



THESIS / THÈSE

MASTER EN SCIENCES INFORMATIQUES

Sur quelques méthodes récentes d'analyse des réseaux de Pétri colorés

Campers, Christine

Award date:
1990

Awarding institution:
Universite de Namur

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Facultés Universitaires Notre-Dame de la Paix à Namur
Institut d'informatique

Année académique 1989-1990

SUR QUELQUES METHODES RECENTES

D'ANALYSE DES RESEAUX DE PETRI

COLORES

Ch. CAMPERS

Résumé.

Dans de nombreux systèmes, on peut rencontrer différents processus qui possèdent une structure et un comportement semblables. Pour abrégé la description des systèmes et réduire leur analyse, il est préférable de traiter de tels processus similaires de façon identique.

Nous montrons comment les réseaux de Pétri peuvent être généralisés pour décrire ces différents processus par un même sous-réseau en conservant la possibilité de les distinguer. Cette généralisation porte le nom de réseau de Pétri coloré.

Pour analyser les réseaux colorés, la méthode de l'invariant a été utilisée avec succès. Nous décrivons deux méthodes d'algèbre linéaire pour trouver les invariants d'un réseau de Pétri coloré.

Nous montrons que la programmation linéaire est un outil supplémentaire pour l'analyse des propriétés structurelles et locales des réseaux de Pétri. Nous présentons une généralisation de cette méthode pour les réseaux colorés basée sur le principe de décomposition de Dantzig et Wolfe.

Abstract.

In many systems a number of different processes have a similar structure and behaviour. To shorten system description and system analysis it is desirable to be able to treat such similar processes in a uniform and succinct way.

We show how Petri nets can be generalized to allow processes to be described by a common subnet, without losing the ability to distinguish between them. This generalization is called coloured Petri net.

To analyse coloured nets, the method of linear invariants has been used successfully. We describe two methods based on linear algebra to find invariants in a coloured Petri net.

We show that linear programming is an alternative tool for the analysis of structural and local properties of Petri nets. We present a generalisation of this method for the coloured nets based on the decomposition principle of Dantzig et Wolfe.

Je remercie sincèrement Monsieur Fichet d'avoir accepté la direction de ce mémoire. Il m'a permis d'accomplir un travail personnel tout en m'apportant une aide efficace et précieuse.

Je tiens également à remercier mon mari, Didier, pour la compréhension dont il a fait preuve ainsi que pour sa collaboration à la dactylographie de ce manuscrit.

Finalement, un grand merci à toutes les personnes qui, de près ou de loin, ont contribué à la réalisation de ce travail.

TABLE DES MATIERES.

	<u>page</u>
Introduction	1
Chapitre 1 - Présentation des réseaux de Pétri colorés	4
I. Préliminaire	
II. Présentation intuitive des réseaux de Pétri colorés	9
III. Définition d'un RPC	14
IV. Exemples de réseaux de Pétri colorés	15
1. Réseau de bases de données	15
2. Le système du téléphone	18
Chapitre 2 - L'algèbre linéaire	22
I. Introduction	22
II. Les réseaux de Pétri	23
III. Les réseaux de Pétri colorés	26
1. Introduction	26
2. Exemple : réseau de bases de données	28
IV. Méthode de recherche des invariants d'un RPC	31
1. Règles de transformation	32
2. Exemples	35
2.1 Réseau de bases de données	35
2.2 Le système du téléphone	37
V. Méthode alternative basée sur les fonctions inverses généralisées	41
1. Invariants d'un RPC avec fonction-identité	43
2. Calcul des invariants symboliques d'un RPC	45
2.1 Cas où W est factorisable	45
2.1.1 Factorisation de W	47
2.1.2 Calcul des invariants symboliques d'un RPC_W à partir des invariants symboliques d'un RPC_a	48
2.1.3 Calcul d'un ensemble d'invariants symboliques de RPC_W	53
2.2 Cas général : W non factorisable	63
VI. Conclusion	67

Chapitre 3 - La programmation linéaire	69
I. Introduction	69
II. Rappels de Programmation linéaire	69
1. Rappels mathématiques	72
2. Algorithme du simplexe	73
3. Interprétation géométrique de l'algorithme du simplexe	74
4. Méthode lexicographique	75
4. Forme révisée de la méthode du simplexe	76
III. Application de la programmation linéaire aux réseaux de Pétri	77
1. Cas des réseaux bornés	78
2. Calcul des invariants	78
3. Cas des réseaux vivants	79
IV. Principe de décomposition de Dantzig et Wolfe	80
1. Introduction	80
2. Domaine d'application	81
3. Formulation du problème original et d'un problème équivalent	81
4. Algorithme de décomposition	84
5. Extension au cas où les S_j ne sont pas tous bornés	86
6. Application aux réseaux colorés	87
6.1 Cas des réseaux colorés	88
6.2 Calcul des invariants	88
6.3 Cas des réseaux vivants	88
7. Conclusion	88
Conclusion	90
Références	91

INTRODUCTION.

L'homme dans son souci d'étudier le monde qui l'entoure a vite découvert l'avantage de la modélisation.

Un modèle est une représentation, souvent mathématique, de ce que l'on considère comme les traits importants de l'objet ou du système étudié. Il permet d'étoffer les connaissances à propos d'un phénomène sans danger, coût ou manipulation du système réel.

Les systèmes informatiques sont très complexes, souvent considérables et constitués d'un grand nombre de composants qui interagissent entre eux. Chaque composant peut lui-même être assez complexe, ainsi que les interactions avec les autres parties du système.

Ce genre de structure se retrouve dans d'autres secteurs. Ainsi, en Economie, en Chimie ou même en Droit, on peut rencontrer de telles organisations.

Donc, malgré la diversité des systèmes à modéliser, il est possible d'extraire quelques points communs. Ceux-ci pourraient constituer les grandes lignes d'un modèle utile dans toutes ces activités.

Ce modèle se base sur une idée fondamentale : la présence de composants séparés et interactifs dont certains peuvent être eux-mêmes des systèmes mais dont le comportement peut être décrit indépendamment des autres, à l'exception de leur interaction.

Chaque élément possède ses propres états. L'état d'un composant se définit par une abstraction de l'information nécessaire pour décrire ses actions.

Souvent, un état dépend de l'histoire du composant. Il peut donc changer au cours du temps.

Pour illustrer ce concept, prenons l'exemple d'une file d'attente dans une banque. Deux parties sont en présence : les clients et les caissiers. Ceux-ci peuvent se trouver dans deux états différents. Un caissier peut être inactif (dans l'attente d'un client) ou occupé (à servir un client). De la même façon, un client peut être inactif (dans la file d'attente) ou occupé (à être servi par un caissier).

On peut rencontrer des activités simultanées par rapport à d'autres d'un composant voisin. Ces composants seront alors concurrents ou parallèles.

Dans un système informatique, par exemple, on cherchera à optimiser l'emploi de diverses ressources ; on aboutit à la notion de partage de ressources. Sur un autre plan, on pourra augmenter la fiabilité ou la tolérance aux défauts d'un système en faisant fonctionner en parallèle, plusieurs sous-systèmes identiques. De même, dans un système économique, des usines peuvent fabriquer certains produits alors que des détaillants en vendent d'autres et des particuliers en consomment d'autres encore, tout cela en même temps.

Ces caractéristiques posent de sérieux problèmes de modélisation et de telles interactions nécessitent une synchronisation entre les activités.

Le transfert d'informations ou de matériaux d'un composant à un autre exige notamment une synchronisation entre les actions des sous-systèmes qui interagissent. L'ordonnancement des tâches des différents composants peut être ardu et les interactions résultantes difficiles à décrire.

C'est pourquoi, depuis quelques années, de nombreux organismes et entreprises tentent de mettre au point des modèles susceptibles de les aider pour l'analyse de tels systèmes.

La modélisation doit être suffisamment fine pour décrire avec précision certains mécanismes délicats tout en donnant une vision globale du système.

Les réseaux de Pétri ont été conçus pour modéliser les concepts d'actions asynchrones et concurrentes. Ils ont été introduits par Carl Adam Pétri dans les années 60-62.

Cependant, la puissance limitée du modèle de base a vite poussé les chercheurs à l'étendre de façon à augmenter la classe des systèmes que l'on peut modéliser.

Le grand nombre de processus différents au comportement et à structure similaires a incité à introduire une extension particulière : les réseaux de Pétri colorés.

C'est ce sujet qui est abordé dans ce travail et plus spécialement, quelques méthodes d'analyse de ces réseaux.

Le premier chapitre est consacré à quelques rappels concernant le modèle général ainsi qu'à une présentation des réseaux de Pétri colorés.

Le reste du travail tente de résumer quelques techniques d'analyse. Au deuxième chapitre, la méthode de l'invariant est présentée et traitée selon deux points de vue différents.

Enfin, au troisième chapitre, c'est la programmation linéaire qui est étudiée ainsi que son application à la théorie des réseaux de Pétri.

CHAPITRE 1 - PRESENTATION DES RESEAUX DE PETRI COLORES.

I. Préliminaire.

Les réseaux de Pétri ont été imaginés dans le but de modéliser le comportement dynamique de systèmes discrets. Ils sont composés de deux types d'éléments : les places et les transitions. Ceux-ci sont reliés par des arcs dirigés d'une place vers une transition ou d'une transition vers une place.

L'ensemble des places permet de représenter les états du système ; l'ensemble des transitions modélise des événements dont l'effet est de modifier l'état du système.

L'exécution d'un réseau de Pétri est contrôlée par la position et le mouvement de marques contenues dans les places du réseau.

Le déplacement de marques est une conséquence du tir d'une transition. Ce tir correspond à l'occurrence de l'événement représenté par la transition considérée. Elle dépend de la satisfaction de préconditions. Celles-ci portent sur le nombre de marques contenues dans les places attachées à la transition associée à l'événement considéré.

Définitions. [1]

- Un réseau de Pétri est un quadruplet $R = (P, T, \text{Pré}, \text{Post})$ où
 - 1) P est un ensemble fini de places ($|P| = m$)
 - 2) T est un ensemble fini de transitions disjoint de P ($|T| = n$)
 - 3) $\text{Pré} : P \times T \rightarrow N$ est l'application d'incidence avant
 - 4) $\text{Post} : P \times T \rightarrow N$ est l'application d'incidence arrière
- Un réseau marqué est un couple (R, M) où R est un réseau de Pétri et $M : P \rightarrow N$ est l'application de marquage.
- $M(p)$ est le marquage de la place $p \in P$ qui représente le nombre de marques présentes dans la place p .

- Dans un réseau marqué $N = (R, M)$ avec $R = (P, T, \text{Pré}, \text{Post})$, une transition $t \in T$ est franchissable pour le marquage M si et seulement si :

$$\forall p \in P : M(p) \geq \text{Pré}(p,t)$$

- Une transition $t \in T$ est activée quand toutes ses places d'entrée sont marquées, c'est-à-dire contiennent un nombre non nul de marques.
- En désignant les transitions d'un réseau par les symboles d'un vocabulaire T , un mot $s \in T^*$ dénote une séquence de transitions. Le mot vide k dénotant une séquence vide.
Une séquence s de T^* est franchissable pour un réseau marqué, permettant d'obtenir un marquage \bar{M} , ce que l'on note $M (s > \bar{M})$, si et seulement si :

soit $s = k$ auquel cas $\bar{M} = M$
soit $s = s't$ avec $s' \in T^*$ et $t \in T$ et de plus, il existe M' tel que

$$M (s' > M' \text{ et } M' (t > \bar{M})$$

On dit que s est une séquence de franchissements pour M .

- S'il existe une séquence de franchissements s telle que $M (s > \bar{M})$ dans le réseau R , alors on dit que \bar{M} est un marquage accessible à partir de M et l'on note :

$$A(R, M) = \{ \bar{M} \in N^m \text{ tq il existe } s : M (s > \bar{M}) \}$$

- En utilisant des notations matricielles pour représenter les applications Pré et Post , on peut définir la matrice d'incidence associée à un réseau de Pétri par la relation :

$$\forall (p,t) \in P \times T : W(p,t) = \text{Post}(p,t) - \text{Pré}(p,t)$$

Propriétés.

a. Réseaux bornés.

Il s'agit d'une classe particulière de réseaux dans lesquels le nombre de marques dans toutes les places est borné quoiqu'il arrive. Cette propriété est souvent recherchée, notamment dans un système où intervient une ressource dont la capacité est limitée.

