

Modélisation de Réseaux de Graphes d'Événements Temporisés avec Conflits dans l'algèbre (Max,+)

Boussad ADDAD, Saïd AMARI, Jean-Jacques LESAGE

Laboratoire Universitaire de Recherche en Production Automatisée,
LURPA – Ecole Normale Supérieure de Cachan
61, avenue du Président Wilson, 94235 Cachan, France.

boussad.addad@lurpa.ens.cachan.fr, said.amari@lurpa.ens-cachan.fr, jean-jacques.lesage@lurpa.ens-cachan.fr

Résumé— Les Graphes d'Événements Temporisés (GET) sont une classe de Réseaux de Petri (RdP) auxquels on peut aisément associer une représentation d'état (Max,+) linéaire qui facilite leur étude. Cependant, les GET ne permettent pas de modéliser des systèmes avec partage de ressources et/ou gestion de conflits. Pour y remédier, nous proposons une modélisation (Max,+) pour les Réseaux de Graphes d'Événements Temporisés avec Conflits (RGETC). Dans un premier temps, nous développons des équations (Max,+) récurrentes régissant l'évolution des RGETC quelque soit la règle d'arbitrage des conflits appliquée. Nous considérons ensuite des règles particulières et aboutissons à des équations semblables à la forme d'état (Max,+) classique.

Mots-clés— Graphe d'événements temporisé (GET), algèbre (Max,+), conflits, règle d'arbitrage.

I. INTRODUCTION

Les réseaux de Petri (RdP) sont un outil puissant de modélisation des systèmes à événements discrets. Ils sont souvent utilisés pour représenter des phénomènes de synchronisation, de parallélisme et de concurrence [1]. Leur application s'étend à de nombreux domaines : production manufacturière, réseaux de communication, transport ...etc. De multiples classes de RdP, aux sémantiques puissantes (colorés, temporisés, ...), ont été élaborées. Force est de constater cependant que les possibilités d'analyse des RdP sont le plus souvent inversement proportionnelles à leurs capacités sémantiques. C'est la raison pour laquelle les graphes d'événements temporisés (GET), sont l'une des classes de RdP les plus prisées pour leurs capacités d'analyse. En effet, la dynamique des GET peut être exprimée à l'aide d'équations linéaires dans l'algèbre (Max,+), ce qui en permet une analyse très poussée, similaire à celle des systèmes linéaires classiques [2]. En revanche, toute place d'un GET ne peut être reliée au maximum qu'à une transition en amont et une transition en aval. Les GET ne permettent donc pas de représenter les phénomènes de concurrence ou de choix multiples qui caractérisent les systèmes avec partage de ressources.

II. MOTIVATIONS ET CONTRIBUTIONS

Dans [3], nous avons proposé une approche permettant d'évaluer analytiquement les temps de réponse des systèmes de commande en réseau. Pour ce faire, nous avons utilisé les GET et leur modélisation (Max,+). Bien évidemment, un réseau de communication est un système possédant de multiples ressources partagées (processeurs, routeurs, commutateurs ...) et l'utilisation des seuls GET n'est pertinente que dans des cas particuliers pour lesquels les conflits (par ex. les accès aux

commutateurs) peuvent être résolus au préalable et les délais d'attente pour l'accès aux ressources peuvent être calculés a priori puis intégrés dans un GET. Pour une étude analytique des systèmes de commande plus complexes, une modélisation des conflits d'accès aux ressources partagées est nécessaire, ce qui rend caduque la modélisation à l'aide des seuls GET.

Dans la littérature, différentes méthodes ont été développées pour répondre à ce type de problème. Dans [4], [5], les auteurs traitent des systèmes avec allocation périodique des ressources partagées. Ils proposent d'imposer cette périodicité en transformant le RdP initial avec conflits en un GET, la représentation d'état de ce dernier étant aisément calculée. Dans [6], une modélisation algébrique, basée sur des automates de type tas de pièces, est obtenue pour les RdP saufs. Pour les RdP à choix libre (CL), une étude analytique assez complète est proposée dans [7]. Pour les processus pouvant commuter entre différents modes de fonctionnement, une approche algébrique à base de systèmes à commutations (Max, +) linéaires est proposée dans [8]. Nous pouvons citer également [9] où des équations (Max, +) linéaires locales sont écrites pour modéliser chaque utilisateur des ressources partagées sans prise en compte des conflits. Une contrainte sur le marquage de la place représentant la ressource est introduite pour ne considérer que les évolutions admissibles dans le système global. Dans [10], une méthode introduisant le concept de tir virtuel d'une transition, est appliquée pour un système de transport. Plus récemment, une approche à base de dioïdes d'intervalles est proposée dans [11]. Elle permet de modéliser (les comportements aux limites seulement) des systèmes de production avec conflits.

Comme on peut le constater, les travaux précédents imposent souvent des contraintes assez fortes qui limitent leur utilisation : périodicité d'allocation des ressources, RdP saufs, sans transitions d'entrée, à CL (donc une transition conflictuelle ne peut pas avoir plus d'une place en amont). Pour la modélisation des réseaux de communication, ces hypothèses sont très contraignantes puisque par exemple : i) l'allocation des ressources n'est le plus souvent pas périodique ni à choix libre mais imposée par d'autres règles comme PAPS (premier arrivé premier servi), ii) le marquage des places du réseau ne peut souvent pas être borné à 1 (lorsqu'il représente l'utilisation d'un buffer de réception des messages émis par un contrôleur par exemple).

L'objet de ce papier est de proposer une modélisation algébrique d'une classe étendue de RdP : les Réseaux de Graphes d'Événements Temporisés avec Conflits ou RGETC. Ceux-ci offrent des capacités sémantiques plus puissantes que

les GET et permettent de relaxer plusieurs des contraintes précédemment exposées. En effet, nous supposons que :

- les GET qui composent le RGETC ne sont pas nécessairement saufs (ils contiennent notamment des transitions d'entrée) ;

- les transitions conflictuelles peuvent avoir plus d'une place en amont ;

- l'allocation des ressources n'est pas obligatoirement périodique.

