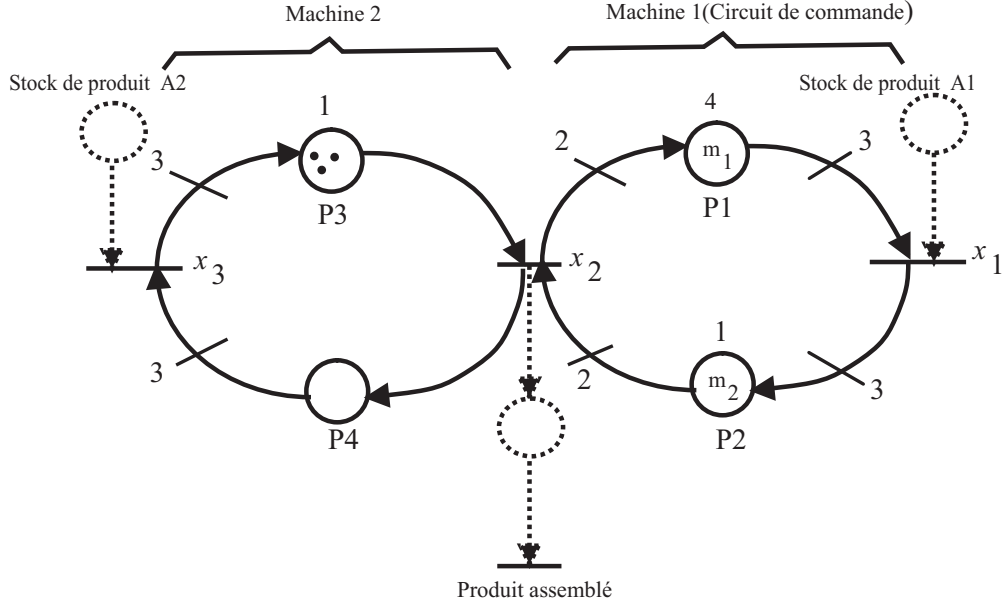


FIGURE 4.3 – GETG modélisant deux machines de production.

Ces équations modélisent la dynamique du GET représenté par la figure 4.4, dans le dioïde $\mathcal{O}_{\min}[[\delta]]$. Elle se codent linéairement dans l'algèbre $(\min, +)$ de la manière suivante :

$$\begin{cases} c_1(t) &= \frac{1}{2} \lfloor \frac{m_1}{3} \rfloor \otimes c_2(t + 4), \\ c_2(t) &= \frac{1}{3} \lfloor \frac{m_2}{2} \rfloor \otimes c_1(t - 1) \oplus 1 \otimes c_3(t - 1), \\ c_3(t) &= c_2(t). \end{cases}$$

Pour obtenir un modèle $(\min, +)$ linéaire, la valeur de $\frac{1}{2} \lfloor \frac{m_1}{3} \rfloor = m'_1$ et $\frac{1}{3} \lfloor \frac{m_2}{2} \rfloor = m'_2$ doivent appartenir à \mathbb{N} . Pour assurer la vivacité du circuit 1, la somme $m'_1 + m'_2$ doit être supérieure à 0.

Le nombre total de jeton dans le processus goulot est donné par la relation 3.4 : $M(c_g) = m'_1 + m'_2 = \frac{T(c_g)}{TC_m}$. Ceci est équivalent à : $M(c_g) = \frac{1+4}{2.5} = 2$. Par conséquent, les marquages possibles du graphe sont : $[2 \ 0 \ 1 \ 0]$, $[0 \ 2 \ 1 \ 0]$ et $[1 \ 1 \ 1 \ 0]$.

A partir de ces marquages, utilisant la condition de linéarisation (4.2), on déduit le marquage m_1 et m_2 du GETG initial.

- Concernant le marquage $[2 \ 0 \ 1 \ 0]$, nous avons $\frac{1}{2} \lfloor \frac{m_1}{3} \rfloor = 2 \Rightarrow m_1 \in \{12, 13, 14\}$.
- Concernant le marquage $[0 \ 2 \ 1 \ 0]$, nous avons $\frac{1}{3} \lfloor \frac{m_2}{2} \rfloor = 2 \Rightarrow m_2 \in \{12, 13\}$.
- Concernant le marquage $[1 \ 1 \ 1 \ 0]$, nous avons $\frac{1}{2} \lfloor \frac{m_1}{3} \rfloor = 1 \Rightarrow m_1 \in \{6, 7, 8\}$ et $\frac{1}{3} \lfloor \frac{m_2}{2} \rfloor = 2 \Rightarrow m_2 \in \{6, 7, 8\}$.

Utilisant la relation (4.8), nous calculons la quantité $\mathcal{K}.M_0$ pour chaque marquage M_0 . Cette quantité, d'un point de vue théorique, correspond à la constante de conservation de jetons dans le graphe. On cherche le marquage qui minimise cette valeur. Cette dernière correspond, dans la pratique, au coût des en-cours utilisés.

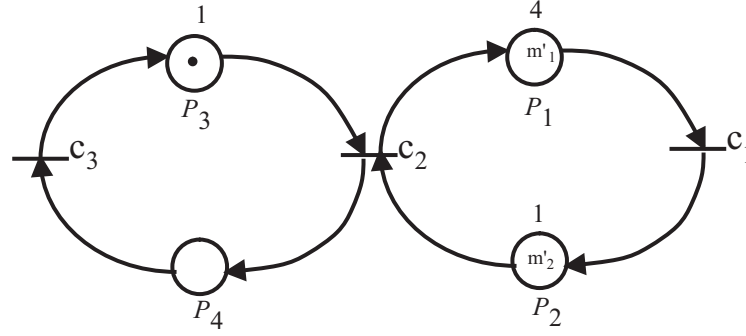


FIGURE 4.4 – GET obtenu après linéarisation du modèle mathématique associé à la figure 4.3.

- Cas où le marquage initial du GETG est égal à :
 - $[12 \ 0 \ 3 \ 0] \Rightarrow TC_m = TC_a = 2.5, f(M_0) = \mathcal{K}.M_0 = 15$
 - $[13 \ 0 \ 3 \ 0] \Rightarrow TC_m = TC_a = 2.5, f(M_0) = \mathcal{K}.M_0 = 16$
 - $[14 \ 0 \ 3 \ 0] \Rightarrow TC_m = TC_a = 2.5, f(M_0) = \mathcal{K}.M_0 = 17$

On remarque que le marquage $[12 \ 0 \ 3 \ 0]$ minimise le coût des en-cours utilisés. Par conséquent, il correspond au marquage m_{op} .

En analysant les résultats de la simulation du comportement dynamique des compteurs associés aux différentes transitions du graphe (figure 4.5), nous remarquons que chaque transition est franchie deux fois sa composante de T-invariant durant 5 u.t, une fois le régime permanent établi. Ce qui donne un temps de cycle de 2.5 u.t.

- Cas où le marquage initial du GETG est égal à :

- $[0 \ 12 \ 3 \ 0] \Rightarrow TC_m = TC_a = 2.5, f(M_0) = \mathcal{K}.M_0 = 15$
- $[0 \ 13 \ 3 \ 0] \Rightarrow TC_m = TC_a = 2.5, f(M_0) = \mathcal{K}.M_0 = 16$

On remarque que $[0 \ 12 \ 3 \ 0]$ correspond au marquage m_{op} qui minimise le coût des en-cours utilisés.