- Soit $N = (R, M)$ avec $R = (P, T, \text{Pré}, \text{Post})$, un réseau marqué. On dit qu'une place $p \in P$ est k-bornée, avec $k \in \mathbb{N}$, si $\bar{M}(p) \leq k$ pour tout marquage accessible $\bar{M} \in A(R, M)$. Dans le cas contraire, on dit que la place p est non bornée.
Une place 1-bornée est aussi appelée place binaire.
- Si toutes les places d'un réseau marqué (R, M) sont k-bornées, où $k \in \mathbb{N}$, alors on dit que ce réseau marqué est borné. Un réseau R est dit borné si le réseau marqué (R, M) est borné pour tout marquage M .

b. Réseaux vivants.

La vivacité d'une transition est une propriété importante pour traduire le bon fonctionnement de nombreux systèmes. Elle exprime le fait qu'à tout moment de l'évolution du système, il sera toujours possible de franchir cette transition. Cette caractéristique est importante pour les systèmes dont l'indisponibilité d'une fonction correspond à une erreur ou une panne.

- On dit qu'une transition t d'un réseau marqué (R, M) est quasi-vivante s'il existe un marquage accessible $\bar{M} \in A(R, M)$ tel que $\bar{M}(t) > 0$. Quand toutes les transitions d'un réseau marqué sont quasi-vivantes, on dit que ce réseau marqué est quasi-vivant.
- On dit qu'une transition t d'un réseau marqué (R, M) est vivante si, pour tout marquage accessible $M \in A(R, M)$, t est quasi-vivante dans le réseau (R, M) .

- Un réseau marqué (R, M) est dit vivant si toutes ses transitions sont vivantes. Un réseau R est vivant s'il existe un marquage M tel que le réseau marqué (R, M) soit vivant.

c. Réseaux sans blocage.

La notion de réseau sans blocage est également très utile en pratique. Elle définit un réseau dans lequel à tout moment, il existe au moins une transition franchissable. Cette propriété permet notamment de déterminer si un système est capable de réaliser l'ensemble des fonctions pour lesquelles il a été conçu.

- Un marquage M d'un réseau R est dit marquage "puits" si aucune transition n'est franchissable depuis M .
- On dit qu'un réseau marqué (R, M) est sans blocage si tout marquage accessible depuis M n'est pas un marquage "puits". Un réseau R est dit sans blocage s'il existe un marquage initial M_0 tel que le réseau marqué (R, M_0) soit sans blocage.

Dans de nombreux systèmes, on peut rencontrer différents processus qui possèdent une structure et un comportement semblables. C'est cette constatation qui a conduit les chercheurs à définir une des extensions du modèle de base des réseaux de Pétri : les réseaux de Pétri colorés (RPC).

Ils fournissent un réel parallélisme entre les processus tout en permettant de les distinguer et procurent aux informaticiens une description des systèmes plus condensée, donc plus claire et leur facilitent le travail d'analyse (fig. 1) [2].

Dans un réseau de Pétri coloré, chaque marque est liée à une couleur. Cette association permet d'identifier les différentes marques à l'intérieur du réseau. De plus, on fait correspondre à chaque place et à chaque transition, un ensemble de couleurs. Une transition pourra être tirée par rapport à chacune de ses couleurs. Lors d'un tir, des marques seront retirées des places d'entrée et d'autres ajoutées aux places de sortie de la façon habituelle à l'exception de la dépendance

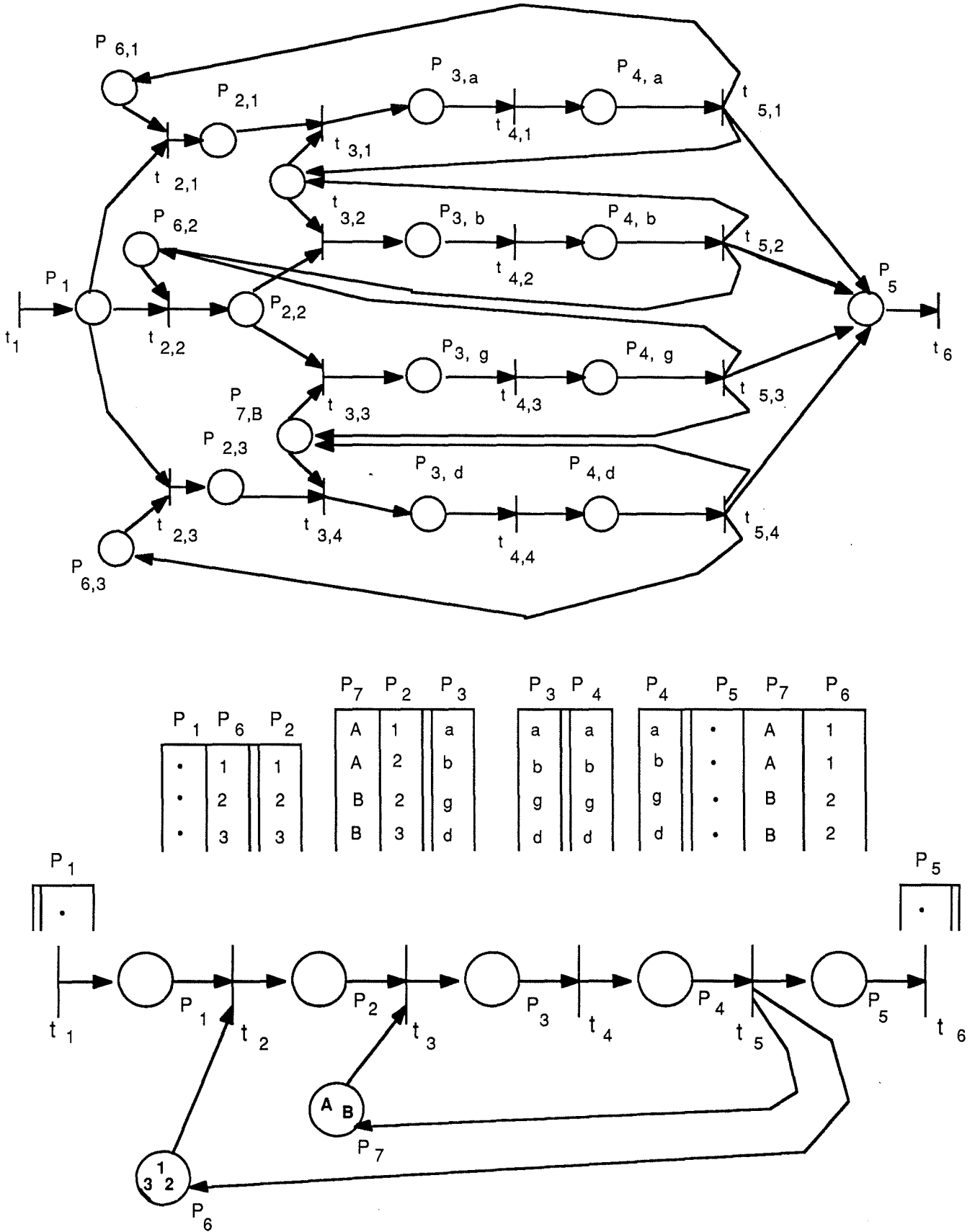


fig. 1 : comparaison entre un réseau de Pétri et un réseau de Pétri coloré modélisant le même système

fonctionnelle qui existe entre la (ou les) couleur(s) de la transition franchie et les couleurs des marques concernées par le tir.

II. Présentation intuitive des réseaux de Pétri colorés.

Pour aider à percevoir davantage l'intérêt des RPC, nous allons présenter un exemple dans lequel un réseau de Pétri assez complexe est transformé en un RPC plus simple qui conserve les caractéristiques du modèle original.

Il s'agit d'un exemple typique de synchronisation : le problème des cinq philosophes [3]. Il a été suggéré en 1968 par Dijkstra et concerne cinq philosophes qui pensent et mangent alternativement (fig. 2).

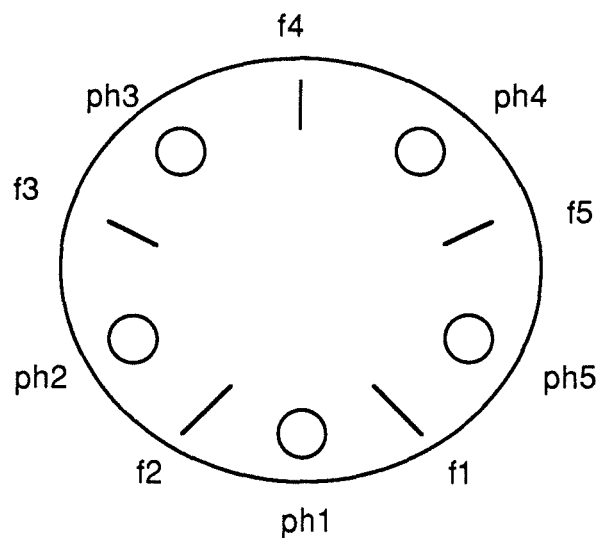


fig. 2 : le problème des cinq philosophes

Pour manger, un philosophe a besoin de deux fourchettes. Malheureusement, la table ronde autour de laquelle ils sont assis, n'a été garnie que de cinq fourchettes et chaque philosophe ne peut utiliser que les deux fourchettes placées à côté de lui. Ceci implique évidemment que deux voisins ne peuvent pas manger en même temps.

Ce système a été décrit par un réseau de Pétri. Sa représentation graphique est donnée à la fig. 3.

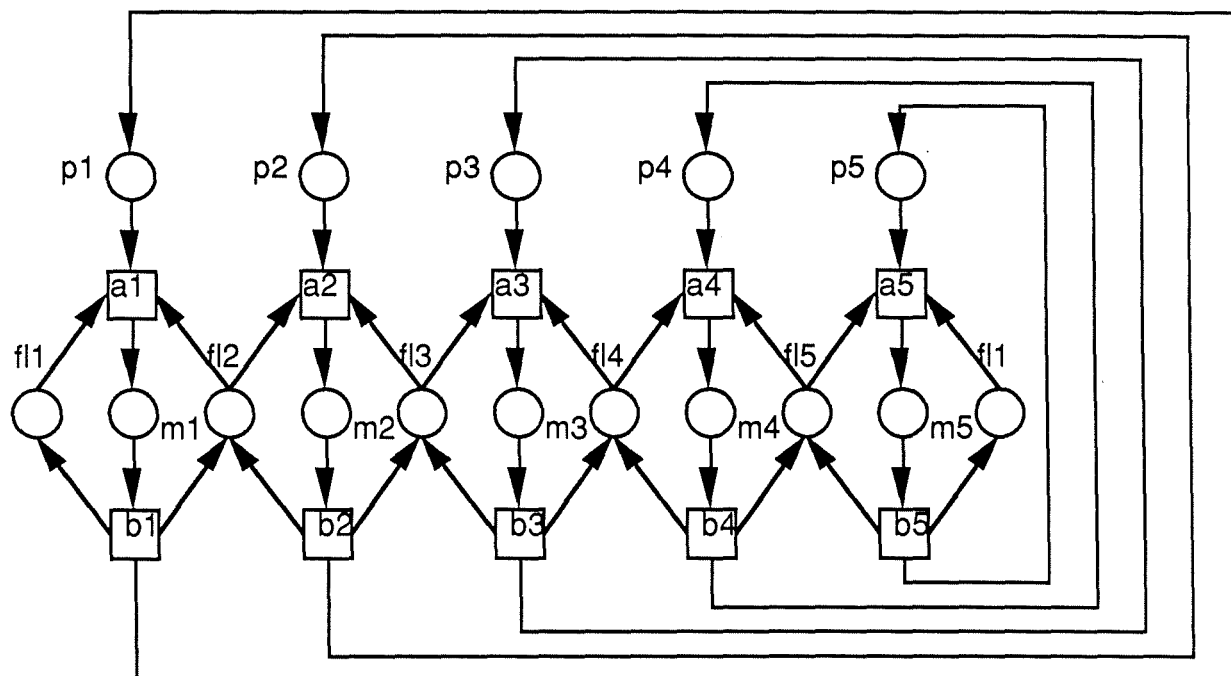


fig. 3 : réseau de Pétri modélisant le problème des cinq philosophes

Il est tentant, vu la répétition de la même structure de base pour chaque philosophe, de condenser ce graphe en une représentation plus simple (fig. 4).