Cependant, dans le cadre de nos travaux nous faisons l'hypothèse qu'une ressource partagée ne peut pas être utilisée par plusieurs utilisateurs simultanément. Comme nous le verrons par la suite, ceci va se traduire de manière formelle par une contrainte sur le marquage du RGETC.

La suite de ce papier est organisée comme suit : la section III rappelle quelques notions sur les GET et leur modélisation (Max,+) linéaire. La section IV est consacrée à l'étude des RGETC. La section IV.A introduit quelques définitions et notations sur les RGETC. La section IV.B présente la modélisation (Max,+) d'un RGETC et les équations récurrentes régissant son évolution. Dans la section V, des équations plus explicites sont données pour des RGETC dont les conflits sont arbitrés par des règles bien connues. La section VI conclut ce papier avec quelques perspectives.

III. RAPPELS : EQUATIONS (MAX,+) LINEAIRES DES GET

Dans cette section, nous allons rappeler la représentation (Max,+) linéaire des GET. Celle-ci nous sera très utile pour la suite. Pour ce faire, nous considérons l'exemple ci-après :

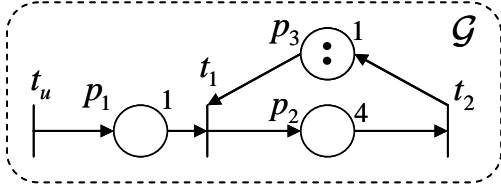


Fig. 1. Un exemple de GET.

Nous supposons dans toute l'étude que seules les places sont temporisées. Soit $x_i(k)$ (resp. $u(k)$) la date de tir de la transition t_i (resp. t_u) pour la $k^{\text{ème}}$ fois. Si le tir des transitions se fait à vitesse maximale (au plus tôt), on peut écrire les équations suivantes :

$$\begin{cases} x_1(k) = \max(u(k) + 1, x_2(k-2) + 1) \\ x_2(k) = x_1(k) + 4 \end{cases} \quad (1)$$

En introduisant une nouvelle variable x_3 avec $x_3(k) = x_2(k-1)$, nous obtenons un système d'équations étendu :

$$\begin{cases} x_1(k) = \max(u(k) + 1, x_3(k-1) + 1) \\ x_2(k) = x_1(k) + 4 \\ x_3(k) = x_2(k-1) \end{cases} \quad (2)$$

Les équations ci-dessus peuvent être réécrites en utilisant les opérateurs de l'algèbre (Max,+), ou (\oplus, \otimes) définis sur l'ensemble $\mathbb{R}_{\max} = \mathbb{R} \cup \{-\infty\}$ et dont les éléments neutres sont respectivement $\varepsilon = -\infty$ et $e = 0$:

$$\begin{cases} x_1(k) = 1 \otimes u(k) \oplus 1 \otimes x_3(k-1) \\ x_2(k) = 4 \otimes x_1(k) \\ x_3(k) = x_2(k-1) \end{cases} \quad (3)$$

qui s'écrivent sous la forme matricielle suivante :

$$X(k) = \hat{A} \otimes X(k) \oplus \hat{A}' \otimes X(k-1) \oplus \hat{B} \otimes U(k) \quad (4)$$

$$\text{où : } X = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}, U = u, \hat{A}' = \begin{pmatrix} \varepsilon & \varepsilon & 1 \\ \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & e & \varepsilon \end{pmatrix}, \hat{A} = \begin{pmatrix} \varepsilon & \varepsilon & \varepsilon \\ 4 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon \end{pmatrix}, \hat{B} = \begin{pmatrix} 1 \\ \varepsilon \end{pmatrix}.$$

L'équation (4) peut se mettre sous une forme explicite grâce au théorème suivant :

Théorème [2] : soit l'équation $X = A \otimes X \oplus B$ et $A \in \mathbb{R}_{\max}^{n \times n}$, $B \in \mathbb{R}_{\max}^{n \times m}$. La solution minimale de cette équation est donnée par : $X = A^* \otimes B$ où la matrice $A^* = \bigoplus_{i \in \mathbb{N}} A^i$ est l'étoile de Kleene de A .

En appliquant le théorème précédent à (4), nous aboutissons à la forme explicite :

$$X(k) = A \otimes X(k-1) \oplus B \otimes U(k) \quad (5)$$

$$\text{où : } \hat{A}^* = \begin{pmatrix} e & \varepsilon & \varepsilon \\ 4 & e & \varepsilon \\ \varepsilon & \varepsilon & e \end{pmatrix}, A = \hat{A}^* \cdot \hat{A}' = \begin{pmatrix} \varepsilon & \varepsilon & 1 \\ \varepsilon & \varepsilon & 5 \\ \varepsilon & e & \varepsilon \end{pmatrix} \text{ et } B = \hat{A}^* \cdot \hat{B} = \begin{pmatrix} 1 \\ 5 \\ \varepsilon \end{pmatrix}.$$

Remarque III. 1 : L'équation (4) et sa forme explicite (5) sont caractéristiques de tout GET. C'est une représentation d'état standard. Elle est similaire à la représentation d'état des systèmes linéaires classiques et elle est très utilisée pour les problèmes d'évaluation de performances temporelles ou de synthèse de commande [2], [12]. Dans la prochaine section, nous proposerons une méthode nous permettant d'obtenir des formes assez semblables pour la classe des RGETC.

IV. MODELISATION DES RGETC DANS L'ALGEBRE (MAX,+)

A. Définitions et notations

Un réseau de Graphes d'Événements Temporisés avec Conflits est un ensemble de GET (utilisateurs) qui partagent une ressource représentée par une *place de conflit* notée \tilde{p} et dont la temporisation est $\tilde{\tau}$, comme montré sur la figure 2.

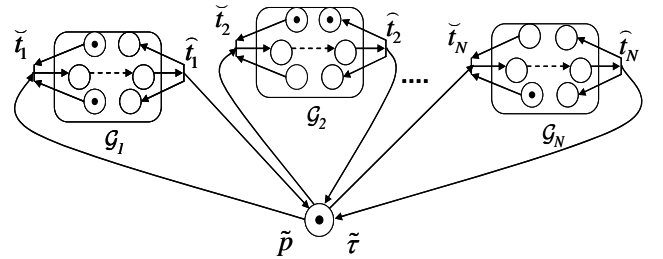


Fig. 2. Un RGETC avec une place de conflit \tilde{p} et N GET.