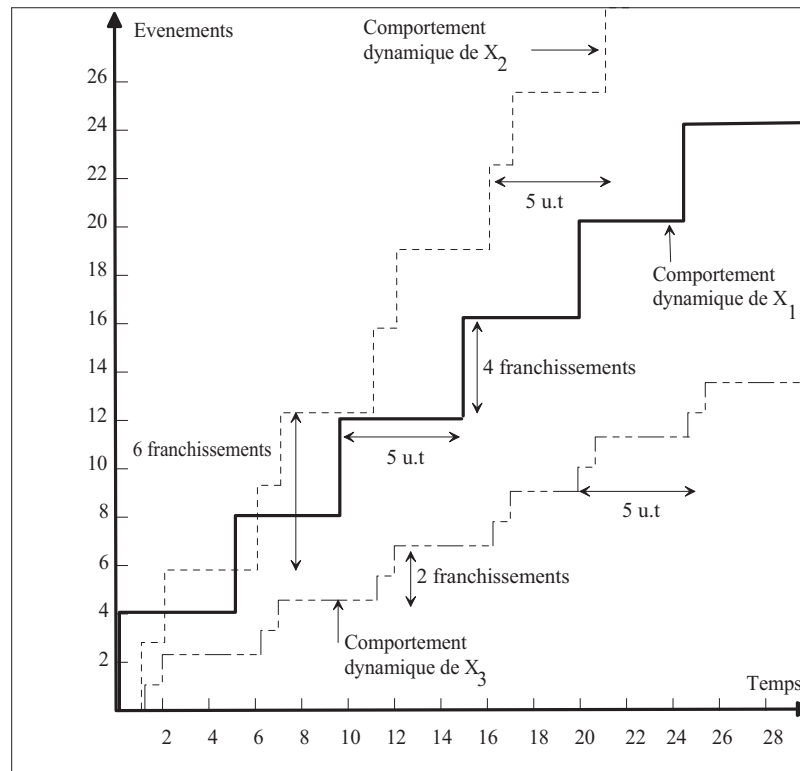


FIGURE 4.5 – Evolution des différents compteurs pour le marquage $[12 \ 0 \ 3 \ 0]$.

- Cas où le marquage initial du GETG est égal à :

- $[6 \ 6 \ 3 \ 0] \Rightarrow TC_m = TC_a = 2.5, f(M_0) = \mathcal{K}.M_0 = 15$
- $[6 \ 7 \ 3 \ 0] \Rightarrow TC_m = TC_a = 2.5, f(M_0) = \mathcal{K}.M_0 = 16$
- $[6 \ 8 \ 3 \ 0] \Rightarrow TC_m = TC_a = 2.5, f(M_0) = \mathcal{K}.M_0 = 17$
- $[7 \ 6 \ 3 \ 0] \Rightarrow TC_m = TC_a = 2.5, f(M_0) = \mathcal{K}.M_0 = 16$
- $[8 \ 6 \ 3 \ 0] \Rightarrow TC_m = TC_a = 2.5, f(M_0) = \mathcal{K}.M_0 = 17$

On remarque que $[6 \ 6 \ 3 \ 0]$ correspond au marquage m_{op} qui minimise le coût des

en-cours utilisés.

Le GETG de la figure 4.3 forme une composante conservative pour les trois marquages : $[12 \ 0 \ 3 \ 0]$, $[0 \ 12 \ 3 \ 0]$ et $[6 \ 6 \ 3 \ 0]$. Ces derniers sont des marquages accessibles et atteignables, une fois le régime permanent établi, quel que soit le marquage choisi.

• **Cas où la condition de linéarisation n'est pas satisfaite.**

Pour le marquage initial suivant : $M_0 = [m_1 \ m_2 \ 4 \ 0]$, la place p_3 ne satisfait pas la condition de linéarisation (4.2). Pour pallier à ce problème, nous procédons en trois étapes :

• **Etape 1** Ajout de deux jetons dans la place p_3 pour vérifier la condition de linéarisation (4.2).

On note que l'ajout de jetons dans la place p_3 fait accélérer l'évolution du circuit composé des places (p_3, p_4) , cependant l'évolution dynamique du graphe est contrôlé par le circuit de commande (p_1, p_2) avec un temps de cycle de 2.5 u.t.

L'utilisation de l'équation (4.6) permet d'obtenir un modèle $(\min, +)$ linéaire approché :

$$\begin{cases} \overline{C}_1(\delta) &= \gamma^{\frac{1}{2} \lfloor \frac{m_1}{3} \rfloor} \delta^4 \overline{C}_2(\delta), \\ \overline{C}_2(\delta) &= \gamma^{\frac{1}{3} \lfloor \frac{m_2}{2} \rfloor} \delta^1 \overline{C}_1(\delta) \oplus \gamma^2 \delta^1 \overline{C}_3(\delta), \\ \overline{C}_3(\delta) &= \overline{C}_2(\delta). \end{cases}$$

Ces équations modélisent le comportement dynamique du GET suivant, donné par la figure 4.6.

Utilisant la même démarche que précédemment, nous déduisons que le marquage $[12 \ 0 \ 6 \ 0]$ permet d'obtenir un temps de cycle de 2.5 u.t .

• **Etape 2** Retrait de un jeton dans la place p_3 pour vérifier la condition de linéarisation (4.2). Sachant que le retrait de jeton du graphe peut ralentir l'évolution du graphe. L'utilisation de l'équation (4.7) permet d'obtenir un modèle $(\min, +)$ linéaire approché :

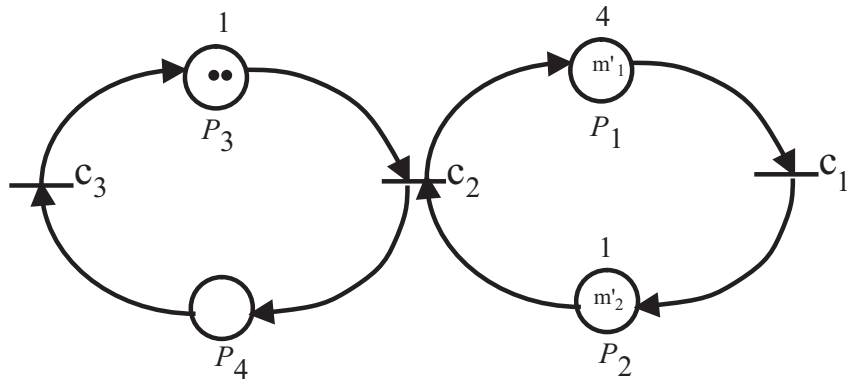


FIGURE 4.6 – GET obtenu après linéarisation du GETG en utilisant l'équation (4.6).

$$\begin{cases} \underline{C}_1(\delta) &= \gamma^{\frac{1}{2} \lfloor \frac{m_1}{3} \rfloor} \delta^4 \underline{C}_2(\delta), \\ \underline{C}_2(\delta) &= \gamma^{\frac{1}{3} \lfloor \frac{m_2}{2} \rfloor} \delta^1 \underline{C}_1(\delta) \oplus \gamma^1 \delta^1 \underline{C}_3(\delta), \\ \underline{C}_3(\delta) &= \underline{C}_2(\delta). \end{cases}$$

Ces équations modélisent le comportement dynamique du GET représenté par la figure 4.7. Suivant la même démarche que précédemment, nous déduisons que le marquage $[12 \ 0 \ 3 \ 0]$ permet d'obtenir un temps de cycle de 2.5 u.t.