Malheureusement, ce schéma ne nous donne plus une description correcte du problème initial. En effet, on voit que cette représentation permet aux philosophes de choisir leurs fourchettes arbitrairement parmi les cinq disposées sur la table. Ainsi, deux voisins peuvent manger simultanément.

Notre but est d'obtenir un graphe de la même taille que la fig. 4 tout en respectant les contraintes de départ. Pour cela, il faut pouvoir distinguer les marques correspondant aux différents philosophes ainsi que celles représentant les cinq fourchettes. A cet effet, nous allons associer un ensemble de couleurs $PH = \{ph_1, ph_2, ph_3, ph_4, ph_5\}$ aux places "pense" et "mange" et un ensemble de couleurs $F = \{f_1, f_2, f_3, f_4, f_5\}$ à la place "fourchettes libres".

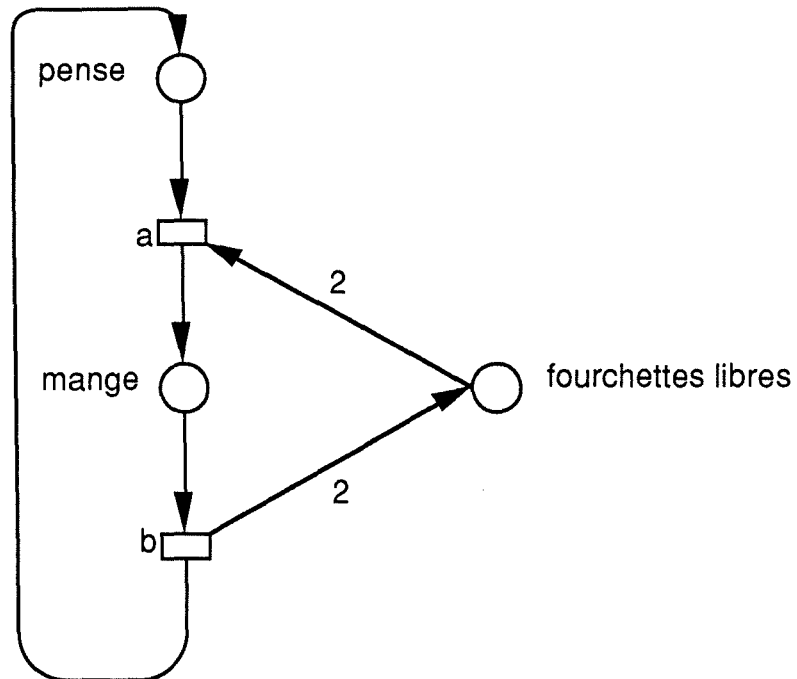


fig. 4 : transformation du réseau de Pétri de la fig. 3

A ce stade de développement, chaque transition a_i de la fig. 2 peut prendre la forme suivante :

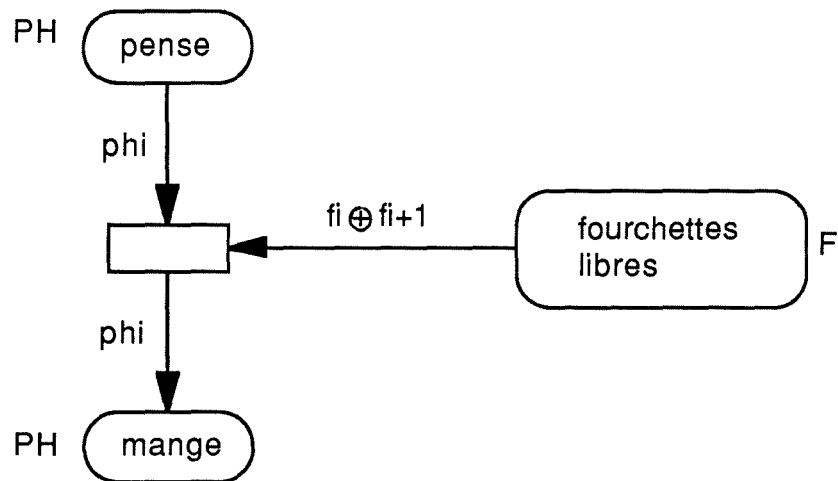


fig. 5 : réseau dont certaines places ont été unifiées

La somme $f_i \oplus f_{i+1}$ indique que le tir de a_i nécessite non seulement une marque ph_i de la place "pense" mais aussi deux marques de la place "fourchettes libres" qui doivent correspondre à deux fourchettes entourant le philosophe ph_i . Le tir de a_i ajoutera alors une marque ph_i dans la place "mange". Une généralisation de ce schéma est donnée à la fig. 6.

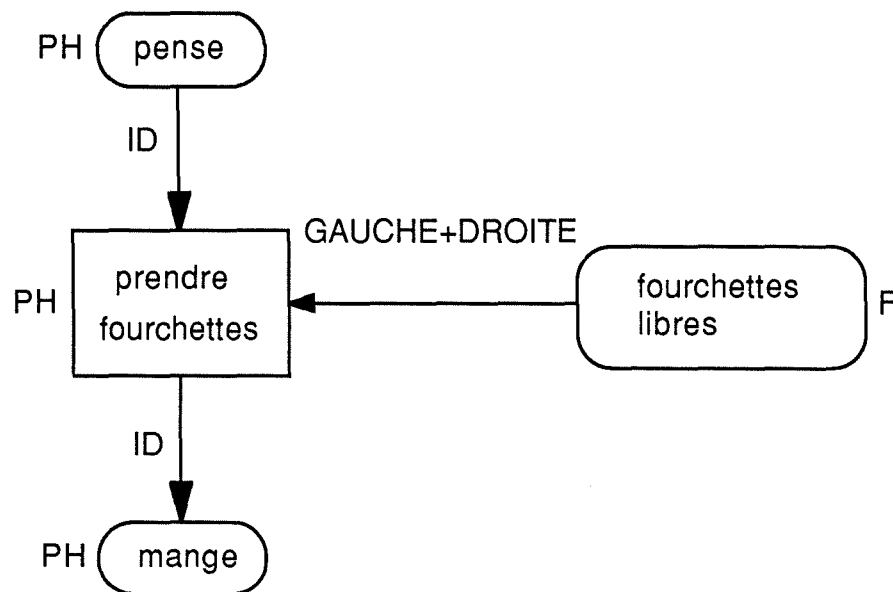


fig. 6 : réseau dont toutes les places ont été unifiées

La notation 'PH' à côté de la transition "prendre fourchettes" indique que celle-ci peut être tirée par rapport à chaque couleur appartenant à l'ensemble PH.

Les fonctions 'ID', 'GAUCHE' et 'DROITE' signifient que pour franchir la transition "prendre fourchettes" pour un philosophe ph_i , il faut retirer la marque $ID(ph_i) = ph_i$ (fonction identité) de la place "pense" et deux marques de la place "fourchettes libres" qui vérifient $GAUCHE(ph_i)$ (la fourchette gauche du philosophe ph_i) et $DROITE(ph_i)$ (sa fourchette droite). Le tir de cette transition ajoutera une marque $ID(ph_i) = ph_i$ à la place "mange".

De manière analogue, on peut remplacer les transitions b_i du réseau de Pétri de base par l'unique transition "déposer fourchettes" à laquelle on associe également l'ensemble de couleurs PH. Nous obtenons ainsi le RPC de la fig. 7 où,

par convention, les arcs non étiquetés correspondent à la fonction identité. On peut finalement démontrer que ce RPC possède les mêmes propriétés que le réseau de départ.

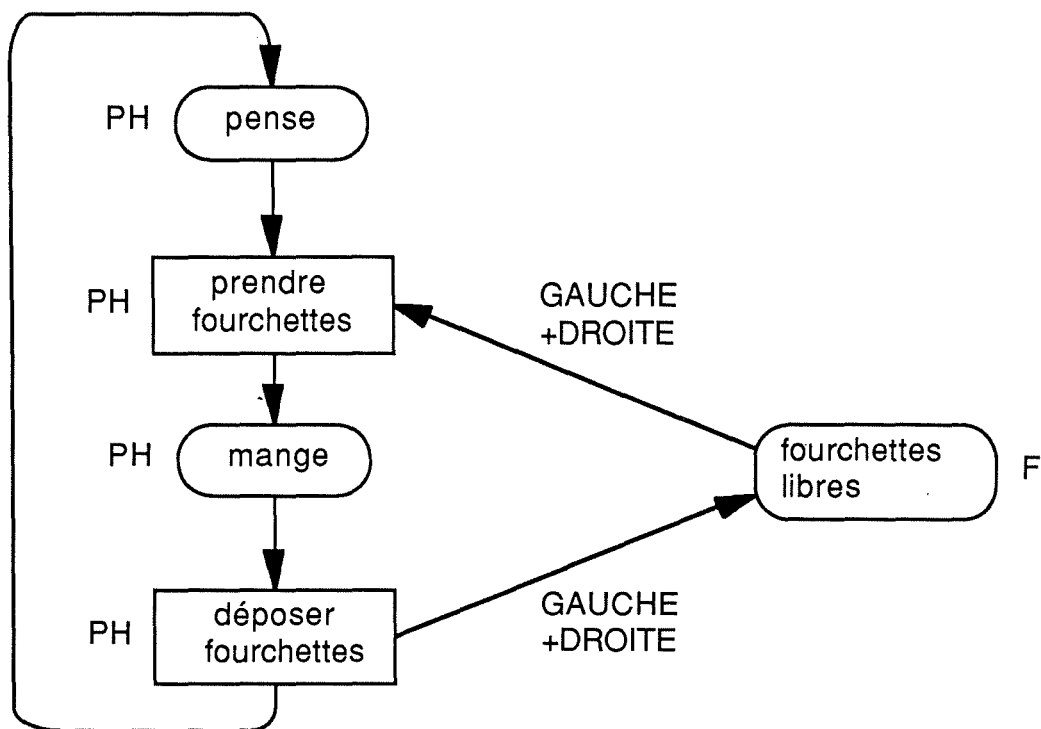


fig. 7 : RPC modélisant le système des cinq philosophes

Il faut finalement signaler que cette construction n'est pas unique. En fait, pour un réseau de Pétri donné, chaque association de places combinée à chaque association de transitions détermine un RPC. On peut obtenir ainsi toute une série de réseaux de Pétri colorés qui se situent entre deux extrêmes. Il s'agit d'un RPC contenant le même nombre de places et de transitions que le réseau d'origine et un RPC composé d'une seule place et d'une seule transition. Dans le premier cas, chaque place et chaque transition sont associés à un ensemble de couleurs composé d'un seul élément. Par contre, dans la deuxième situation, l'unique place et l'unique transition du RPC ont chacune respectivement une couleur pour chaque place et pour chaque transition du réseau de Pétri initial. C'est à l'informaticien de choisir le bon degré d'abstraction pour obtenir un réseau simplifié tout en lui conservant une taille suffisante pour l'analyse.

III. Définition d'un RPC. [4]

Un réseau de Pétri coloré est un quintuplet $= (P, T, C, W, M_0)$ où

- 1) P est un ensemble de places
- 2) T est un ensemble de transitions
- 3) $P \cap T = \emptyset$, $P \cup T \neq \emptyset$
- 4) $C : P \cup T \rightarrow P \text{ (CS)} - \{\emptyset\}$ est la fonction couleur
- 5) $W : P \times T \rightarrow [C(t) \rightarrow [C(p) \rightarrow Z]]_f$ est la fonction d'incidence
- 6) $M_0 : P \rightarrow [C(p) \rightarrow N]_f$ est le marquage initial

sachant que : • CS est un ensemble non vide d'éléments appelés couleurs

- $[A \rightarrow D]_f$ désigne l'ensemble des fonctions $g \in [A \rightarrow D]$ dont le support $\{a \in A \text{ tq } g(a) \neq 0\}$ est fini.

où A est un ensemble non vide

D est soit N , soit Z

rq. : pour A fini, on a $[A \rightarrow D]_f = [A \rightarrow D]$

Chaque fonction $h \in [C(t) \rightarrow [C(p) \rightarrow Z]]$ possède une extension linéaire unique dans $[[C(t) \rightarrow Z] \rightarrow [C(p) \rightarrow Z]]$ que l'on notera également h .