La figure 2 représente un réseau comprenant N graphes d'événements \mathcal{G}_i ($1 \leq i \leq N$) en concurrence pour consommer le jeton de la place \tilde{p} . Posons les notations suivantes :

- l'unique transition du GET \mathcal{G}_i qui est en aval de la place \tilde{p} est notée \tilde{t}_i et l'unique transition en amont est notée \hat{t}_i , comme représenté sur la figure 2 ;
- chaque GET \mathcal{G}_i est représenté par son équation d'état :

$$X_i(k) = \hat{A}_i \otimes X_i(k) \oplus \hat{A}'_i \otimes X_i(k-1) \oplus \hat{B}_i \otimes U_i(k) \quad (6)$$

où $X_i = (x_i^1 \ x_i^2 \ \dots \ x_i^{n_i})^t$ est le vecteur d'état de dimension $n_i \times 1$ et $U_i = (u_i^1 \ u_i^2 \ \dots \ u_i^{m_i})^t$ est le vecteur d'entrée de dimension $m_i \times 1$. Dans la suite de l'étude, la $i^{\text{ème}}$ ligne d'une matrice M est notée $M(i, :)$, $M(i, j)$ désigne l'élément d'intersection entre sa ligne i et sa colonne j et M puissance i au sens (Max,+) est : $M^{(i)} = M \otimes M^{(i-1)} = \underset{j=1}{\overset{i}{\otimes}} M$.

Comme nous l'avions annoncé dans la section II, la ressource ne peut être utilisée que par un seul utilisateur à la fois. Ceci explique la représentation sur la figure 2 d'un seul jeton dans la place \tilde{p} . Le marquage de cette place est borné à 1 : égal à zéro si la ressource est en cours d'utilisation et égal à 1 sinon.

B. Equations (Max,+) récurrentes des RGETC

Dans l'étude des GET, des dateurs sont en général associés aux transitions. Dans notre cas, nous associons également un dateur au jeton de \tilde{p} . La date de disponibilité, pour la $k^{\text{ème}}$ fois, du jeton de la place de conflit \tilde{p} est notée $\tilde{\psi}(k)$. Par abus de langage, nous parlerons de date de disponibilité du $k^{\text{ème}}$ jeton entrant dans la place \tilde{p} .

Supposons, sans perte de généralité (spg), que la première composante du vecteur X_i (donc x_i^1) est le dateur associé à la transition \tilde{t}_i . Si le $k^{\text{ème}}$ jeton entrant dans \tilde{p} est consommé par le GET \mathcal{G}_i et que ce dernier ait déjà consommé $(k_i - 1)$ jetons de \tilde{p} , nous avons :

$$x_i^1(k_i) = \hat{A}_i(1, :) \otimes X_i(k_i) \oplus \hat{A}_i'(1, :) \otimes X_i(k_i - 1) \oplus \hat{B}_i(1, :) \otimes U_i(k_i) \oplus \tilde{\psi}(k) \quad (7)$$

Comme les dates de tir des autres transitions de \mathcal{G}_i (x_i^j avec $j \geq 2$) ne dépendent pas directement du jeton de \tilde{p} (ces transitions n'étant pas en aval de \tilde{p}), les composantes x_i^j de $X_i(k_i)$ dans (6) restent inchangées pour $j \geq 2$, soit :

$$x_i^j(k_i) = \hat{A}_i(j, :) \otimes X_i(k_i) \oplus \hat{A}_i'(j, :) \otimes X_i(k_i - 1) \oplus \hat{B}_i(j, :) \otimes U_i(k_i) \quad (8)$$

En combinant (7) et (8), nous aboutissons à la forme matricielle :

$$X_i(k_i) = \hat{A}_i \otimes X_i(k_i) \oplus \hat{A}_i' \otimes X_i(k_i - 1) \oplus \hat{B}_i \otimes U_i(k_i) \oplus \hat{F}_i \otimes \Psi(k) \quad (9)$$

où toutes les matrices et vecteurs impliqués sont exactement les mêmes que ceux de (6). Les seuls nouveaux éléments sont $\Psi(k)$ et la matrice \hat{F}_i de dimension $n_i \times 1$. Nous pouvons vérifier aisément qu'ils sont donnés par : $\hat{F}_i = (e \ \varepsilon \ \dots \ \varepsilon)^t$ et $\Psi(k) = \tilde{\psi}(k)$. Toutes les composantes de \hat{F}_i sont nulles (égales à ε) sauf la première composante (égale à e).

En appliquant le théorème de la section III pour (9), nous aboutissons finalement à la solution :

$$X_i(k_i) = A_i \otimes X_i(k_i - 1) \oplus B_i \otimes U_i(k_i) \oplus F_i \otimes \Psi(k) \quad (10)$$

où $A_i = \hat{A}_i^* \otimes \hat{A}_i'$, $B_i = \hat{A}_i^* \otimes \hat{B}_i$ et $F_i = \hat{A}_i^* \otimes \hat{F}_i$.

Par ailleurs, sachant que le $k^{\text{ème}}$ jeton de la place \tilde{p} est consommé par le GET \mathcal{G}_i , le $(k+1)^{\text{ème}}$ jeton de \tilde{p} est généré

avec le tir, pour la $k_i^{\text{ème}}$ fois, de la transition \tilde{t}_i . Supposons (spg), que le dateur associé à \tilde{t}_i est la dernière composante du vecteur X_i (soit $x_i^{n_i}$). Nous pouvons alors écrire :

$$\tilde{\psi}(k+1) = \tilde{\tau} \otimes x_i^{n_i}(k_i) \quad (11)$$

Ce qui peut se traduire sous une forme matricielle par :

$$\Psi(k+1) = G_i \otimes X_i(k_i) \quad (12)$$

où $G_i = (\varepsilon \ \dots \ \varepsilon \ \tilde{\tau})$ est une matrice de dimension $1 \times n_i$ et dont les composantes sont toutes nulles sauf la dernière qui est égale à la temporisation de la place \tilde{p} , soit $\tilde{\tau}$.