• **Etape 3** On remarque que les deux marquages, $[12 \ 0 \ 6 \ 0]$ et $[12 \ 0 \ 3 \ 0]$, permettent d'obtenir un Temps de Cycle de 2.5 u.t. Par conséquent, le marquage $[12 \ 0 \ 4 \ 0]$ permet d'obtenir un Temps de Cycle de 2.5 u.t du fait que $3 \leq (m_{p_3} = 4) \leq 6$.

4.3.3 Application à un Atelier Flexible

Considérons un atelier flexible représenté par le GETG de la figure 4.10. Il est composé de trois types de machines pour fabriquer deux types de produits : Produit A et produit B. Chaque produit subit successivement une série d'opérations dans un ordre bien défini. Les gammes de fabrication de ces produits sont données par :

- Produit A : Machine 1, Machine 2, Machine 3.
- Produit B : Machine 2, Machine 1.

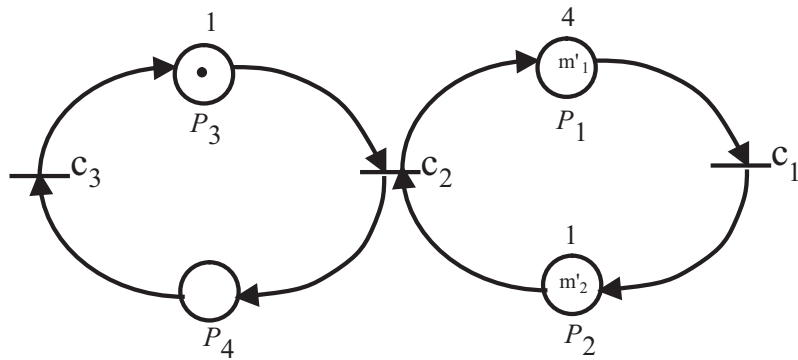


FIGURE 4.7 – GET obtenu après linéarisation du GETG en utilisant l'équation (4.7).

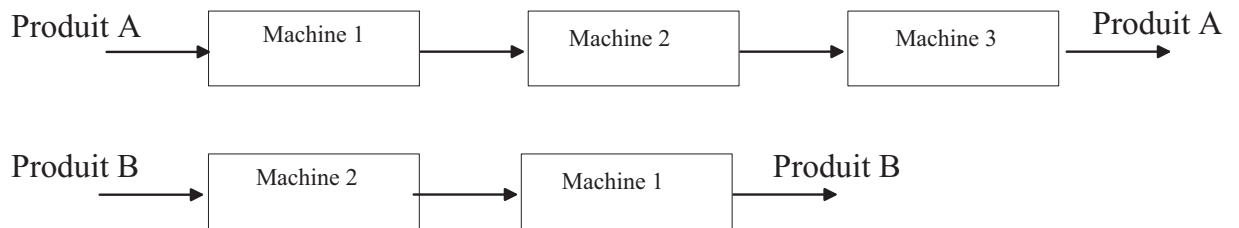


FIGURE 4.8 – Gamme de fabrication.

Les durées nécessaires pour effectuer les opérations considérées sur chacune des machines sont données par la table suivante :

	Produit A	Produit B
Machine 1	1	2
Machine 2	3	1
Machine 3	2	X

Nous souhaitons fabriquer $3/5$ du produit A et $2/5$ du produit B. Les séquences d'entrée sur les machines respectant ces ratios, sont les suivantes : A, A, A, B, B sur Machine 1, Machine 2, alors que la Machine 3 ne fabrique que des produits A.

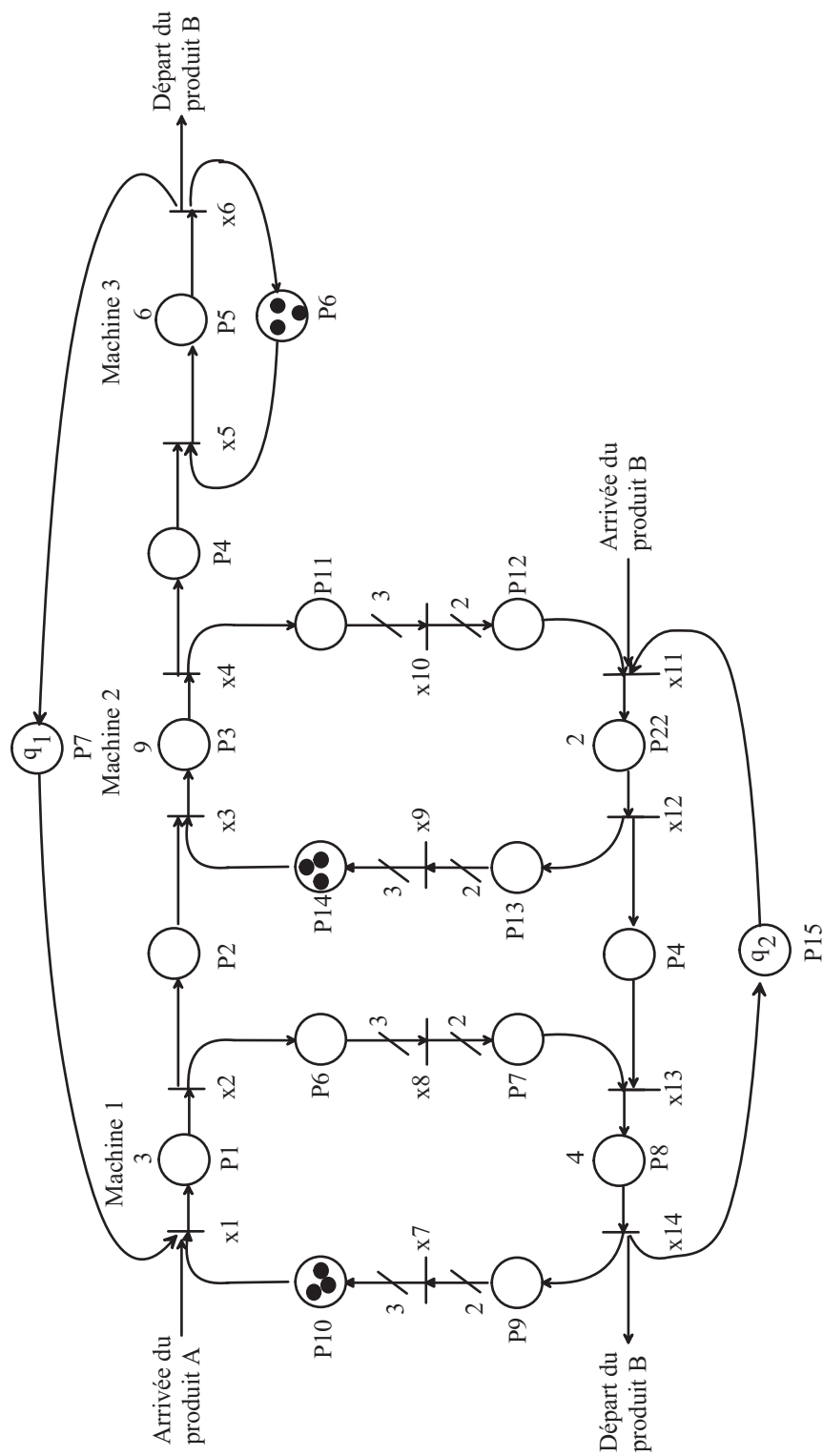


FIGURE 4.9 – Atelier flexible, 3 machines.