A chaque fonction linéaire h , nous associons une matrice $h \in Z^{c_p \times c_t}$ qui représente la transformation linéaire considérée où c_p et c_t sont les cardinaux des supports $[C(p) \rightarrow Z]$ et $[C(t) \rightarrow Z]$, respectivement. Par la suite, nous noterons de la même façon les fonctions linéaires et leur matrice associée.

La fonction d'incidence W est en fait, une matrice de fonctions. Nous pouvons ainsi considérer W comme une matrice définie par blocs. Chaque bloc $W(p_i, t_j)$ est la matrice associée à :

$$[C(t_j) \rightarrow [C(p_i) \rightarrow Z]]$$

Pour terminer, signalons que nous n'allons considérer dans ce travail que des ensembles de couleurs finis.

IV. Exemples de réseaux de Pétri colorés.

1. Réseau de bases de données. [3]

Considérons un groupe de gestionnaires de bases de données communiquant entre eux. Notons le GBD $=\{g_1, g_2, \dots, g_n\}$, $n > 0$. Chacun peut mettre à jour sa propre base de données mais doit envoyer simultanément un message à tous les autres pour les en informer. Après expédition de ces messages, il doit attendre que tous les autres aient reçu son message, réalisé la mise à jour et renvoyé une confirmation. Quand toutes ces confirmations ont été reçues par le gestionnaire à l'origine de la mise à jour, celui-ci redevient inactif. A ce moment (mais pas avant), un autre gestionnaire peut prendre le relai, réaliser une mise à jour et envoyer des messages.

Chaque gestionnaire peut être dans 3 états : "inactif", "attente" (de confirmation) et "accompli" (une mise à jour sur demande d'un autre gestionnaire).

Les gestionnaires communiquent entre eux au moyen d'un ensemble fixé de buffers, $BM = \{ \langle e, d \rangle \mid e, d \in GBD \text{ et } e \neq d \}$, où e représente l'expéditeur et d le destinataire.

Chaque buffer peut être dans 4 états différents : "inutilisé", "envoyé", "reçu" et "confirmé". Le système peut être décrit par le réseau de Petri coloré de la fig.8.

E est un ensemble ne contenant qu'un seul élément ε . Par la suite, nous écrirons souvent n au lieu de $n\varepsilon$ où $n \in \mathbb{N}$.

Intuitivement, ε représente des marques sans couleur. Pour tout ensemble de couleurs C , nous définissons $ABS \in [C \rightarrow [E \rightarrow Z]_f]$ et $ID \in [C \rightarrow [C \rightarrow Z]_f]$ par

$$\forall c \in C : [ABS(c) = \varepsilon \text{ et } ID(c) = c]$$

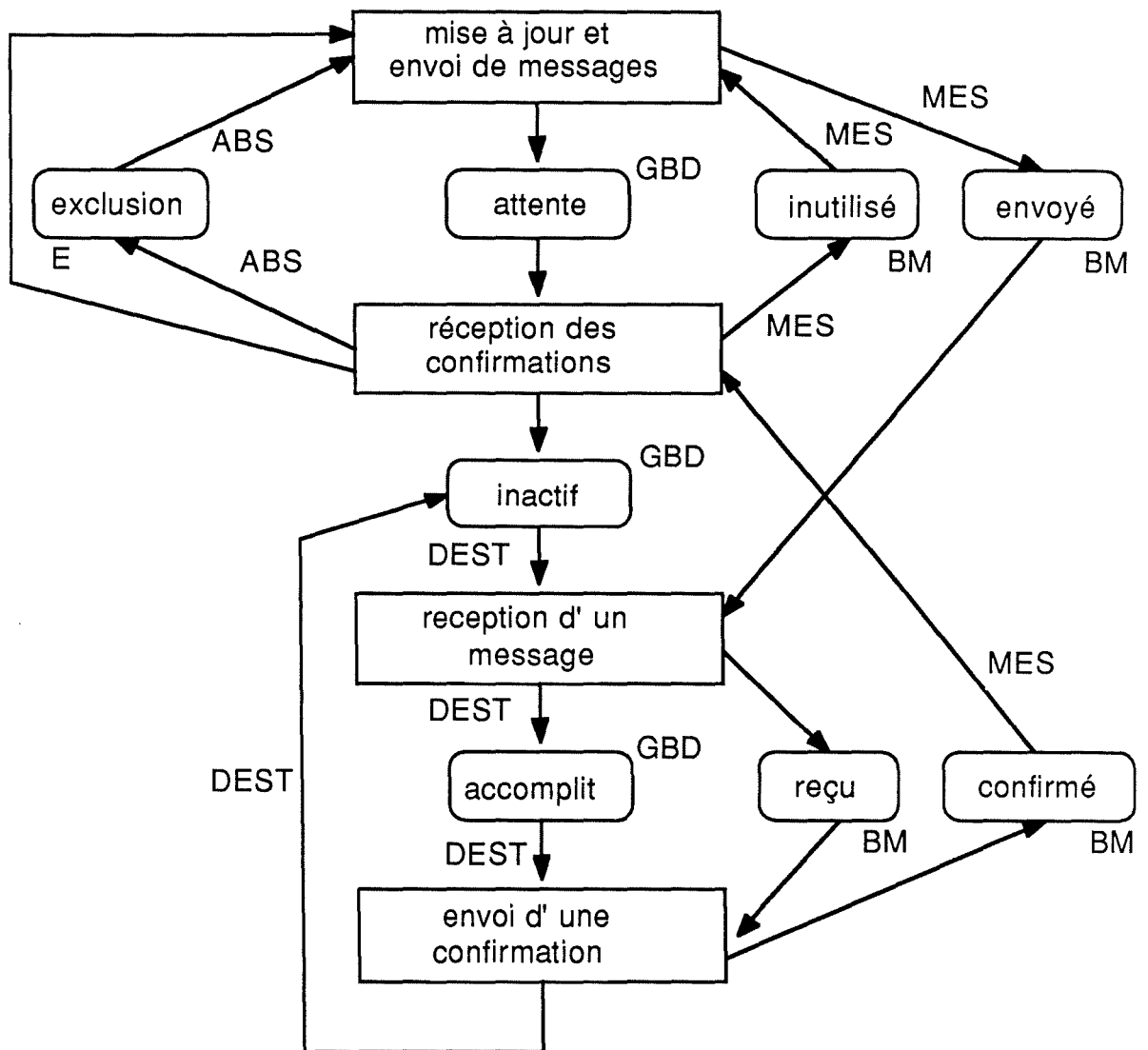


fig. 8 : RPC schématisant le réseau de bases de données

En fait, (l'extension linéaire de) ABS compte le nombre de marques dans son argument en ignorant leur couleur. ABS (x) sera souvent écrit $|x|$. Par exemple,

$$\text{ABS}(3u - v + 2w) = |3u - v + 2w| = 4\varepsilon = 4 \text{ pour } u, v, w \in C.$$

Les fonctions DEST et MES sont définies par :

$$\forall \langle e, d \rangle \in \text{BM} \quad [\text{DEST}(\langle e, d \rangle) = d]$$

$$\forall e \in \text{GBD} \quad [\text{MES}(e) = \sum \langle e, d \rangle]$$

Le marquage initial est le suivant :

$$M_0(\text{inactif}) = \sum \text{GBD}$$

$$M_0(\text{inutilisé}) = \sum \text{BM}$$

$$M_0(\text{exclusion}) = 1$$

et aucune autre place n'est marquée.

La matrice d'incidence est donnée à la fig. 9.

		mise à jour et envoi de messages	réception des confirmations	réception d'un message	envoi d'une confirmation
		GBD	GBD	BM	BM
inactif	GBD	-ID	ID	-DEST	DEST
attente	GBD	ID	-ID		
accomplit	GBD			DEST	-DEST
exclusion	E	- ABS	ABS		
inutilisé	BM	- MES	MES		
envoyé	BM	MES		-ID	
reçu	BM			ID	-ID
confirmé	BM		- MES		ID

fig. 9 : matrice d'incidence associée au RPC de la fig. 8

2. Le système du téléphone. [5]

Le réseau de Pétri coloré de la fig. 10 schématise le comportement des téléphones individuels, vu par les utilisateurs.

U est un ensemble de couleurs qui représente les différents numéros de téléphone. Dans l'expression $V = U \times U$, la première composante correspond au téléphone appelant alors que la deuxième symbolise le téléphone appelé. P1 et P2 sont les deux projections de V sur sa première et deuxième composantes respectivement.

En décrochant le combiné téléphonique, on entend un signal continu ; ceci correspond au passage de l'état "inactif" à l'état "continu". En composant le numéro, le signal s'interrompt et le téléphone passe dans l'état "sans signal". Ceci peut conduire à deux situations différentes : ou bien le téléphone appelé est occupé et on entend un signal "court" sur l'autre combiné ; ou bien il est inoccupé, il sonne et un signal "long" se fait entendre chez l'abonné qui a formé le numéro.

Dans le premier cas, l'utilisateur appelant pourra raccrocher le combiné ; le téléphone repassera ainsi dans l'état "inactif". Dans le second cas, ou bien la personne appelée est absente et le téléphone appelant peut redevenir "inactif", ou bien elle décroche son combiné. Les téléphones sont alors "connectés" jusqu'au moment où le premier abonné coupe la communication et met son téléphone dans l'état "inactif". Le téléphone appelé se retrouve alors "déconnecté" et peut être raccroché afin de redevenir "inactif".

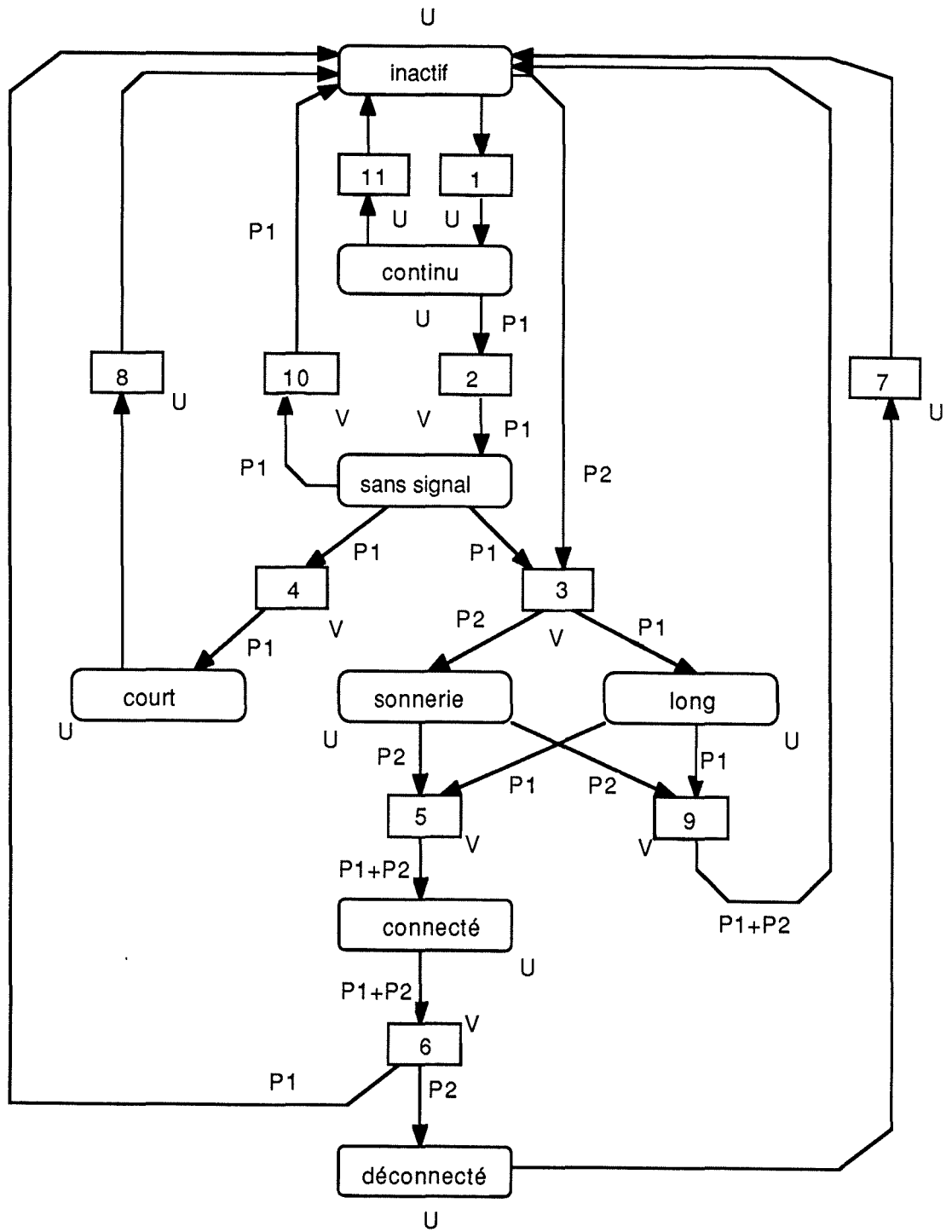


fig. 10 : RPC modélisant le système du téléphone

Le réseau de Pétri de la fig. 10 décrit le comportement des téléphones individuels mais en dit assez peu à propos de leur synchronisation. Celle-ci est décrite par le RPC de la fig. 11 qui représente l'échange entre les téléphones.