En résumé, si le $k^{\text{ème}}$ jeton de \tilde{p} est consommé par le GET \mathcal{G}_i , les équations (max,+) récurrentes suivantes sont vérifiées pour $1 \leq i \leq N$ et $k = \sum_{i=1}^N (k_i - 1)$:

$$\begin{cases} X_i(k_i) = A_i \otimes X_i(k_i - 1) \oplus B_i \otimes U_i(k_i) \oplus F_i \otimes \Psi(k) \\ \Psi(k+1) = G_i \otimes X_i(k_i) \end{cases} \quad (13)$$

Comme on peut le remarquer, nous avons obtenu des équations (Max,+) récurrentes régissant l'évolution des RGETC indépendamment de toute politique d'arbitrage des conflits. Toutes les évolutions possibles du RGETC sont représentées à travers ces équations quelque soit la politique appliquée. Les matrices qui y sont impliquées sont calculées systématiquement suivant la méthode développée précédemment. Ces équations sont importantes et seront à la base de tous les résultats qui suivront.

Exemple IV.1 : pour illustrer la méthode précédente, nous considérons l'exemple de la figure 3 ci-dessous :

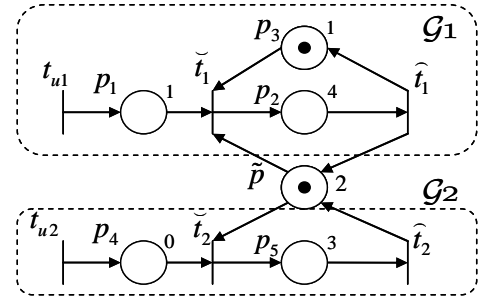


Fig. 3. Un RGETC avec deux GET et une place de conflit.

En associant les dateurs x_1^1 , x_1^2 , x_2^1 et x_2^2 aux transitions : \tilde{t}_1 , \tilde{t}_1 , \tilde{t}_2 et \tilde{t}_2 , les deux GET \mathcal{G}_1 et \mathcal{G}_2 sont représentés par :

$$\mathcal{G}_1: \begin{cases} x_1^1(k_1) = 1 \otimes u_1(k_1) \oplus 1 \otimes x_1^2(k_1 - 1) \\ x_1^2(k_1) = 4 \otimes x_1^1(k_1) \end{cases} \quad (14)$$

$$\mathcal{G}_2: \begin{cases} x_2^1(k_2) = e \otimes u_2(k_2) \\ x_2^2(k_2) = 3 \otimes x_2^1(k_2) \end{cases} \quad (15)$$

Si le $k^{\text{ème}}$ jeton de \tilde{p} est consommé par le GET \mathcal{G}_1 et que ce dernier a déjà consommé $(k_1 - 1)$ jetons de \tilde{p} , nous avons :

$$\begin{cases} x_1^1(k_1) = 1 \otimes u_1(k_1) \oplus 1 \otimes x_1^2(k_1 - 1) \oplus \tilde{\psi}(k) \\ x_1^2(k_1) = 4 \otimes x_1^1(k_1) \\ \tilde{\psi}(k+1) = 2 \otimes x_1^2(k_1) \end{cases} \quad (16)$$

Ces équations peuvent s'écrire aisément sous la forme (13).

Nous laissons le lecteur écrire les équations relatives au GET \mathcal{G}_2 et aboutir finalement à la forme (13) où :

$$A_1 = \begin{pmatrix} \varepsilon & 1 \\ \varepsilon & 5 \end{pmatrix}, \quad B_1 = \begin{pmatrix} 1 \\ 5 \end{pmatrix}, \quad A_2 = \begin{pmatrix} \varepsilon & \varepsilon \\ 3 & \varepsilon \end{pmatrix}, \quad B_2 = \begin{pmatrix} e \\ \varepsilon \end{pmatrix},$$

$$F_1 = (e \ 4)^t, \quad G_1 = (\varepsilon \ 2), \quad F_2 = (e \ \varepsilon)^t, \quad G_2 = (\varepsilon \ 2).$$

Remarques IV : i) La contrainte sur le marquage de la place de conflit ne s'est pas traduite explicitement dans la modélisation des RGETC mais seulement de manière implicite dans le raisonnement en parlant du « jeton de la place \tilde{p} » ii) dans la présentation des RGETC, nous n'avons envisagé qu'une seule place de conflit (ou ressource partagée). Les équations (13) peuvent être obtenues pour des RGETC avec plusieurs ressources de manière similaire. Dans ce cas, $\Psi(k)$ serait un vecteur et non un scalaire. Un exemple avec deux places de conflit est traité dans la section V.B.

V. CONSEQUENCES ET APPLICATIONS

Dans cette section, nous allons montrer comment utiliser les équations (13) pour une modélisation plus explicite d'un RGETC connaissant la politique appliquée pour l'arbitrage des conflits. Nous considérons deux politiques : la règle du Premier Arrivé Premier Servi (PAPS) et une règle d'allocation périodique de la ressource.

A. Règle du Premier Arrivé Premier Servi

Reprenons le RGETC de la figure 2. La règle PAPS est exprimée comme suit : la transition \tilde{t}_i qui demande le jeton de \tilde{p} en premier gagne ce jeton. On dit qu'une transition \tilde{t}_i demande le jeton de \tilde{p} pour la $k_i^{\text{ème}}$ fois, à la date $\sigma_i(k_i)$, si toutes les places en amont de \tilde{t}_i (exceptée \tilde{p}) ont un jeton disponible. Sur la figure. 3 par exemple, la transition \tilde{t}_1 demande le jeton pour la $k_1^{\text{ème}}$ fois à partir du moment où il y a un jeton disponible dans les places p_1 et p_3 , soit à la date : $\sigma_1(k_1) = 1 \otimes u_1(k_1) \oplus 1 \otimes x_1^2(k_1 - 1)$. Nous pouvons noter que $\sigma_1(k_1)$ a la même expression que la date de franchissement de la transition \tilde{t}_1 en l'absence de la place de conflit (voir équation (14)). Dans le cas général, nous avons :

$$\sigma_i(k_i) = A_i(1,:) \otimes X_i(k_i - 1) \oplus B_i(1,:) \otimes U_i(k_i) \quad (17)$$