Ces produits sont véhiculés par des ressources de transport (palettes) qui sont disponibles en nombre limité dans l'atelier. La ressource de transport suit le produit tout le long de sa fabrication et est libérée et à nouveau disponible à l'entrée de la première machine sitôt que le produit est terminé. Dans la suite, nous faisons l'hypothèse que chaque ressources de transport ne peut transporter qu'un seul type de produit.

L'évolution dynamique de ce GETG, est donné par les équations récurrentes suivantes codées dans le dioïde $\mathcal{O}_{\min}[[\delta]]$:

$$\left\{ \begin{array}{l} X_1(\delta) = \gamma^{q_1} X_6(\delta) \oplus \gamma^3 \mu_3 X_7(\delta), \\ X_2(\delta) = \delta^3 X_1(\delta), \\ X_3(\delta) = \gamma^3 \mu_3 X_9(\delta) \oplus X_2(\delta), \\ X_4(\delta) = \delta^9 X_3(\delta), \\ X_5(\delta) = \gamma^3 X_6(\delta) \oplus X_4(\delta), \\ X_6(\delta) = \delta^6 X_5(\delta), \\ X_7(\delta) = \mu_{\frac{1}{2}} X_{14}(\delta), \\ X_8(\delta) = \mu_{\frac{1}{3}} X_2(\delta), \\ X_9(\delta) = \mu_{\frac{1}{2}} X_{12}(\delta), \\ X_{10}(\delta) = \mu_{\frac{1}{3}} X_4(\delta), \\ X_{11}(\delta) = \mu_2 X_{10}(\delta) \oplus \gamma^{q_2} X_{14}(\delta), \\ X_{12}(\delta) = \delta^2 X_{11}(\delta), \\ X_{13}(\delta) = \mu_2 X_8(\delta) \oplus X_{12}(\delta), \\ X_{14}(\delta) = \delta^4 X_{13}(\delta). \end{array} \right.$$

Ce GETG admet le T-invariant suivant :

Transition	x ₁	x ₂	x ₃	x ₄	x ₅	x ₆	x ₇	x ₈	x ₉	x ₁₀	x ₁₁	x ₁₂	x ₁₃	x ₁₄
Θ	3	3	3	3	3	3	1	1	1	1	2	2	2	2

Les temps de cycle des circuits de commande relatifs à chaque machine sont de 7 u.t pour la machine 1, 11 u.t pour la machine 2 et 6 u.t pour la machine 3. Comme la machine la plus chargée est la machine 2, nous choisissons d'obtenir un temps de cycle égal à 11 u.t. On suppose que dans la configuration initiale, q_1 correspond

au nombre de palettes réservées aux produits A et q_2 palettes réservées aux produits B. Quelles que soit l'allocation des palettes, on ne pourra donc jamais dépasser la fréquence de machines menantes, c'est-à-dire $3/5$ produit A et $2/5$ de produits B en moyenne toutes les 11 u.t. L'objectif consiste à déterminer les nombres q_1 et q_2 pour atteindre le temps de cycle de 11 u.t de la machine menante M_2 .

On remarque que le marquage des places vérifie la condition de linéarisation (4.2). En utilisant le changement de variable (4.1), on obtient un modèle en fonction de l'opérateur γ et δ dans le dioïde $\mathcal{O}_{\min}[[\delta]]$.

$$\left\{ \begin{array}{l} C_1(\delta) = \gamma^{\lfloor \frac{q_1}{3} \rfloor} C_6(\delta) \oplus \gamma^{\lfloor \frac{3}{3} \rfloor} C_7(\delta), \\ C_2(\delta) = \delta^3 C_1(\delta), \\ C_3(\delta) = \gamma^3 C_9(\delta) \oplus C_2(\delta), \\ C_4(\delta) = \delta^9 C_3(\delta), \\ C_5(\delta) = \gamma^{\lfloor \frac{q_1}{3} \rfloor} C_6(\delta) \oplus C_4(\delta), \\ C_6(\delta) = \delta^6 C_5(\delta), \\ C_7(\delta) = C_{14}(\delta), \\ C_8(\delta) = C_2(\delta), \\ C_9(\delta) = C_{12}(\delta), \\ C_{10}(\delta) = C_4(\delta), \\ C_{11}(\delta) = C_{10}(\delta) \oplus \gamma^{\lfloor \frac{q_2}{2} \rfloor} C_{14}(\delta), \\ C_{12}(\delta) = \delta^2 C_{11}(\delta), \\ C_{13}(\delta) = C_8(\delta) \oplus C_{12}(\delta), \\ C_{14}(\delta) = \delta^4 C_{13}(\delta). \end{array} \right.$$

Ce système d'équations modélise l'évolution dynamique du GET ordinaire représenté par la figure 4.10.

$$TC = \max\left(\frac{11}{1}, \frac{7}{1}, \frac{6}{1}, \frac{6}{q'_2}, \frac{18}{q'_1}, \frac{24}{q'_1 + q'_2 + 1}\right).$$

Ces contraintes se lisent en termes de circuits du graphe. $\frac{6}{q'_2}$ correspond au circuit $(c_{11}, c_{12}, c_{13}, c_{14})$, $\frac{18}{q'_1}$ au circuit $(c_1, c_2, c_3, c_4, c_5, c_6)$. Pour saturer la machine menante, il faut avoir un temps de cycle égal à 11 u.t. Par conséquent, on obtient :