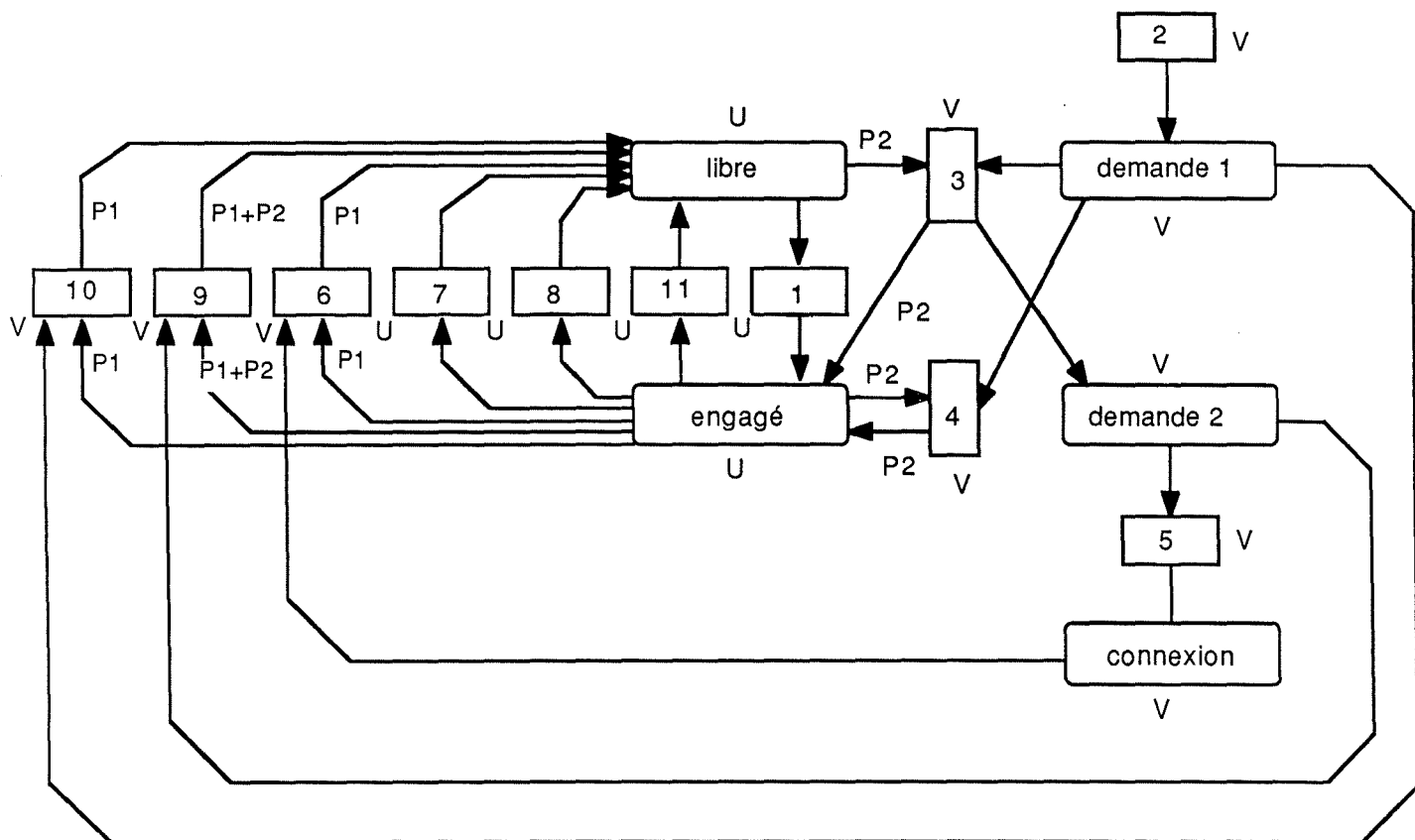


fig. 11 : RPC modélisant la synchronisation dans le système du téléphone

Dans ce réseau, chaque téléphone peut être "libre" ou "engagé". Une paire $v = (u_1, u_2)$ de V est dans l'état "demande 1" si le numéro u_2 a été formé sur le combiné correspondant au numéro u_1 .

Si le combiné numéroté u_2 est "libre", l'appel (u_1, u_2) peut progresser jusqu'à l'état "demande 2" et lorsque l'utilisateur appelé décroche son téléphone, il y a "connexion" entre u_1 et u_2 .

Le système du téléphone se schématise donc par la fig. 11 superposée à la fig. 10.

Son marquage initial est le suivant :

$$M_0(\text{inactif}) = M_0(\text{libre}) = \sum U,$$

les autres places ne sont pas marquées.

Avant d'examiner sa matrice d'incidence, on peut observer que le réseau de la fig. 11 comporte une "sidecondition" (la transition 4 peut seulement être franchie si le téléphone appelé est "engagé", mais le tir de la transition 4 ne change pas le marquage de la place "engagé"). Cette condition spéciale n'est pas reflétée dans la matrice d'incidence, puisqu'aucune marque n'a été ajoutée ou enlevée par ce franchissement. Cette observation nous amène à conclure qu'il n'existe pas une correspondance bijective entre les matrices d'incidence et les réseaux de Pétri colorés munis de "sideconditions". Cette matrice d'incidence est représentée ci-dessous.

		1	2	3	4	5	6	7	8	9	10	11
		U	V	V	V	V	V	U	U	V	V	U
inactif continu sans signal	U	-ID		-P2			P1	ID	ID	P1+P2	P1	ID
	U	ID	-P1									-ID
	U		P1	-P1	-P1						-P1	
long court sonnerie	U			P1		-P1				-P1		
	U				P1				-ID			
connecté déconnecté	U					P1+P2	-P1-P2					
	U						P2	-ID				
libre	U	-ID		-P2			P1	ID	ID	P1+P2	P1	ID
	U											
engagé demande 1	U	ID		P2			-P1	-ID	-ID	-P1-P2	-P1	-ID
	V		ID	-ID	-ID						-ID	
demande 2	V			ID		-ID				-ID		
	V					ID	-ID		-ID			
connexion	V								-ID			

fig. 12 : matrice d'incidence correspondant au système du téléphone

CHAPITRE 2 - L'ALGÈBRE LINÉAIRE.

I. Introduction.

Plusieurs voies d'analyse ont été exploitées dans le cadre des réseaux de Pétri.

L'idée la plus naturelle est de construire le graphe des marquages du réseau. Si l'ensemble des marquages accessibles est fini, on pourra directement déduire de ce graphe certaines propriétés du réseau. Lorsque cet ensemble est infini, cela devient plus complexe mais une astuce permet de fournir des renseignements importants sur le réseau.

Cependant, la construction de cet arbre ne résout pas tous les problèmes d'analyse des réseaux. En effet, l'artifice utilisé apporte avec lui une perte d'informations. Il arrive aussi que le nombre de marquages accessibles d'un réseau (même borné) soit trop grand pour que cette construction soit utilisable.

De plus, même si la complexité des algorithmes basés sur cette méthode est difficile à déterminer exactement, il est possible d'en donner une borne inférieure. On peut montrer que ce problème présente un coût exponentiel en temps et en espace. Ainsi, le temps et l'espace mémoire requis se traduisent par une fonction exponentielle de la taille du réseau de Pétri [6].

Certains outils dérivés de la théorie des graphes peuvent également être utiles pour étudier les réseaux de Pétri.

Nous allons explorer dans ce chapitre, une autre voie d'analyse qui est celle de l'algèbre linéaire. Celle-ci a rendu possible l'étude des propriétés "structurelles" du réseau, c'est-à-dire indépendantes du marquage et a donné, également, quelques résultats dépendants du marquage.

Dans ce cadre, nous allons développer la méthode, dite de l'invariant selon deux approches différentes. La première, introduite par K. Jensen [5] a pour but de simplifier les calculs à l'aide de quatre règles de transformation. La seconde, proposée par M. Silva et al. [4] se base sur le concept de fonction inverse généralisée.

II. Les réseaux de Pétri.

Définitions.

Soit (R, M) un réseau marqué avec $R = (P, T, W, M_0)$

- Un ensemble pondéré de transitions est une fonction $x \in [T \rightarrow Z]$. Il est positif ssi $x(t) \geq 0 \forall t \in T$ et $x(t) > 0$ pour au moins une fonction $x \in [T \rightarrow Z]$.
- Un ensemble pondéré de places est une fonction de $[P \rightarrow Z]$.
- Si un marquage M' est accessible à partir d'un autre marquage M alors il existe un ensemble pondéré de transitions x tq

$$M' = M + W * x$$

Remarque : '*' et '+' représentent respectivement la multiplication et l'addition matricielles.

On peut alors introduire la méthode de l'invariant à l'aide du théorème suivant [3].

Théorème 1 : Soit v un ensemble pondéré de places. Si $v * W = 0$ alors $v * M' = v * M$ pour tous marquages M et M' où M' est accessible à partir de M .

On dit alors que v est un invariant.

<u>dém.</u> : $v * M' = v * (M + W * x)$	(M' est accessible à partir de M)
$= v * M + v * (W * x)$	(distributivité de * par rapport à +)
$= v * M + (v * W) * x$	(associativité de *)
$= v * M$	(par hypothèse)

Remarque : comme toute combinaison linéaire d'invariants est aussi un invariant, on peut en construire un grand nombre.

Chaque invariant conduit à une équation satisfaite par tous les marquages accessibles à partir du marquage initial. On peut ainsi tirer une série de propriétés du système comme le montre l'exemple simple des lecteurs-rédacteurs.

Il s'agit de n processus ($n > 0$) qui peuvent lire et écrire dans une mémoire partagée sous certaines contraintes. Premièrement, plusieurs processus peuvent

lire simultanément mais lorsqu'un processus écrit à un endroit précis de la mémoire, aucun autre ne peut lire, ni écrire. Deuxièmement, il n'existe pas de priorité entre les opérations d'écriture et de lecture.

Chaque processus peut être dans cinq états différents :

- TL : le processus travaille localement et n'utilise pas la mémoire partagée
- AL : le processus est en attente de lecture dans la mémoire partagée
- AE : le processus est en attente d'écriture dans la mémoire partagée
- L : le processus lit dans la mémoire partagée
- E : le processus écrit dans la mémoire partagée.

La place S (synchronisation) assure l'exclusion mutuelle entre un rédacteur et un lecteur ou un autre rédacteur.