D'après la définition de la règle PAPS, le $k^{\text{ème}}$ jeton de \tilde{p} est à consommer par la transition le demandant le plus tôt. En supposant que chaque transition \tilde{t}_i a déjà consommé $(k_i - 1)$ jetons de \tilde{p} , la transition \tilde{t}_j qui satisfait la condition $\sigma_j(k_j) = \min_{1 \leq i \leq N} \sigma_i(k_i)$, gagne le jeton. Naturellement, nous pouvons envisager une demande simultanée (donc conflit réel) du jeton par plusieurs transitions conflictuelles. Dans ce cas, pour obtenir un comportement complètement déterministe (qui est requis dans le cadre de cette étude), il faut privilégier une transition par rapport aux autres (seulement quand cette simultanéité survient). Pour ce faire, nous pouvons définir une fonction : $\arg \min_{1 \leq i \leq N} (\sigma_i(k_i))$ qui

renvoie l'indice j qui vérifie : $\sigma_j(k_j) = \min_{1 \leq i \leq N} \sigma_i(k_i)$ et dont la transition \tilde{t}_j est privilégiée en cas d'égalité. Ainsi, si le $k^{\text{ème}}$ jeton de \tilde{p} est gagné par la transition \tilde{t}_j (qui vérifie donc $j = \arg \min_{1 \leq i \leq N} (\sigma_i(k_i))$), nous avons les équations :

$$\begin{cases} X_j(k_j) = A_j \otimes X_j(k_j - 1) \oplus B_j \otimes U_j(k_j) \oplus F_j \otimes \Psi(k) \\ \Psi(k+1) = G_j \otimes X_j(k_j) \end{cases} \quad (18)$$

À présent, $X_j(k_j)$ est connu et nous pouvons calculer la nouvelle date de demande de \tilde{t}_j , soit : $\sigma_j(k_j + 1) = A_j(1,:) \otimes X_j(k_j) \oplus B_j(1,:) \otimes U_j(k_j + 1)$. Le même mécanisme peut être répété pour le $(k+1)^{\text{ème}}$ jeton et ainsi de suite. En résumé, à la $k^{\text{ème}}$ étape ($k^{\text{ème}}$ jeton de \tilde{p}), les équations suivantes sont vérifiées :

$$\begin{cases} j = \arg \min_{1 \leq i \leq N} (\sigma_i(k_i)) \\ X_j(k_j) = A_j \otimes X_j(k_j - 1) \oplus B_j \otimes U_j(k_j) \oplus F_j \otimes \Psi(k) \\ \Psi(k+1) = G_j \otimes X_j(k_j) \\ \sigma_j(k_j + 1) = A_j(1,:) \otimes X_j(k_j) \oplus B_j(1,:) \otimes U_j(k_j + 1) \end{cases} \quad (19)$$

Remarque V.1 : les équations (19) peuvent être utilisées pour une simulation relativement rapide des systèmes fonctionnant suivant la règle PAPS.

Exemple V.1 : nous avons appliqué la règle PAPS au RGETC de la figure 3 en supposant que \tilde{t}_1 est prioritaire par rapport à \tilde{t}_2 en cas de demandes simultanées. Nous avons considéré que : tous les jetons sont disponibles initialement, $u_1(1) = 0$, $u_2(1) = 0$ et que les transitions d'entrée sont franchies périodiquement, toutes les 3 unités de temps pour t_{u1} et 7 unités pour t_{u2} . Pour des raisons de visibilité, nous avons limité le nombre de franchissement de t_{u1} à 6 et de t_{u2} à 9. Nous avons alors obtenu les résultats de la figure 4 :

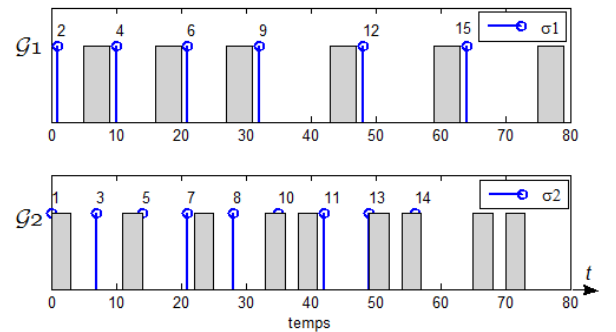


Fig. 4. Allocation PAPS de la ressource partagée de la Fig. 3. Chaque demande à la date σ_i engendre une consommation d'un jeton dont le numéro est affiché à côté de la date σ_i .

Sur la figure 4, on peut noter que le GET \mathcal{G}_1 consomme les jetons numéros : 2, 4, 6, 9, 12 et 15. Le GET \mathcal{G}_2 consomme quant à lui les jetons numéros : 1, 3, 5, 7, 8, 10, 11, 13, et 14. Ces séquences d'attribution du jeton de \tilde{p} sont, comme on peut le voir, non prévisibles malgré la périodicité des

transitions d'entrée. Ceci est dû au fait que la date σ_1 dépend non seulement de l'entrée $u_1(k_1)$ mais également de l'état du système ($x_1^2(k_1 - 1)$). Nous pouvons remarquer également qu'il y a demande simultanée à l'instant 21 ($\sigma_1(3) = \sigma_2(4)$). Comme \tilde{t}_1 est prioritaire par rapport à \tilde{t}_2 , le jeton numéro 6 est attribué à \tilde{t}_1 (ou consommé par \mathcal{G}_1).