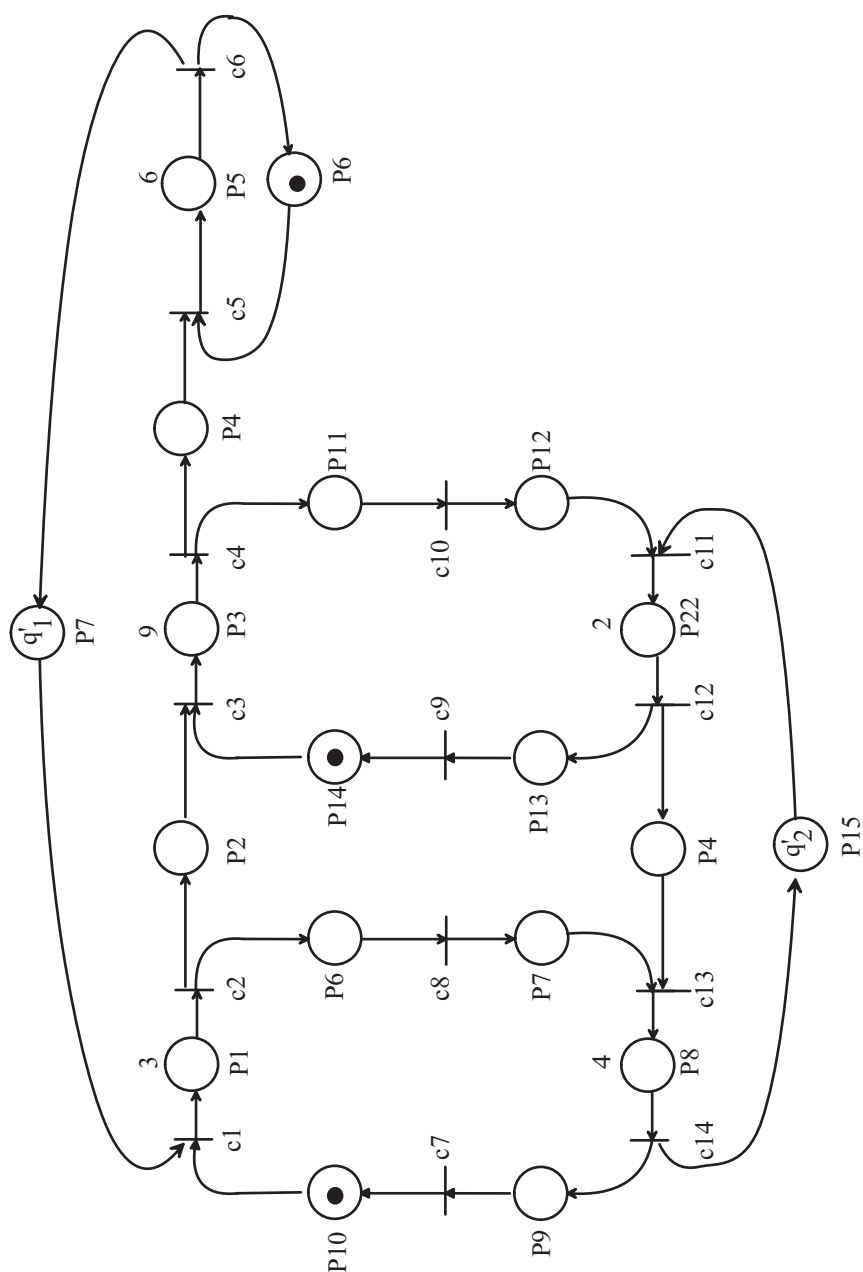


FIGURE 4.10 – Atelier flexible linéarisé.

$$q'_1 \in N - \{0, 1\}, \quad q'_2 \in N^*$$

En utilisant la condition de linéarisation 4.2, on trouve que le nombre de palettes à utiliser est donné par :

$$q_1 \in N - \{0, 1, 2\}, \quad q'_2 \in N - \{0, 1\}$$

Pour trouver le nombre de palette minimum, nous utilisons la relation (4.8), pour calculer la quantité $\mathcal{K}.M_0$ pour chaque valeur possible de q_1 et q_2 . On remarque bien que pour la valeur de $q_1 = 3$ et $q_2 = 2$, on a la valeur minimale de la quantité $\mathcal{K}.M_0$ correspondante au coût des ressources à utiliser et qui assure un temps de cycle de 11 u.t

4.4 Conclusion

Dans ce chapitre nous nous sommes intéressés à l'allocation de ressources dans le cas des systèmes modélisés par des GETG. le problème traité consiste à déterminer les ressources à utiliser dans le but d'atteindre un temps de cycle souhaité. Pour atteindre cet objectif, en utilisant une méthode de linéarisation, des équations récurrentes linéaires ont été déterminées sous certaines conditions sur le marquage initial. Ces équations linéaires permettent de d'établir le graphe d'événements temporisé équivalent. Le marquage de ce dernier est déterminé en utilisant la théorie spectrale dans les dioïdes. Par la suite, en exploitant la condition de linéarisation, nous déterminons le marquage saturant le "processus goulot" et qui minimise une somme pondérée des marquages des places du GETG.

Contrairement aux méthodes existantes, cette méthode se caractérise par son efficacité et sa simplicité, par rapport aux différentes heuristiques et méthodes développés, dans le cas où les contraintes imposées, condition de linéarisation (4.2) et le temps de cycle atteignable (4.35), sont satisfaites.

Conclusion et Perspectives

Les travaux présentés dans ce mémoire ont pour sujet principal les Systèmes à Événements Discrets qui peuvent être modélisés par les Graphes d'Événements Temporisés Généralisés. Ces graphes, qui sont une sous classe des Réseaux de Petri, permettent de modéliser, par exemple, de façon simple des opérations d'assemblage et de désassemblage de produits, présentes dans certains systèmes de production. Il a été montré, contrairement aux Graphes d'Événements Temporisés ordinaires, que ces graphes n'admettent pas une représentation linéaire dans l'algèbre $(\min, +)$. Ceci est dû à la présence des poids sur les arcs qui engendre une non linéarité dans l'évolution du comportement dynamique de ces modèles. Cette non linéarité empêche l'utilisation des résultats développés dans le cadre de l'analyse des systèmes linéaires dans l'algèbre $(\min, +)$.

Dans cette thèse, nous avons abordé deux problématiques associées aux systèmes modélisables par des GETG : évaluation de performances et allocation de ressources.

L'évaluation de performances fait l'objet du troisième chapitre. Dans ce dernier, nous avons proposé une nouvelle méthode de linéarisation du modèle mathématique régissant l'évolution dynamique d'un GETG, dans le but d'obtenir un modèle $(\min, +)$ linéaire. Ce dernier permet de déduire certains indicateurs de performances, comme le temps de cycle, en utilisant la théorie spectrale développée dans le cadre de l'analyse de performances des systèmes linéaires dans les dioïdes [Gaubert 1992]. Cette méthode a pour principe de synchroniser l'évolution de chaque compteur du graphe avec une entrée impulsionnelle. Cette dernière, de par son amplitude qui tend vers l'infini, permet de linéariser toute équation récurrente associée à une transition du graphe.

Les résultats de ce chapitre ouvrent la voie à deux pistes de recherche :

- La première porte sur la recherche des paramètres de performances en fonction des paramètres temporels du modèle, tout en développant des solutions sous forme de formules mathématiques applicables directement, sans conditions restrictives, sur le GETG initial.
- La deuxième piste consiste à rechercher la fonction de transfert d'un GETG dans le cas d'un graphe non fortement connexe en ajoutant des entrées impulsionnelles au niveau de chaque transition du graphe. Sachant que la relation entrées-sorties de ces graphes est non linéaire, à cause de la présence des parties entières qui sont utilisées pour préserver l'aspect discret de l'état du système. Cette fonction de transfert pourrait être utilisée pour faire, par exemple, de la commande en boucle fermée tout en développant un correcteur pour atteindre des performances souhaitées.

Le problème d'allocation de ressources des GETG est abordé dans le quatrième chapitre. Il consiste à déterminer une répartition minimale des ressources en vue d'atteindre des performances souhaitées. Pour atteindre cet objectif, nous avons procédé par la recherche des équations récurrentes linéaires équivalentes, en s'inspirant des travaux déjà existants. Ces équations linéaires permettent de déduire le GET équivalent. Le marquage recherché se calcule en utilisant la théorie spectrale et les performances souhaitées. Par la suite, en exploitant la condition de linéarisation, nous déterminons le marquage saturant le "processus goulot" et qui minimise une somme pondérée des marquages des places du GETG.