La représentation graphique du réseau de Pétri qui modélise ce système, sa matrice d'incidence et les trois invariants qui en découlent sont donnés ci-dessous :

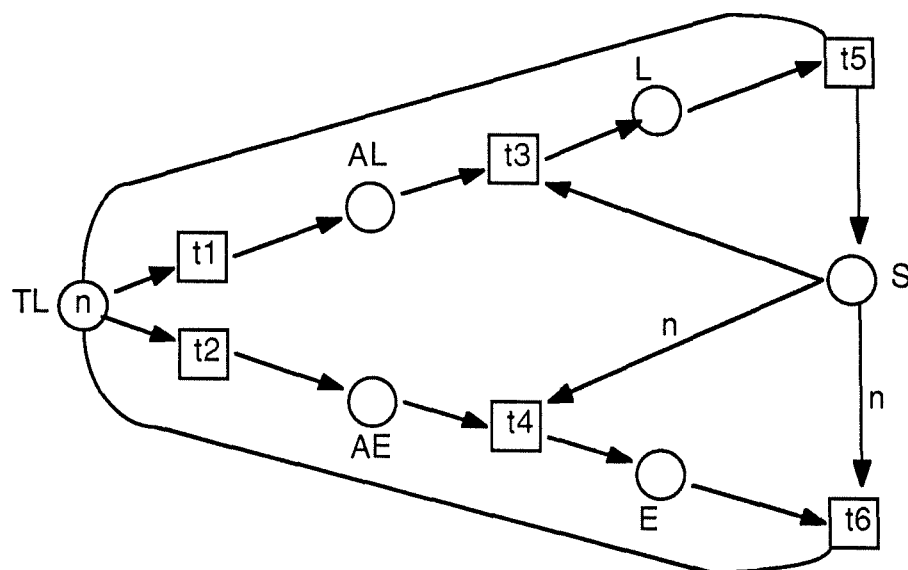


fig. 13 : réseau de Pétri modélisant le système des lecteurs-rédacteurs

P \ T	T						m0	invariants		
	t1	t2	t3	t4	t5	t6		i1	i2	i3
TL	-1	-1			1	1	n	1		-1
AL	1		-1					1		-1
AE		1		-1				1		-1
L			1		-1			1	1	
E				1		-1		1	n	n-1
S			-1	-n	1	n	n		1	1

fig. 14 : matrice d'incidence et invariants associés au réseau de Pétri de la fig. 13

A partir de i_1 , on peut établir :

$$M(TL) + M(AL) + M(AE) + M(L) + M(E) = n$$

Cette équation nous permet d'affirmer que le nombre de processus est constant.

Le deuxième invariant nous conduit à :

$$M(L) + n \cdot M(E) + M(S) = n$$

Ainsi, si un processus écrit, aucun autre ne peut lire, ni écrire. De plus, le nombre de lecteurs est compris entre 0 et n et s'il n'y a aucun processus lecteur ou rédacteur, $M(S) = n$. La transition t_3 pourra donc être franchie, s'il y a au moins un processus en attente de lecture (une marque dans la place AL) et la transition t_4 pourra l'être à condition qu'il y ait au moins un processus en attente d'écriture (une marque dans la place AE).

Le troisième invariant, qui est une combinaison linéaire des deux premiers

$$M(TL) + M(AL) + M(AE) = (n - 1) \cdot M(E) + M(S)$$

nous apprend que $M(AL) \leq M(S)$ si aucun processus n'écrit. Donc, la transition t_3 pourra être tirée si au moins un processus est en attente de lecture.

On peut finalement déduire, à l'aide de ces équations, que le réseau de Pétri considéré est sans blocage. En effet,

si $M(TL) + M(L) + M(E) > 0$, les transitions t_1, t_2, t_5 ou t_6 peuvent être franchies.

Par contre,

si $M(TL) + M(L) + M(E) = 0$, on peut voir à partir des invariants i_1 et i_2 que $M(AL) + M(AE) = n$, $M(S) = n$ donc que t_3 et t_4 sont franchissables.

Ces quelques déductions ont montré à quel point les invariants pouvaient être d'un grand intérêt dans l'analyse des réseaux de Pétri.

Des algorithmes existent pour extraire les invariants d'un réseau de Pétri, notamment par la triangulation de la matrice d'incidence.

III. Les réseaux de Pétri colorés.

1. Introduction

Kurt Jensen [3] a étendu cette méthode en remplaçant les matrices d'entiers par des matrices de fonctions linéaires entre les ensembles de couleurs. Pour cela, il a dû généraliser la multiplication matricielle en substituant à chaque produit d'entiers une composition ou une application de fonctions.

Définition.

Soient $a = (a_{ij})_{1 \leq i \leq r, 1 \leq j \leq s}$ et $b = (b_{jk})_{1 \leq j \leq s, 1 \leq k \leq t}$ deux matrices.

On définit $a * b = (c_{ik})_{1 \leq i \leq r, 1 \leq k \leq t}$ par

$$c_{ik} = \sum a_{ij} b_{jk} \quad \forall i \in \{1, \dots, r\} \text{ et } \forall k \in \{1, \dots, t\}$$

où la juxtaposition $a_{ij} b_{jk}$ représente la composition ou l'application de fonctions.

On obtient ainsi de nouvelles définitions.

Définitions.

- Un marquage M' est accessible à partir d'un autre marquage ssi il existe un ensemble pondéré de transitions x tq

$$M' = M + W * x$$

- Soit un ensemble non vide. Un ensemble pondéré de places (par rapport à U) est une fonction v définie sur P tq

$$v(P) \in [C(p) \rightarrow [U \rightarrow Z]_f] \quad \forall p \in P$$

On peut alors généraliser le théorème 1 aux RPC.

Théorème 2 : Soit v un ensemble pondéré de places (par rapport à U non vide) et $O = (O_j)_{1 \leq j \leq m}$ une matrice de fonctions nulles

$$O_j \in [[C(t_j) \rightarrow Z]_f \rightarrow [U \rightarrow Z]_f].$$

Si $v * W = O$ alors

$$v * M' = v * M \quad \forall M, M'$$

où M' est accessible à partir de M .

On dit alors que v est un invariant.

Montrons maintenant par un exemple l'aide apportée par cette technique à l'étude des RPC.

2. Exemple : réseau de bases de données.

La matrice d'incidence de ce réseau ainsi que ses invariants sont montrés à la fig. 15.

		mise à jour et envoi de message				m_0	invariants								
		réception des confirmations		réception d'un message			envoi d'une confirmation		i1	i2	i3	i4	i5	i6	i7
		GBD	GBD	BM	BM		GBD	BM	E	GBD	BM	GBD	E		
Inactif	GBD	-ID	ID	-DEST	DEST	Σ GBD	ID								
attente	GBD	ID	-ID				ID		ABS		MES	(n-1)ID			
accomplit	GBD			DEST	-DEST		ID			ID		-ID	ABS		
exclusion	E	-ABS	ABS			1			ID				(n-1)ID		
inutilisé	BM	-MES	MES			Σ BM		ID							
envoyé	BM	MES		-ID				ID			-ID	-DEST	ABS		
reçu	BM			ID	-ID			ID		-DEST	-ID				
confirmé	BM		-MES		ID			ID			-ID	-DEST	ABS		

fig. 15 : matrice d'incidence et invariants liés au réseau de bases de données

Pour le moment, ne tenons pas compte des invariants (i6) et (i7).

A partir de (i1) qui peut se réécrire :

$$M(\text{inactif}) + M(\text{attente}) + M(\text{accomplit}) = \Sigma \text{GBD}$$

on conclut que chaque gestionnaire ne peut être que dans un des trois états précités.

A partir de (i2)

$$M(\text{inutilisé}) + M(\text{envoyé}) + M(\text{reçu}) + M(\text{confirmé}) = \Sigma \text{BM}$$

on déduit que chaque buffer ne peut être que dans un des quatre états cités ci-dessus.

A partir de (i3)

$$| M(\text{attente}) | + M(\text{exclusion}) = 1$$

on tire qu'un gestionnaire au plus peut être dans l'état "attente".

L'invariant (i4) qui se réécrit

$$M(\text{accompli}) = \text{DEST}(M(\text{reçu}))$$

signifie qu'un gestionnaire accomplit une mise à jour si et seulement si il y a un message qui lui est adressé i.e. que le buffer est dans l'état "reçu".

La relation tirée de (i5)

$$\text{MES}(M(\text{attente})) = M(\text{envoyé}) + M(\text{reçu}) + M(\text{confirmé})$$

impose à tous les buffers d'un gestionnaire "en attente" d'être "envoyé", "reçu" ou "confirmé" (donc, aucun ne peut être "inutilisé"). De plus, quand il n'est pas "en attente", aucun de ses buffers n'est "envoyé", "reçu" ou "confirmé" (ils sont tous "inutilisés").

A côté de toutes ces caractéristiques, il est possible de déduire de ces invariants que ce RPC est sans blocage. Pour cela notons M un marquage accessible à partir du marquage initial. Plusieurs cas sont à considérer.

- Prenons comme première hypothèse un gestionnaire g C GBD en "attente" dans M. D'après (i5)

$$\text{MES}(M(\text{attente})) = M(\text{envoyé}) + M(\text{reçu}) + M(\text{confirmé})$$

on peut mettre en évidence que les buffers de g sont tous dans un des trois états "envoyé", "reçu" ou "confirmé".

- Considérons d'abord le cas où au moins un de ses buffers <g, d> C BM est dans l'état "reçu". La relation tirée de (i4)

$$M(\text{accompli}) = \text{DEST}(M(\text{reçu}))$$

permet de conclure que d "accomplit" une mise à jour. La transition "envoi d'une confirmation" est ainsi activée (par rapport à la couleur d).

- Imaginons ensuite qu'au moins un de ses buffers <g, d> C BM soit dans l'état "envoyé". Dans ce cas, l'invariant (i3)

$$| M(\text{attente}) | + M(\text{exclusion}) = 1$$

interdit au gestionnaire d (d ≠ g) d'être en "attente". D'après (i1)

$$M(\text{inactif}) + M(\text{attente}) + M(\text{accomplit}) = \sum \text{GBD}$$

il ne peut être qu'"inactif" ou dans l'état "accomplit". S'il accomplit une mise à jour, nous en concluons d'après (i4)

$$M(\text{accomplit}) = \text{DEST}(M(\text{reçu}))$$

qu'il existe un buffer <e , d> C BM qui est "reçu". D'autre part, la relation tirée de (i5)

$$\text{MES}(M(\text{attente})) = M(\text{envoyé}) + M(\text{reçu}) + M(\text{confirmé})$$

affirme que e est en "attente". Ainsi, d'après (i3)

$$| M(\text{attente}) | + M(\text{exclusion}) = 1$$

e doit se confondre avec g. Cela signifie que le buffer <g , d> est à la fois "envoyé" et "reçu", ce qui est en contradiction avec l'invariant (i2) qui impose

$$M(\text{inutilisé}) + M(\text{envoyé}) + M(\text{reçu}) + M(\text{confirmé}) = \sum \text{BM}$$

On déduit ainsi de (i1)

$$M(\text{inactif}) + M(\text{attente}) + M(\text{accomplit}) = \sum \text{GBD}$$

que d doit être "inactif" et que la transition "réception d'un message" est activée (par rapport à la couleur d).

- Il reste à considérer la situation où tous les buffers de g sont dans l'état "confirmé". Dans ce cas, c'est la transition "réception d'une confirmation" qui est franchissable (par rapport à la couleur g).

- Supposons maintenant qu'un gestionnaire g C GBD soit dans l'état "accomplit" dans M, la relation issue de (i4)

$$M(\text{accompli}) = \text{DEST } (M(\text{reçu}))$$

oblige un de ses buffers à être dans l'état "reçu", ce qui signifie que la transition "envoi d'une confirmation" est activée (par rapport à la couleur g).

- Supposons finalement qu'aucun gestionnaire ne soit en "attente" ou dans l'état "accompli" dans le marquage M. Il résulte de (i1)

$$M(\text{inactif}) + M(\text{attente}) + M(\text{accompli}) = \sum \text{GBD}$$

que tous les gestionnaires sont "inactifs". L'invariant (i3)

$$| M(\text{attente}) | + M(\text{exclusion}) = 1$$

impose à la place "exclusion" d'être marquée et les deux relations

$$\text{MES}(M(\text{attente})) = M(\text{envoyé}) + M(\text{reçu}) + M(\text{confirmé})$$

$$M(\text{inutilisé}) + M(\text{envoyé}) + M(\text{reçu}) + M(\text{confirmé}) = \sum \text{BM}$$

nous indiquent que tous les buffers sont "inutilisés". Nous concluons alors que la transition "mise à jour et envoi de messages" est activée (par rapport à n'importe quelle couleur de GBD).

Ce raisonnement nous prouve qu'il est toujours possible de franchir une transition du réseau et cela quelle que soit la situation. Nous avons ainsi démontré que ce RPC est un réseau sans blocage.

IV. Méthode de recherche des invariants d'un RPC.

L'exemple précédent a mis en évidence l'intérêt de la méthode de l'invariant appliquée aux RPC. Il reste à trouver un moyen qui permette de les calculer. Kurt Jensen [5] a proposé quatre transformations qui simplifient la matrice d'incidence d'un RPC tout en lui conservant le même ensemble d'invariants.