B. Allocation périodique de la ressource partagée

Supposons maintenant que le jeton de la place de conflit soit attribué suivant une séquence périodique de la forme $(L)^\omega = LLLL \dots$, où

$$L = \underbrace{\mathcal{G}_1 \dots \mathcal{G}_1}_{m_1} \underbrace{\mathcal{G}_2 \dots \mathcal{G}_2}_{m_2} \dots \underbrace{\mathcal{G}_n \dots \mathcal{G}_n}_{m_n} \triangleq (\mathcal{G}_1)^{m_1} \cdot (\mathcal{G}_2)^{m_2} \dots (\mathcal{G}_n)^{m_n}.$$

Autrement dit, à un cycle donné, le jeton est attribué m_i fois pour le GET \mathcal{G}_i et au total $\bar{m} = \sum_{i=1}^n m_i$ fois. A la fin du $l^{\text{ème}}$ cycle, le jeton a déjà été attribué $k = l \cdot \bar{m}$ fois au total. Soit l_i le nombre de jetons attribués au GET \mathcal{G}_i à l'issue du $l^{\text{ème}}$ cycle, soit $l_i = l \times m_i$. Posons :

$$\bar{X}_i(l) = X_i(l_i) \quad (20)$$

Au début du $(l+1)^{\text{ème}}$ cycle, le premier jeton (numéro $k+1$) est attribué à \mathcal{G}_1 et le précédent jeton était pour \mathcal{G}_n . En utilisant les équations (13), nous avons :

$$\Psi(k+1) = G_n \otimes X_n(k_n) \quad (21)$$

$$X_1(k_1) = A_1 \otimes X_1(k_1 - 1) \oplus B_1 \otimes U_1(k_1) \oplus F_1 \otimes \Psi(k+1) \quad (22)$$

Comme le jeton est déjà attribué l_1 fois à \mathcal{G}_1 (donc $k_1 - 1 = l_1$) et l_n fois à \mathcal{G}_n ($k_n = l_n$), l'équation précédente devient :

$$X_1(l_1 + 1) = A_1 \otimes X_1(l_1) \oplus B_1 \otimes U_1(l_1 + 1) \oplus F_1 \otimes G_1 \otimes X_n(l_n) \quad (23)$$

En combinant (23) avec (20), nous obtenons :

$$X_1(l_1 + 1) = A_1 \otimes \bar{X}_1(l) \oplus B_1 \otimes U_1(l_1 + 1) \oplus F_1 \otimes G_n \otimes \bar{X}_n(l) \quad (24)$$

Les $(m_1 - 1)$ jetons suivants sont attribués au même GET i.e. \mathcal{G}_1 . Ceci implique, en utilisant (13) que :

$$X_1(l_1 + 2) = A_1 \otimes X_1(l_1 + 1) \oplus B_1 \otimes U_1(l_1 + 2) \oplus F_1 \otimes G_1 \otimes X_1(l_1 + 1)$$

En posant : $\bar{A}_1 = F_1 \otimes G_1 \oplus A_1$, l'équation précédente devient :

$$X_1(l_1 + 2) = \bar{A}_1 \otimes X_1(l_1 + 1) \oplus B_1 \otimes U_1(l_1 + 2)$$

De la même manière, nous avons :

$$\begin{aligned} X_1(l_1 + 3) &= \bar{A}_1 \otimes X_1(l_1 + 2) \oplus B_1 \otimes U_1(l_1 + 3) \\ &= \bar{A}_1^{(2)} \otimes X_1(l_1 + 1) \oplus \bar{A}_1 \otimes B_1 \otimes U_1(l_1 + 2) \oplus B_1 \otimes U_1(l_1 + 3) \end{aligned}$$

En continuant par récurrence, nous obtenons :

$$X_1(l_1 + m_1) = \bar{A}_1^{(m_1-1)} \otimes X_1(l_1 + 1) \oplus \bigoplus_{i=0}^{(m_1-2)} \bar{A}_1^{(i)} \otimes B_1 \otimes U_1(l_1 + m_1 - i) \quad (25)$$

Or, nous avons : $l_1 = l \cdot m_1$ et donc : $l_1 + m_1 = (l+1) \cdot m_1$, ce qui implique que : $X_1(l_1 + m_1) = \bar{X}_1(l+1)$. En utilisant cette propriété dans (25) et en remplaçant $X_1(l_1 + 1)$ par son expression (24), nous avons :

$$\begin{aligned} \bar{X}_1(l+1) &= \bar{A}_1^{(m_1-1)} \otimes A_1 \otimes \bar{X}_1(l) \oplus \bar{A}_1^{(m_1-1)} \otimes B_1 \otimes U_1(l_1 + 1) \oplus \\ &\bigoplus_{i=0}^{(m_1-2)} \bar{A}_1^{(i)} \otimes B_1 \otimes U_1(l_1 + m_1 - i) \oplus \bar{A}_1^{(m_1-1)} \otimes F_1 \otimes G_n \otimes \bar{X}_n(l) \end{aligned} \quad (26)$$

Cette équation peut se réécrire :

$$\begin{aligned} \bar{X}_1(l+1) &= \bar{A}_1^{(m_1-1)} \otimes A_1 \otimes \bar{X}_1(l) \oplus \bar{A}_1^{(m_1-1)} \otimes F_1 \otimes G_n \otimes \bar{X}_n(l) \\ &\bigoplus_{i=0}^{(m_1-1)} \bar{A}_1^{(i)} \otimes B_1 \otimes U_1(l_1 + m_1 - i) \end{aligned} \quad (27)$$

qui est sous la forme :

$$\bar{X}_1(l+1) = \alpha_1 \otimes \bar{X}_1(l) \oplus \beta_1 \otimes \bar{X}_n(l) \oplus \gamma_1 \otimes \bar{U}_1(l) \quad (28)$$

où $\alpha_1 = \bar{A}_1^{(m_1-1)} \otimes A_1$, $\beta_1 = \bar{A}_1^{(m_1-1)} \otimes F_1 \otimes G_n$,

$$\gamma_1 = \left(\bar{A}_1^{(0)} \quad \bar{A}_1^{(1)} \quad \dots \quad \bar{A}_1^{(m_1-1)} \right) \otimes B_1 \text{ et}$$

$$\bar{U}_1(l) = \left(U_1(l_1 + m_1) \quad U_1(l_1 + m_1 - 1) \quad \dots \quad U_1(l_1 + 1) \right)^t.$$

En reprenant la même procédure pour \mathcal{G}_2 , nous avons :

$$\bar{X}_2(l+1) = \alpha_2 \otimes \bar{X}_2(l) \oplus \beta_2 \otimes \bar{X}_1(l+1) \oplus \gamma_2 \otimes \bar{U}_2(l) \quad (29)$$