Comme amélioration à ce travail, des travaux pourraient être développés pour la généralisation de cette méthode à des cas où $TC_a \neq \frac{T(c_g)}{n}$. Pour ce faire, nous procéderions par ajout et retrait de jetons dans le graphe dans le but d'obtenir un TC_a cerné entre deux valeurs : une valeur min obtenue après avoir ajouté des jetons dans le graphe, et une valeur max obtenu après avoir enlevé des jetons dans les places où la condition de linéarisation n'est pas satisfaite. C'est deux valeurs de temps de cycle permettraient d'encadrer le marquage optimal recherché entre deux vecteurs marquages.

Publications Personnelles

Dans cette partie, on expose la liste des articles publiés durant cette thèse.

1. Revues internationales avec comité de lecture

- **A. BENFEKIR**, S. HAMACI, J.-L. BOIMOND and K. LABADI " Performance evaluation of nonlinear weighted T-system" International Journal of Systems Science. Taylor and Francis Publishers. Vol.44, Issue 10, p.p 1948-1955, 2013.
- **A. BENFKIR**, S. HAMACI and A-M. DARCHERIF, "Allocating Resources of Weighted T-Systems for Adaptive Behavior" International Journal of Operational Research, Volume 14 - Issue 1, pp. 31-49, 2012.

2. Conférences internationales avec comité de lecture

- **A. BENFEKIR**, S. HAMACI, A-M. DARCHERIF, L. LAVAL and M. BOUHAMIDA "On the nonlinear dynamic behavior of relaxed discrete-event systems in idempotent semi rings" IEEE conference 18th International Conference on Methods and Models in Automation and Robotics (MMAR), 26-29 August, 2013, Poland.
- S. HAMACI, **A. BENFEKIR**, J-L. BOIMOND, "Dioid approach for Performance Evaluation of Weighted T-System", 16th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), ISBN: 978-1-4577-0018-7, pages 1-8. September 5-9, Toulouse, France, 2011.
- S. HAMACI, **A. BENFKIR**, A-M. DARCHERIF "Dynamic Behavior of TEGM for desired cycle time" XVII International Conference on Systems Science, 14-16 September, 2010, Wroclaw, Poland.

3. Communications nationales

- **A. BENFEKIR**, S. HAMACI, L. LAVAL et M. BOUHAMIDA "Sur le comportement dynamique des SED non linéaires dans des demi-anneaux idempotents, 16ème Rencontre Non Linéaire 2013, Paris.

Annexes

A.1 Transformée en γ

La transformée en γ joue un rôle analogue à la transformée en z dans la théorie des systèmes linéaires continus en temps discret. Elle transforme les produits de convolution en des produits de séries formelles. La sortie d'un GET mono-entrée/mono-sortie apparaît alors comme le produit d'une série de transfert par une série d'entrée. La structure algébrique à utiliser pour une telle représentation est en conséquence celle d'un dioïde de séries formelles.

A.1.1 Définition

Pour un dateur $\{d(k)\}_{k \in \mathbb{Z}}$ (cf. remarque 2.4), la transformée en γ , notée $D(\gamma)$ est définie comme la série formelle :

$$D(\gamma) = \bigoplus_{k \in \mathbb{Z}} d(k) \gamma^k.$$

Supposons deux dateurs reliés par l'égalité $x_1(k) = x_2(k - 1)$, ce qui correspond à deux transitions séparées par une place contenant un jeton. La transformée en γ de chacun des dateurs est :

$$\begin{aligned} X_1(\gamma) &= \bigoplus_{k \in \mathbb{Z}} x_1(k) \gamma^k \\ &= \bigoplus_{k \in \mathbb{Z}} x_2(k - 1) \gamma^k \\ &= \gamma^1 \bigoplus_{k \in \mathbb{Z}} x_2(k - 1) \gamma^{k-1} \\ &= \gamma^1 X_2(\gamma). \end{aligned}$$

Il apparaît que multiplier une série en γ par γ^1 revient à décaler la séquence de une unité. Il est donc possible d'interpréter l'opérateur γ comme opérateur de décalage événementiel, ce que l'on écrit parfois $\gamma x(k) = x(k - 1)$. Ceci est illustré par la figure A.1.

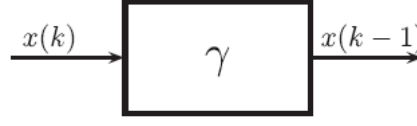


Figure A.1 – Opérateur de décalage événementiel.

Remarque 1. (Dioïde $\overline{\mathbb{Z}}_{min}[\gamma]$). Un ensemble de séries formelles à coefficients sur un dioïde complet peut également être muni d’une structure de dioïde complet dont la loi \oplus est la somme de séries formelles et la loi \otimes le produit de séries formelles. La transformée en γ des dateurs peut donc être considérée comme appartenant à un dioïde de séries formelles en γ .

Nous noterons $\overline{\mathbb{Z}}_{min}[\gamma]$ le dioïde complet des séries formelles en γ à coefficients dans $\overline{\mathbb{Z}}_{min}$ et exposants dans \mathbb{Z} . L’élément neutre de l’addition de $\overline{\mathbb{Z}}_{min}[\gamma]$ est la série $\epsilon(\gamma) = \bigoplus_{k \in \mathbb{Z}} \epsilon \gamma^k$ où $\epsilon = -\infty$ est l’élément neutre de l’addition dans $\overline{\mathbb{Z}}_{min}$ et l’élément neutre de la multiplication est la série $e(\gamma) = e\gamma^0$ (où $e = 0$ est l’élément neutre de la multiplication de $\overline{\mathbb{Z}}_{min}$).

A.1.2 Matrice de transfert

Pour un GET, il était toujours possible d’obtenir une représentation d’état canonique (2.8). Considérons la transformée en γ des dateurs $u(\cdot)$, $x(\cdot)$ et $y(\cdot)$ de l’équation (2.8), on a

$$X(\gamma) = \gamma A X(\gamma) \oplus B U(\gamma).$$

Il s’agit d’une équation implicite. Le résultat donne la plus petite solution

$$X(\gamma) = (\gamma A)^* B U(\gamma).$$

En reportant cette solution dans l’expression de la sortie Y , on obtient

$$Y(\gamma) = C(\gamma A)^* B U(\gamma) = H(\gamma) U(\gamma), \text{ avec } H(\gamma) = C(\gamma A)^* B.$$

Cette équation exprime le comportement entrée-sortie du système. La série formelle :

$$H(\gamma) = C(\gamma A)^* B \in (\overline{\mathbb{Z}}_{min})^{q \times p} \quad (\text{A.1})$$

est appelée série (ou matrice) de transfert du système.