Sa méthode est basée sur les caractéristiques propres aux matrices d'incidence des RPC. Celles-ci sont, en général, peu denses, non carrées et possèdent une forte dépendance entre ses colonnes. De plus, beaucoup d'éléments matriciels sont des fonctions commutatives (la fonction identité, par exemple) et il existe plusieurs solutions à l'équation matricielle homogène. Les transformations vont bénéficier de ces propriétés et ne seront donc pas recommandées pour d'autres types de matrices.

1. Règles de transformation.

Nous allons définir quatre règles de transformation et prouver leur "justesse" c'est-à-dire leur capacité à conserver intact l'ensemble des invariants du système.

Nous allons travailler avec des matrices un peu plus générales, où chaque ligne peut être associée à un ensemble de places. Chaque place est liée à au plus, une ligne et possède un facteur de poids permettant de passer des solutions de l'équation matricielle homogène aux invariants.

Définitions.

- Soit P l'ensemble des places d'un RPC. Définissons pour chaque place $p \in P$ un ensemble de couleurs $C(p)$ et un domaine d'application $D(p) = [C(p) \rightarrow Z]_f$. Une matrice (avec des places $p \in P$ et des facteurs de poids associés) est bien formée (par rapport à P) ssi elle possède les propriétés suivantes :
 - tous ses éléments sont des fonctions linéaires
 - chaque colonne est associée à un ensemble non vide C et chaque élément de la colonne a $[C \rightarrow Z]_f$ pour domaine.
 - chaque ligne est reliée à un ensemble non vide C et chaque élément de la ligne a $[C \rightarrow Z]_f$ comme codomaine.
 - toutes les places associées aux lignes sont des éléments de P
 - chaque place est attachée à une ligne au plus
 - chaque place p reliée à une ligne avec le codomaine D a un facteur de poids qui est une fonction linéaire de $D(p)$ vers P .
- Le domaine (codomaine) d'une colonne (ligne) d'une matrice bien formée est définie comme le domaine (codomaine) de ses éléments.
- Pour chaque RPC, la matrice d'incidence est bien formée par rapport à l'ensemble des places du réseau.

- Soit u une solution de l'équation matricielle homogène

$$u * W = 0$$

d'une matrice bien formée par rapport à un ensemble de places P . L'invariant v correspondant est défini par :

$$\forall p \in P, v(p) = \begin{cases} u(r) \circ w & \text{si } p \text{ est associée à la ligne } r \text{ avec le facteur de poids } w \\ 0 & \text{si } p \text{ n'est associée à aucune ligne} \end{cases}$$

On dit qu'un invariant couvre une place p si le poids $v(p)$ diffère de la fonction nulle 0.

- Une fonction de la forme $h \in [[A \rightarrow Z]_f \rightarrow [B \rightarrow Z]_f]$ est pseudosurjective ssi $\forall b \in B$, il existe $g \in [A \rightarrow Z]_f$ et il existe $z \in Z - \{0\}$ tq $[h(g) = zb]$.

La surjectivité implique la pseudosurjectivité.

A chaque étape de transformation, nous supposons que la matrice est bien formée et de la forme $W = (W_{ij})_{1 \leq i \leq n, 1 \leq j \leq m}$ où $1 \leq n, m < \infty$.

Transformation 1 : si la colonne j de la matrice (i.e. les fonctions $(W_{ij})_{1 \leq i \leq n}$) a le domaine A_j , que D est un ensemble non vide, que $A'_j = [D \rightarrow Z]_f$ et que $h \in [A_j \rightarrow A'_j]$ est une fonction pseudosurjective linéaire, remplacez la colonne j par $(W_{ij} \circ h)_{1 \leq i \leq n}$ et le domaine A_j par A'_j .

Transformation 2 : si deux colonnes de la matrice j et k ($j \neq k$) possèdent respectivement les domaines A_j et A_k et que $h \in [A_j \rightarrow A_k]$ est une fonction linéaire, remplacez la colonne j par $(W_{ij} + W_{ik} \circ h)_{1 \leq i \leq n}$.

Transformation 3 : si tous les éléments de la colonne j de la matrice sont des fonctions nulles, supprimez la colonne j .

Transformation 4 : si tous les éléments de la colonne j de la matrice sont des fonctions nulles sauf deux éléments différents W_{ij} et W_{kj} qui vérifient $W_{ij} = h$ et $W_{kj} = -g \circ h$, où g et h sont des fonctions linéaires (h pseudosurjective), remplacez la ligne k par $(g \circ W_{ij} + W_{kj})_{1 \leq j \leq m}$.

Pour chaque place p liée à la ligne i avec le facteur de poids w , associez à p un facteur de poids $w \circ g$. Si $w \circ g$ diffère de la fonction nulle, associez p à la ligne k .

Supprimez la ligne i et la colonne j .

Les transformations 1 et 2 peuvent être utilisées pour simplifier les éléments de la matrice mais ne modifient pas sa taille. Seules les transformations 3 et 4 permettent de la réduire.

Définition.

Une matrice W' (avec des domaines, codomains, places et facteurs de poids associés) est engendable à partir d'une matrice d'incidence W d'un RPC ssi il existe une séquence de transformations de types 1 - 4, qui engendrent W' à partir de W .

Les quatre règles sont indépendantes dans le sens où l'omission de l'une d'elles risque de réduire l'ensemble des matrices engendrables.

Proposition 1.

Toute matrice engendable à partir d'une matrice d'incidence avec l'ensemble de places P est bien formée par rapport à P .

dém. : il suffit de vérifier que chaque transformation conserve cette propriété.

Théorème 3 (justesse) : soit W' une matrice engendable à partir d'une matrice W d'un RPC. Alors W et W' ont exactement le même ensemble d'invariants.

dém. : pour chaque transformation, nous allons démontrer que son application ne modifie pas l'ensemble des invariants.

transformation 1 : nous allons prouver que :

$$\sum v_i \circ w_{ij} = O_1 \iff \sum v_i \circ (W_{ij} \circ h) = O_2$$

où O_1 et O_2 sont les fonctions nulles sur les domaines A_j et A'_j respectivement. Comme W_{ij} et h sont des fonctions linéaires, nous avons :

$$(\sum v_i \circ (W_{ij} \circ h))(g) = (\sum v_i \circ W_{ij})(h(g')) \quad \forall g' \in A'_j$$

Il suffit donc de montrer que :

$$\forall g \in A_j \quad [(\sum v_i \circ W_{ij})(g) = 0] \Leftrightarrow \forall g' \in A'_j \quad [(\sum v_i \circ W_{ij})h(g') = 0]$$

Or, \Rightarrow car h est une fonction

\Leftarrow à cause de la pseudosurjectivité de h et de la linéarité des fonctions.

Transformation 2 : comme les fonctions sont linéaires, nous avons :

$$\sum v_i \circ W_{ij} = 0 \quad \text{et} \quad \sum v_i \circ W_{ik} = 0$$

\Leftrightarrow

$$\sum v_i \circ (W_{ij} + W_{ik} \circ h) = 0 \quad \text{et} \quad \sum v_i \circ W_{ik} = 0$$

Transformation 3 : la colonne j correspond à une équation qui est toujours satisfaite. Elle peut donc être supprimée sans modifier l'ensemble des invariants.

Transformation 4 : la colonne j correspond à une équation dont toute solution doit vérifier $u(i) = u(k) \circ g$. Dans ce cas, la linéarité des fonctions impliquées nous permet de combiner les deux lignes i et k sans changer le système d'équations. La colonne j peut alors être supprimée puisque l'équation correspondante sera toujours satisfaite (en modifiant les facteurs de poids).

Ces quatre règles de transformation ont été inspirées de la méthode d'élimination de Gauss qui est utilisée pour des matrices dont tous les éléments appartiennent à un champ. Malheureusement, les éléments matriciels, dans le cas des RPC, font seulement partie d'un anneau non commutatif. Ainsi, la division de deux éléments peut être impossible. Dans cette situation, aucun algorithme général n'est connu pour résoudre des équations matricielles homogènes. Cela signifie qu'en général, il n'est pas possible de trouver tous les invariants à l'aide de ces règles. Présentons néanmoins deux exemples où cette technique est parfaitement applicable.

2. Exemples.

2.1 Réseau de bases de données.

La fig. 16 est constituée de quatre matrices m_0 , m_1 , m_2 et m_3 . La matrice m_0 coïncide avec la matrice d'incidence du réseau de bases de données tandis que

m_1 , m_2 et m_3 correspondent aux matrices obtenues successivement à partir de m_0 , à l'aide des règles de transformation.

Chacune d'elles est composée des lignes et des colonnes indiquées par une astérisque. m_0 , par exemple, est formée des colonnes c_1 , c_2 , c_3 et c_4 et des huit premières lignes du tableau.

		c1	c2	c3	c4	c5	c6	
		DBM	DBM	MB	MB	DBM	MB	
m0		*	*	*	*	T2 c1+c2	1	
m1		T2 c5	*	*	*	*	T2 c3+c4	
m2		*	T2 c6	2	*	*	*	
m3		*	*	*	T3 4	T4 3		
inactif	DBM	*	*	*	-ID	ID	-DEST	DEST
attente	DBM	*	*	*	ID	-ID	DEST	-DEST
accompli	DBM	*	*	*	*	*	*	*
exclusion	E	*	*	*	-ABS	ABS		
inutilisé	MB	*	*	*	-MES	MES		
confirmé	MB	*	*	*	MES		-ID	MES
reçu	MB	*	*	*			ID	-ID
envoyé	MB	*	*	*		-MES	ID	-MES
envoyé,confirmé	MB	*	*	*		-MES	ID	ID

fig. 16 : transformation de la matrice d'incidence du réseau de bases de données

La création ou la disparition d'une ligne ou d'une colonne lors d'une transformation est indiquée par l'identifiant de la règle choisie. Ces identifiants sont indicés par un numéro indiquant l'ordre dans lequel les transformations ont été appliquées.

De plus, la création d'une nouvelle colonne à partir de deux autres est rappelée dans les notations et sa suppression est accompagnée d'un numéro indiquant celle qu'elle a contribué à concevoir.

Cette représentation, bien que compacte, comporte l'intérêt de visualiser l'évolution de la matrice d'incidence sur un seul tableau.

La matrice m_3 ainsi obtenue est largement simplifiée et il devient facile de trouver les cinq solutions à l'équation matricielle homogène correspondante. Cette matrice est reprise à la fig. 17 accompagnée de ses cinq invariants qui ont été exploités au paragraphe précédent.

m 3		c2	c4	m 0	l1	l2	l3	l4	l5
		DBM	MB		DBM	MB	E	DBM	MB
inactif	DBM	ID	DEST	ΣDBM	ID				
attente	DBM	-ID			ID		ABS		MES
accompli	DBM		-DEST		ID			ID	
exclusion	E	ABS		1			ID		
inutilisé	MB	MES		ΣMB		ID			
envoyé, confirmé	MB	-MES	ID			ID			-ID
reçu	MB		-ID			ID		-DEST	-ID

fig. 17 : matrice d'incidence et invariants associés

2.2 Le système du téléphone.

Comme nous l'avons dit, ce réseau comporte une "sidecondition" qui ne se reflète pas dans la matrice d'incidence. Puisque les invariants sont définis à partir de cette matrice, on ne peut rien retirer à propos des propriétés du système liées à cette "sidecondition".

Il reste cependant quelques invariants intéressants à partir desquels nous allons pouvoir tirer un certain nombre de conclusions concernant le réseau étudié.