En remplaçant $\bar{X}_1(l+1)$ par son expression (28), nous avons :

$$\begin{aligned} \bar{X}_2(l+1) &= \alpha_2 \otimes \bar{X}_2(l) \oplus \beta_2 \otimes \alpha_1 \otimes \bar{X}_1(l) \oplus \beta_2 \otimes \beta_1 \otimes \bar{X}_n(l) \\ &\oplus \beta_2 \otimes \gamma_1 \otimes \bar{U}_1(l) \oplus \gamma_2 \otimes \bar{U}_2(l) \end{aligned} \quad (30)$$

En posant, $\bar{A}_q = F_q \otimes G_q \oplus A_q$, $\alpha_q = \bar{A}_q^{(m_q-1)} \otimes A_q$, $a_{qn} = \bigotimes_{i=1}^q \beta_i$, $\beta_q = \bar{A}_q^{(m_q-1)} \otimes F_q \otimes G_{(q-1)}$, $\gamma_q = \left(\bar{A}_q^0 \quad \bar{A}_q^1 \quad \dots \quad \bar{A}_q^{(m_q-1)} \right) \otimes B_q$

et $\bar{U}_q(l) = \left(U_q(l_q + m_q) \quad U_q(l_q + m_q - 1) \quad \dots \quad U_q(l_q + 1) \right)^t$, l'expression (30) peut se généraliser pour $2 \leq q \leq n$:

$$\bar{X}_q(l+1) = \bigoplus_{j=1}^q \left[a_{qj} \otimes \bar{X}_j(l) \right] \oplus a_{qn} \otimes \bar{X}_n(l) \oplus \bigoplus_{j=1}^q \left[b_{qj} \otimes \bar{U}_j(l) \right] \quad (31)$$

où $a_{qj} = \left[\bigotimes_{i=j+1}^q \beta_i \right] \otimes \alpha_j$ et $b_{qj} = \left[\bigotimes_{i=j+1}^q \beta_i \right] \otimes \gamma_j$ pour $j < q$ et $a_{qq} = \alpha_q$ et $b_{qq} = \gamma_q$. En ajoutant les composantes $a_{qj} = I_\varepsilon$ et $b_{qj} = I_\varepsilon$ (I_ε étant la matrice nulle avec la dimension adéquate) pour $(n-1) \geq j > q$, nous aboutissons finalement à :

$$\bar{X}(l+1) = A \otimes \bar{X}(l) \oplus B \otimes \bar{U}(l+1) \quad (32)$$

où $A(i, j) = a_{ij}$, $B(i, j) = b_{ij}$, $\bar{X}(l) = \left(\bar{X}_1(l) \quad \bar{X}_2(l) \quad \dots \quad \bar{X}_n(l) \right)^t$, $\bar{U}(l+1) = \left(\bar{U}_1(l) \quad \bar{U}_2(l) \quad \dots \quad \bar{U}_n(l) \right)^t$. Comme on peut le constater, la forme (32) est une représentation d'état (max,+)

linéaire standard. Naturellement, cette forme peut être appliquée pour l'analyse du comportement des RGETC en régime permanent en utilisant les propriétés spectrales de la matrice A . Dans le cas d'un système autonome ou non contraint (pas de rupture de stock par exemple), la matrice B est nulle et par conséquent l'équation (32) devient :

$$\bar{X}(l+1) = A \otimes \bar{X}(l) \quad (33)$$

Le taux de franchissement asymptotique de la transition \tilde{t}_i du

GET \mathcal{G}_i est donné par : $\lambda_i = \lim_{k \rightarrow \infty} k / x_i^1(k) = m_i / \rho(A_{IR})$ où

$\rho(A_{IR})$ est la valeur propre de la matrice irréductible A_{IR} (A_{IR} étant une matrice dont les composantes sont celles de A mais en ne gardant que les dateurs des transitions qui ont une place en aval marquée et impliquées dans les circuits). Voir [2] et [12] pour plus de détails sur la théorie spectrale et les GET.

Remarque V.2 : On peut démontrer que les équations (13) peuvent être reformulées, dans le cas d'un RGETC cyclique, par un système d'équations linéaires à paramètres variants

périodiques. Par conséquent, on peut appliquer les transformations présentées dans [13] pour aboutir à la forme standard (32). Pour des raisons évidentes, nous nous contentons, pour le présent papier, de la méthode présentée précédemment.

Exemple V.2 : pour illustrer la méthode proposée, nous considérons un système inspiré de celui traité dans [6] (nous avons transformé les transitions temporisées du RdP original en places temporisées), composé de deux ressources partagées et de deux utilisateurs (figure 5) :

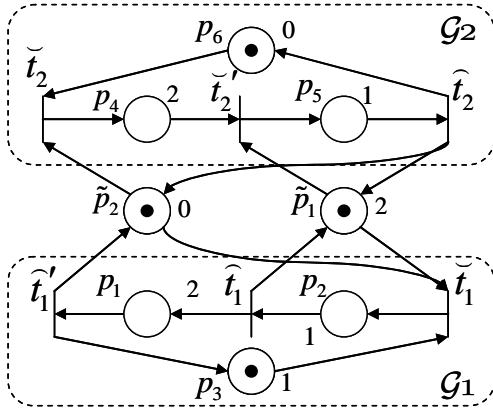


Fig. 5. Le RGETC étudié (deux GET et deux places de conflit).