Exemple A.1. Pour le modèle GET d'une machine présenté dans la figure A.2, on obtient la représentation par des séries formelles en γ suivante :

$$\begin{cases} X(\gamma) &= \begin{pmatrix} \varepsilon & \gamma^2 \\ 3 & \varepsilon \end{pmatrix} X(\gamma) \oplus \begin{pmatrix} e \\ \varepsilon \end{pmatrix} U(\gamma) \\ Y(\gamma) &= \begin{pmatrix} \varepsilon & e \end{pmatrix} X(\gamma) \end{cases}$$

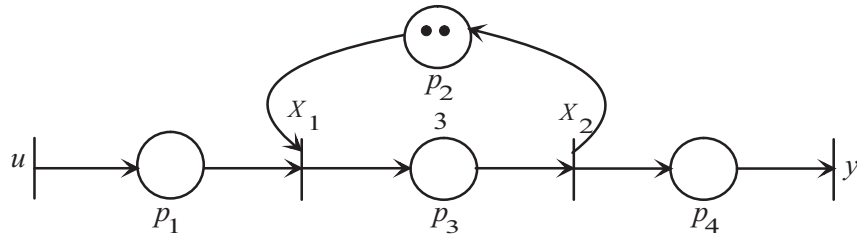


Figure A.2 – Modèle GET d'une machine.

Le calcul de la matrice de transfert $H(\gamma) = C(\gamma A)^* B$ donne $H(\gamma) = (3\gamma^2)^* 3$.

Pour le système décrit dans l'exemple précédent, nous considérons le signal d'entrée $U(\gamma) = e = e\gamma^0$, la sortie résultante vaut $Y(\gamma) = H(\gamma)U(\gamma) = (3\gamma^2)^* 3 \otimes e = (3\gamma^2)^* 3 = H(\gamma)$. Autrement dit, pour $U(\gamma) = e$ (entrée impulsionnelle), la sortie $Y(\gamma)$ coïncide avec la réponse impulsionnelle du système.

A.2 Transformée en δ

Dans le domaine temporel la situation est complètement duale, puisque une variable compteur $\{c(t)\}_{t \in \mathbb{Z}}$ est une application croissante : $\mathbb{Z} \rightarrow \mathbb{Z}_{min}$. Si on note δ l'opérateur de décalage, on a

$$C(\delta) = \bigoplus_{t \in \mathbb{Z}} c(t) \delta^t.$$

De même que précédemment, la propriété de monotonie (on se souviendra que l'ordre dans le dioïde $\overline{\mathbb{Z}_{min}}$ est l'inverse de l'ordre naturel, i.e., $a \succeq b, a \oplus b = a \Leftrightarrow a \leq b$) d'une trajectoire de compteur $c(t)$ se traduit par

$$\forall t \in \mathbb{Z}, \{c(t+1) \preceq c(t)\} \Leftrightarrow \{c(t) = c(t) \oplus c(t+1)\},$$

En passant à la transformée en δ , on obtient :

$$\{C(\delta) = C(\delta) \oplus \gamma^{-1}C(\delta)\} \Leftrightarrow \{C(\delta) = (\gamma^{-1})^*C(\delta)\}.$$

Exemple A.2. Pour l'exemple de la figure (A.2), on obtient la représentation par des séries formelles en δ suivante :

$$\begin{cases} X(\delta) &= \begin{pmatrix} \varepsilon & 2 \\ \delta^3 & \varepsilon \end{pmatrix} X(\delta) \oplus \begin{pmatrix} e \\ \varepsilon \end{pmatrix} U(\delta) \\ Y(\delta) &= \begin{pmatrix} \varepsilon & e \end{pmatrix} X(\delta) \end{cases}$$

Le calcul de la matrice de transfert $H(\delta) = C(\delta A)^*B$ donne $H(\delta) = (2\delta^3)^*\delta^3$.

Bibliographie

- [Baccelli 1992] F. Baccelli, G. Cohen, G.-J. Olsder et J.-P. Quadrat. *Synchronization and linearity : An Algebra for Discrete Event Systems*. Wiley and Sons, 1992.
- [Balbo 1998] G. Balbo et M. Silva. Performance models for discrete event systems with synchronizations. *MATCH Advanced Schools*, Editorial Kronos, Zaragoza, Espagne, 1998.
- [Baynat 2000] B. Baynat. *Théorie des files d’attentes : des chaines de markov aux réseaux à forme produit*. Ed Hermès, 2000.
- [Benfekir 2012] A. Benfekir, S. Hamaci et A.-M. Darcherif. *Allocating Resources of Weighted T-System for Adaptive Behaviour*. *International Journal of Operational Research*, vol. 14, no. 01, pages 31–49, 2012.
- [Benfekir 2013a] A. Benfekir, S. Hamaci, J.-L. Boimond et K. Labadi. *Performance Evaluation of Nonlinear Weighted T-System*. *International Journal of Systems Science*, vol. 44, no. 10, pages 1948–1955, 2013.
- [Benfekir 2013b] A. Benfekir, S. Hamaci, A.-M. Darcherif, L. Laval et M. Bouhamida. *On the Nonlinear Dynamic Behavior of Unrelaxed Timed Petri Nets in Idempotent Semirings*. *IEEE Conference, 18th International conference on Methods and Models in Automation and Robotics*, Midzyzdroje, Poland, vol. xx, pages xx–xx, 2013.
- [Chao 1993] D.T. Chao, M. Zhou et D.T. Wang. *Multiple Weighted Marked Graphs*. In *IFAC 12th Triennial World Congress*, pages 371–374, Sydney, Australie, 1993.
- [Chretienne 1983] P. Chretienne. *Les réseaux de Petri temporisés*. Thèse d’état, Université de Paris VI, 1983.
- [Cohen 1983] G. Cohen, D. Dubois, J.-P. Quadrat et M. Viot. *Analyse du comportement périodique des systèmes de production par la théorie des dioïdes*. Rapport de recherche 191, INRIA, Le Chesnay, France, 1983.
- [Cohen 1985] G. Cohen, D. Dubois, J.-P. Quadrat et M. Viot. *A linear system theoretic view of discrete event processes and its use for performance evaluation in manufacturing*. *IEEE Trans. on Automatic Control*, vol. 30, no. 3, pages 210–220, March 1985.

- [Cohen 1998a] G. Cohen, S. Gaubert et J.-P. Quadrat. *Timed-Event Graphs with Multipliers and Homogeneous Min-Plus Systems*. IEEE Transaction on Automatic Control, vol. 43, no. 9, pages 1296–1302, 1998.
- [Cohen 1998b] G. Cohen, S. Gaubert et J.-P. Quadrat. *Timed-Event Graphs with Multipliers and Homogenous Min-Plus Systems*. IEEE Trans. on Automatic Control, vol. 43, no. 9, pages 1296–1302, 1998.
- [Compos 1991] J. Compos, G. Chiola et M. Silva. *Properties and Performance Bounds for Closed Free Choice Synchronized Monoclass Queueing Networks*. IEEE Trans. on Automatic Control, vol. 36, no. 12, pages 1368–1382, 1991.
- [David 1992] R. David et H. Alla. *Du grafctet au réseaux de petri*. Editions Hermès, Paris, 1992.
- [Gaubert 1992] S. Gaubert. *Théorie des systèmes linéaires dans les dioïdes*. Thèse, Ecole des Mines de Paris, July 1992.
- [Gaubert 1995] S. Gaubert. *Resource Optimization and $(\min, +)$ Spectral Theory*. IEEE Trans. on Automatic Control, vol. 40, no. 11, pages 1931–1934, November 1995.
- [Gondran 1985] M. Gondran et M. Minoux. *Graphes et algorithmes*. Eyrolles, Paris, France, 1985.
- [Gunawardena 1998] J. Gunawardena, editeur. *Idempotency*. Publications of the Newton Institute. Cambridge University Press, 1998.
- [Hamaci 2004] S. Hamaci, J-L. Boimond, S. Lahaye et Mostefaoui.M. *Sur la linéarisation des graphes d'événements temporisés discrets avec multiplieurs dans l'algèbre $(\min, +)$* . Revue e-STA, Sciences et Technologies de l'Automatique, vol. 1, no. 3, 2004.
- [Hamaci 2005] S Hamaci, J-L Boimond et S Lahaye. *Performance Analysis of Timed Event Graphs with Multipliers Using $(\min, +)$ Algebra*. In 2st International Conference on Informatics in Control Automation and Robotics (ICINCO'05), volume 4, pages 16–21, Barcelone, Espagne, 2005.
- [Hamaci 2010] S. Hamaci, A. Benfekir et A-M. Darcherif. *Dynamic Behavior of TEGM for Desired Cycle Time*. XVII International Conference on Systems Science , 14-16 September, 2010, Wroclaw, Poland, 2010.

- [Hillion 1989] H.P Hillion. *Modélisation et analyse des systèmes de production discrets par les réseaux de Petri temporisés*. Thèse, Université de Paris IV, France, 1989.
- [Lopez 2001] P. Lopez et F. Roubellat. *Ordonnancement de la Production*. Edition Hermès, 2001.
- [Munier 1993] A. Munier. *Régime asymptotique optimal d'un graphe d'événements temporisé généralisé : application à un problème d'assemblage*. APII, vol. 27, no. 5, pages 487–513, 1993.
- [Murata 1989] T. Murata. *Petri Nets : Properties, Analysis and Applications*. In IEEE Proceedings : Special issue on Discrete Event Systems, volume 77, pages 541–581, 1989.
- [Nakamura 1999] M. Nakamura et M. Silva. *Cycle Time Computation in Deterministically Timed Weighted Marked Graphs*. In IEEE-ETFA, pages 1037–1046, 1999.
- [Petri 1962] C.-A. Petri. *Kommunikation mit Automaten*. Thèse, Institut für Instrumentelle Mathematik, Bonn, Germany, 1962.
- [Proth 1995] J.M. Proth et X. Xie. *Les réseaux de petri pour la conception et la gestion des systèmes de production*. Edition MASSON, 1995.
- [Ramchandani 1974] C. Ramchandani. *Analysis of Asynchronous Concurrent Systems by Timed Petri Nets*. Thèse d'état, Université de Paris VI, 1974.
- [Sauer 2003] N. Sauer. *Marking Optimization of Weighted Marked Graphs*. Journal of Discrete Event Dynamic Systems, vol. 13, pages 245–262, 2003.
- [Sauer 2004] N. Sauer. *Contribution à l'évaluation et l'optimisation des systèmes de production*. Rapport d'habilitation à diriger des recherches, Ecole des mines de Nantes, 2004.
- [Sifakis 1980] J. Sifakis. *Performance Evaluation of Systems using Petri Nets*. Net Theory and Application lecture notes in computer science, spring-verlag,, 1980.
- [Teruel 1992] E. Teruel, P. Chrzastowski-Wachtel, J.M. Colom et M. Silva. *On Weighted T-Systems*. In Advances in Petri nets, Lecture Notes in Computer Science, volume 616, pages 348–367, 1992.
- [Trouillet 2001] B. Trouillet, A. Benasser et J.-C. Gentina. *Sur la modélisation du comportement dynamique des graphes d'événements pondérés*. In G. Jua-

nole et R. Valette, editeurs, MSR'2001, pages 447–462, Toulouse, France, October 2001. Hermès.

[Trouillet 2003] B. Trouillet. *Contribution à la commande des systèmes linéaires dans les diodes*. Thèse, Ecole Centrale de Lille, France, 2003.

Contribution à l'analyse de performances des SED non linéaires dans l'algèbre $(\min, +)$

Abderrahim BENFEKIR

Résumé

Cette thèse s'inscrit dans le cadre de la théorie des systèmes linéaires dans les dioïdes. Cette théorie concerne la sous-classe des systèmes à événements discrets modélisables par les Graphes d'Événements Temporisés (GET). La dynamique de ces graphes peut être représentée par des équations récurrentes linéaires sur des structures algébriques particulières telles que l'algèbre $(\max, +)$ ou l'algèbre $(\min, +)$. Ce mémoire est consacré à l'analyse de performances des systèmes dynamiques qui peuvent être modélisés graphiquement par des Graphes d'Événements Temporisés Généralisés (GETG). Ces derniers, contrairement au GET, n'admettent pas une représentation linéaire dans l'algèbre $(\min, +)$. Pour pallier à ce problème de non linéarité, nous avons utilisé une approche de modélisation définie sur un dioïde d'opérateurs muni de deux lois internes : loi additive (\oplus) correspondant à l'opération (\min) , et loi multiplicative (\otimes) équivalente à la loi de composition usuelle. Le modèle d'état obtenu, est utilisé pour évaluer les performances des GETG. Pour cela, nous avons proposé une nouvelle méthode qui a pour but de linéariser le modèle mathématique régissant l'évolution dynamique du modèle graphique, dans le but d'obtenir un modèle $(\min, +)$ linéaire. La deuxième partie de cette thèse est consacrée au problème qui consiste à déterminer les ressources à utiliser dans une ligne de production, en vue d'atteindre des performances souhaitée. Ceci est équivalent à déterminer le marquage initial de la partie commande du GETG.

Mots-clés : Réseaux de Petri, graphes d'événements temporisés généralisés, graphes d'événements temporisés, dioïde, algèbre $(\min, +)$, linéarisation, temps de cycle, optimisation.

Abstract

This thesis is part of the theory of linear systems over dioids. This theory concerns the subclass of discrete event dynamic systems modeled by Timed Event Graphs (TEG). The dynamics of these graphs can be represented by linear recurrence equations over specific algebraic structures such as $(\max, +)$ algebra or $(\min, +)$ algebra. This report is devoted to the performance analysis of dynamic systems which can be represented graphically by Generalized Timed Event Graphs (GTEG). These type of graphs, unlike TEG, do not admit a linear representation in $(\min, +)$ algebra. To mitigate the problem of nonlinearity, we used a modeling approach defined on a dioid operators endowed with \min as (\oplus) usual composition as (\otimes) . The obtained state model is used to evaluate the performance of GTEG. For this, we proposed a new method to linearize the mathematical model governing the dynamic evolution of the graphical model in order to obtain a linear model in $(\min, +)$ algebra. The second part of this work is devoted to the problem of determining the resources to use in a production line, in order to achieve desired performance. These is equivalent to determining the initial marking of the control part of the GTEG.

Keywords: Petri Nets, generalized timed event graphs, timed event graphs, dioid, $(\min, +)$ algebra, linearization, cycle time, optimization.