Dans cet exemple, nous avons deux GET avec deux places de conflit. En associant les dateurs $x_1^1, x_1^2, x_1^3, x_2^1, x_2^2, x_2^3$ aux transitions $\tilde{t}_1, \hat{t}_1, \tilde{t}'_1, \tilde{t}_2, \hat{t}_2, \tilde{t}'_2$ et les dates de disponibilité $\tilde{\psi}_1, \tilde{\psi}_2$ aux jetons des places \tilde{p}_1, \tilde{p}_2 , nous trouvons la forme

$$(13) \text{ avec : } X_1 = (x_1^1 \ x_1^2 \ x_1^3)^t, \ X_2 = (x_2^1 \ x_2^2 \ x_2^3)^t,$$

$$A_1 = \begin{pmatrix} \varepsilon & \varepsilon & 1 \\ \varepsilon & \varepsilon & 2 \\ \varepsilon & \varepsilon & 4 \end{pmatrix}, \ A_2 = \begin{pmatrix} \varepsilon & \varepsilon & e \\ \varepsilon & \varepsilon & 2 \\ \varepsilon & \varepsilon & 3 \end{pmatrix}, \ F_1 = \begin{pmatrix} e & e \\ 1 & 1 \\ 3 & 3 \end{pmatrix}, \ F_2 = \begin{pmatrix} \varepsilon & e \\ e & 2 \\ 1 & 3 \end{pmatrix},$$

$$G_1 = \begin{pmatrix} \varepsilon & 2 & \varepsilon \\ \varepsilon & \varepsilon & e \end{pmatrix} \text{ et } G_2 = \begin{pmatrix} \varepsilon & \varepsilon & 2 \\ \varepsilon & \varepsilon & e \end{pmatrix}.$$

En utilisant les résultats précédents relatifs à l'allocation périodique des ressources et pour une séquence $L = (G_1)^m \cdot (G_2)^m$, nous obtenons la forme (33) où :

$$A = \begin{pmatrix} \varepsilon & \varepsilon & 4m-3 & \varepsilon & \varepsilon & 4m-2 \\ \varepsilon & \varepsilon & 4m-2 & \varepsilon & \varepsilon & 4m-1 \\ \varepsilon & \varepsilon & 4m & \varepsilon & \varepsilon & 4m+1 \\ \varepsilon & \varepsilon & 7m-3 & \varepsilon & \varepsilon & 7m-2 \\ \varepsilon & \varepsilon & 7m-1 & \varepsilon & \varepsilon & 7m \\ \varepsilon & \varepsilon & 7m & \varepsilon & \varepsilon & 7m+1 \end{pmatrix},$$

et en ne gardant que les dateurs des transitions avec des places en aval marquées (x_1^3 et x_2^3), nous obtenons :

$$A_{IR} = \begin{pmatrix} 4m & 4m+1 \\ 7m & 7m+1 \end{pmatrix}.$$

La matrice A_{IR} a une valeur propre unique : $\rho(A_{IR}) = 7m+1$ et par conséquent : $\lambda_1 = \lambda_2 = m/(7m+1)$, ce qui donne le

même résultat que celui obtenu dans [6]. L'avantage de notre méthode est d'éviter de construire un automate de type tas de pièces ou de transformer le RdP en un GET équivalent. Dans un cas plus général, la première méthode s'applique seulement si le RdP est sauf ; la taille du GET équivalent de la seconde méthode devient très volumineux si la séquence L est longue (Voir dans [6] les résultats de transformation du RdP en GET pour la séquence $L = (G_1)^m \cdot (G_2)^m$ où $m=1$ puis $m=2$).

IV. CONCLUSION

Dans cette étude, nous avons présenté une approche algébrique (Max,+) pour la modélisation de réseaux de graphes d'événements temporisés avec conflits. Des formes semblables à la représentation d'état des GET sont données indépendamment de la politique d'arbitrage de conflits appliquée. Pour une politique d'arbitrage périodique, la forme standard a été obtenue systématiquement sans passer par un GET équivalent ou un automate de type tas. Pour les travaux futurs, nous projetons d'appliquer les résultats de cette étude à l'évaluation de délais dans les réseaux de communication complexes. Dans un contexte plus général, il serait intéressant de se pencher sur une relaxation de l'hypothèse sur le marquage des places de conflits et d'étendre l'analyse à des classes de RdP plus larges.

RÉFÉRENCES

- [1] Murata T. Petri nets: properties analysis and applications. Proceedings of the *IEEE*, 77(4), pp. 541-580, 1989.
- [2] Baccelli F., Cohen G., Olsder G.-J. et Quadrat J.-P. Synchronization and linearity: an algebra for discrete event systems, Wiley, 1992.
- [3] Addad B. et Amari S. Modeling and response time evaluation of Ethernet based automation systems using Max-Plus algebra. proc of 4th IEEE Conference on Automation Science and Engineering, Washington DC, USA, pp. 418-423, 2008.
- [4] Hillion H. et Proth J.-M. Performance evaluation of jobshop systems using timed event graphs, *IEEE Transaction on Automatic Control*, 34(1), pp. 3-9, 1989.
- [5] Trouillet B. Sur l'évaluation du comportement logique des systèmes de production manufacturière par méthodes exactes. Thèse, LAIL – Université de Lille 1, 2003.
- [6] Gaubert S. et Mairesse J. Modeling and analysis of timed Petri nets using heaps of pieces. *IEEE Transactions on Automatic Control*, 44(4), pp. 683-697, 1999.
- [7] Baccelli F., Foss S. et Gaujal B. Free choice Petri nets: an algebraic approach. *IEEE Transactions on Automatic Control*, 41(12), pp. 1751-1778, 1996.
- [8] Van Den Boom T.-J.-J. et De Shutter B. Modeling and control of discrete event systems using switching Max-Plus linear systems, *Control Engineering Practice*, 14(10), pp. 1199-1211, 2006.
- [9] Correia A., Abbas Turki A., Bouyekhf et El Moudni A. A dioid model for invariant resource sharing problems. *IEEE Transactions on systems Man and Cybernetics (Systems and Humans)*, 39(4), pp. 770-781, 2009.
- [10] Nait S.-M., Manier M.-A., El Moudni A. et Wack M. Petri net with conflicts and (max,plus) algebra for transportation systems, proc of 11th IFAC Symposium on Transportation Systems, The Netherlands, 2006.
- [11] Boutin O., Cottenceau B., L'Anton A. et Loiseau J.-J. Modeling systems with periodic routing functions in dioid (Min,+), proc of 13th IFAC Symposium on Information Control Problems in Manufacturing, Moscow, Russia, pp. 1377-1382, 2009.
- [12] Cohen G., Gaubert S., et Quadrat J.-P. Max plus algebra and systems theory: where we are and where to go now, *Annual Reviews in Control*, vol. 23, pp. 207-219, 1999.
- [13] Lahaye S., Boimond J. L. et Hardouin L., Linear periodic systems over dioids, *Discrete Event Dynamic Systems*, 14(2), pp 133-152, 2004.