La matrice d'incidence ainsi que ses transformations successives sont présentés aux fig. 18 et fig. 19. Les mêmes notations que celles décrites pour l'exemple précédent ont été utilisées.

		c1	c2	c3	c4	c5	c6	c7	c8	c9	c10	c11	c12	c13	
		U	V	V	V	V	U	U	U	V	V	U	V	U	
m 0		*	*	*	*	*	*	*	*	*	*	*	T_2^1	c_2+c_4	c_7-c_8
m 1		*	T_2^1	*	*	*	T_2^2	*	*	*	*	LC_3	*	*	
m 2		*	*	*	*	*	*	*	*	*	*	*	T_4^4	T_4^4	
inactif	U	*	*	*	-ID	-P1	-P2	P1	ID	ID	P1+P2	P1	ID	-ID	
continu	U	*	*	*	ID	-P1	-P2						-ID	-P1	
sans signal	U	*	*	T_4^4		P1	-P1					-P1			
long	U	*	*	*		P1	-P1								
court	U	*	*	T_4^4		P1	-P1		-ID				P1	ID	
sonnerie	U	*	*	*		P2	-P2			-P2					
connecté	U	*	*	*		P1+P2	-P1-P2								
déconnecté	U	*	*	T_4^4		P2	-P2	-ID	ID					-ID	
libre	U	*	*	*	-ID	-P2		P1	ID	ID	P1+P2	P1	ID		
engagé	U	*	*	*	ID	P2	-P2	-P1	-ID	-ID	-P1-P2	-P1	-ID		
demande 1	V	*	*	*	ID	-ID	-ID								
demande 2	V	*	*	*	ID	-ID	-ID				-ID	-ID			
connexion	V	*	*	*		ID	-ID								
cont.court,	U	*	*	T_4^4	ID		P1	P2		-ID					
déconnecté	U	*	*	*											

fig. 18 : transformations de la matrice d'incidence liée au système du téléphone (I)

		c1	c3	c4	c5	c6	c8	c9	c10	c14	c15
		U	V	V	V	V	U	V	V	V	V
	m2	*	*	*	*	*	*	*	*	*	*
	m3	*	LC ₅ c10-c9	*	*	*	LC ₈ -c1	LC ₇ c5+c6 c1, P2	LC ₈ c4, c1, P1	T2 ₉ c6 + c1, P1	T2 ₁₀ c5+c14
	m4	*	*	*	T2 ₁₀ c15	T2 ₉ c14	*	*	*	*	*
inactif	U	-ID	-P2			P1	D	P1+P2	P1		
cont, court, déconn	U	D	P1			P2	-ID			P1+P2	P1+P2
sans signal	U		-P1	-P1						-P1	
long	U		P1		-P1			-P1			-P1
sonnerie	U		P2		-P2			-P2			-P2
connecté	U				P1+P2	-P1-P2				-P1-P2	
libre	U	-ID	-P2			P1	D	P1+P2	P1		
engagé	U	D	P2			-P1	-D	-P1-P2	-P1		
demande 1	V		-ID	-ID					-ID		
demande 2	V		D		-ID						-ID
connexion	V				D	-ID					

fig. 19 : transformation de la matrice d'incidence liée au système du téléphone (II)

La matrice simplifiée associée aux huit invariants obtenus sont repris à la fig. 20.

m4		c1	c4	c14	c15	m0	i1	i2	i3	i4	i5	i6	i7	i8
		U	V	V	V		U	U	U	U	U	U	U	E
inactif	U	-ID				ΣU	D		D					
cont, court, déconn	U	D	P1	P1+P2	P1+P2		D							
sans signal	U		-P1				D			-ID				
long	U				-P1		D						-ID	-ABS
sonnerie	U				-P2		D						-ID	ABS
connecté	U				-P1-P2		D				-ID			
libre	U	-ID				ΣU	D	D	-ID					
engagé	U	D					D							
demande 1	V		-ID							P1				
demande 2	V				-ID							P1	P2	
connexion	V				-ID							P1+P2		

fig. : 20 matrice simplifiée et invariants

Leur interprétation est simple. Le premier, notamment

$$M(\text{inactif}) + M(\text{continu}) + M(\text{court}) + M(\text{déconnecté}) + M(\text{sans signal}) + M(\text{long}) + M(\text{sonnerie}) + M(\text{connecté}) = \sum u.$$

rappelle qu'un téléphone doit se trouver obligatoirement dans un des huit états cités.

Nous pouvons ensuite déduire de (i2)

$$M(\text{libre}) + M(\text{engagé}) = \sum u$$

qu'un téléphone est soit "libre", soit "engagé".

La relation tirée de (i3)

$$M(\text{inactif}) = M(\text{libre})$$

signifie qu'un téléphone est "libre" si et seulement si il est "inactif".

Il découle de (i4)

$$M(\text{sans signal}) = P1(M(\text{demande 1}))$$

qu'un téléphone sur lequel on compose un numéro (i.e. dans l'état "sans signal") correspond à un téléphone en attente dans l'état "demande 1".

L'invariant (i5)

$$M(\text{connecté}) = (P1 + P2)(M(\text{connexion}))$$

exprime qu'un téléphone est "connecté" si et seulement si son numéro est l'un des deux appartenant à une paire v de la place "connexion".

Il résulte de (i6)

$$M(\text{long}) = P1(M(\text{demande 2}))$$

qu'un abonné percevant un signal "long" dans son combiné est le propriétaire d'un téléphone en attente dans la place "demande 2".

L'égalité issue de (i7)

$$M(\text{sonnerie}) = P2(M(\text{demande 2}))$$

affirme que les téléphones sur lesquels retentit une sonnerie sont ceux pour lesquels un appel est en attente dans la place "demande 2".

On conclut finalement à partir de (i8)

$$| M(\text{long}) | = | M(\text{sonnerie}) |$$

que le nombre de téléphones sur lesquels retentit une sonnerie est identique au nombre de téléphones dans lesquels on peut entendre un signal "long".

Nous avons illustré l'application des règles exposées mais il reste une question importante à résoudre : quelle règle choisir ? La décision du moment judicieux pour appliquer T2 peut notamment être difficile. Il faut pour cela découvrir des colonnes qui forment des combinaisons linéaires d'autres colonnes.

La technique consiste à chercher des colonnes qui possèdent le plus d'éléments non nuls sur les mêmes lignes. Nous pouvons observer, par exemple, que les quatre lignes dénommées "inactif", "cont., court, déconn.", "libre", et "engagé" de c_1 et c_6 sont toutes deux composées d'éléments non nuls.

Il nous reste à déterminer la colonne qui sera supprimée en faveur de c_{14} . La raison pour laquelle notre choix s'est porté sur c_6 est simple : le retrait de c_6 était immédiat alors que celui de c_1 nécessitait au préalable l'application de T1 de façon à remplacer c_1 par $c_1 \circ P1$.

En général, la règle consiste à éliminer la colonne la plus compliquée i.e. celle qui comporte le plus d'éléments non nuls ou les fonctions les plus complexes.

Une autre simplification consiste à utiliser les combinaisons linéaires. Celles-ci n'apparaissent pas toujours directement et plusieurs étapes sont parfois nécessaires. Il est préférable de commencer en combinant deux colonnes dans le but d'obtenir le maximum d'éléments communs avec une existante. Il faut ensuite chercher une troisième colonne comportant des éléments non nuls dans les mêmes lignes. En cas d'échec, il est toujours possible de continuer le processus en associant la colonne obtenue avec une nouvelle.

Nous pouvons combiner, par exemple, les colonnes c_1 et c_6 suivant la relation " $c_6 - c_1 \circ P2$ " (fig. 21). Le résultat obtenu comporte quatre occurrences de "P1+P2" qui figurent dans d'autres colonnes. La meilleure correspondance existe avec c_9 où seules les lignes intitulées "long", "sonnerie", "connecté" et "demande 2" diffèrent. De plus, en examinant attentivement le tableau, nous pouvons nous

rendre compte que c_5 possède exactement les quatre éléments non nuls qu'il manquait à la combinaison de c_1 et c_6 .

Finalement, la combinaison linéaire qui permet d'éliminer c_9 peut s'écrire :

$$c_5 + c_6 - c_1 \circ P_2$$

c_1	c_6	$c_6 - c_1 \circ P_2$	c_5	$c_5 + c_6 - c_1 \circ P_2$
-ID	P1	P1+P2		P1+P2
ID	P2			
			-P1	-P1
			-P2	-P2
	-P1-P2	-P1-P2	P1+P2	
-ID	P1	P1+P2		P1+P2
ID	-P1	-P1-P2		-P1-P2
			-ID	-ID
	-ID	-ID	ID	

fig. 21 : élimination de la colonne c_9

V. Méthode alternative basée sur les fonctions inverses généralisées.

Un autre point de vue est celui de M. Silva, J. Martinez, P. Ladet et H. Alla [4]. Ils ont mis au point une technique alternative pour calculer les invariants d'un RPC et insistent sur le fait que leur méthode n'est pas mutuellement exclusive avec les règles de transformation de K. Jensen mais simplement complémentaire.

La compréhension de leur raisonnement nécessite quelques préliminaires.

Définitions.

- Soit $h \in \mathbb{Q}^{n \times m}$ une fonction linéaire

$\text{Im}(h)$ est l'image de h

i.e. $\text{Im}(h) = \{y \in \mathbb{Q}^m \mid \text{il existe } x \in \mathbb{Q}^n \text{ tel que } x^T h = y^T\}$

$\text{Ker}(h)$ est le noyau de h

i.e. $\text{Ker}(h) = \{x \in \mathbb{Q}^n \mid x^T h = 0\}$

$\dim(h)$ est la dimension de $\text{Im}(h)$

$\dim(h) = \text{rang}(h) = m'$

- Soient CK et CR les sous-espaces complémentaires de $\text{Ker}(h)$ et $\text{Im}(h)$

i.e. $\mathbb{Q}^n = CK + \text{Ker}(h)$ et $\mathbb{Q}^m = CR + \text{Im}(h)$

h^* est appelée l'inverse généralisée de h (the (CK-CR) generalized inverse of h [7]) si

- 1) $\text{Im}(h^*) = CK$

- 2) $\text{Ker}(h^*) = CR$

- 3) $h \cdot h^* \cdot h = h$

- 4) $h^* \cdot h \cdot h^* = h^*$

(définition algébrique)

Définition équivalente :

Notons $h' = h_{CK}$ (i.e. h limitée à CK). Nous sommes ainsi assurés que

$$h'^{-1} : \text{Im}(h) \rightarrow CK$$

existe).

Pour $x \in \mathbb{Q}^m$, considérons que

$$x = r_1 + r_2 \quad \text{avec} \quad r_1 \in \text{Im}(h) \quad \text{et} \quad r_2 \in CR$$

La fonction h^* définie par :

$$h^* x = h'^{-1} r_1$$

est appelée fonction inverse généralisée de h . (définition fonctionnelle)

Propriété : les lignes de la matrice $(\text{id}_n - h \cdot h^*)$ génèrent $\text{Ker}(h)$

$$\text{Im}(\text{id}_n - h \cdot h^*) = \text{Ker}(h)$$

où $\text{id}_n \in Q^{n \times n}$ est la fonction identité.

La proposition suivante nous renseigne une manière pratique de calculer l'inverse généralisée d'une fonction linéaire.

Soient $t \in Q^{m' \times n}$ et $r \in Q^{m \times m'}$ des transformations linéaires tq

$$\text{rang}(t \cdot h) = \text{rang}(h \cdot r) = \text{rang}(h) = m'$$

Proposition 2 : la fonction $h^* = r \cdot (t \cdot h \cdot r)^{-1} \cdot t$ est une inverse généralisée de h .

dém. : 1) et 2) proviennent directement de l'équivalence des définitions algébrique et fonctionnelle

$$\begin{aligned} 3) \text{rang}(h) = \text{rang}(t \cdot h) &\Rightarrow h = v \cdot t \cdot h \quad (v \in Q^{n \times m}) \\ h = v \cdot t \cdot h &= v \cdot \text{id}_m \cdot t \cdot h = v \cdot (t \cdot h \cdot r) \cdot (t \cdot h \cdot r)^{-1} \cdot t \cdot h \\ &= v \cdot t \cdot h \cdot r \cdot (t \cdot h \cdot r)^{-1} \cdot t \cdot h = h \cdot r \cdot (t \cdot h \cdot r)^{-1} \cdot t \cdot h \\ &= h \cdot h^* \cdot h \end{aligned}$$

$$\begin{aligned} 4) h^* \cdot h \cdot h^* &= r \cdot (t \cdot h \cdot r)^{-1} \cdot t \cdot h \cdot r \cdot (t \cdot h \cdot r)^{-1} \cdot t \\ &= r \cdot (t \cdot h \cdot r)^{-1} \cdot t \\ &= h^* \end{aligned}$$

1. Invariants d'un RPC avec fonction-identité.

Définition.

Soit $\text{RPC}_a = (P, T, C, A, M_0)$ où

1) $P = \{p_1, \dots, p_n\}$

2) $T = \{t_1, \dots, t_m\}$

3) $C(p_i) = C(t_j) = D$, $\forall p_i \in P$ et $\forall t_j \in T$ et où D est un ensemble fini non vide
de couleurs

4) $A(p_i, t_j) = a_{ij} \cdot \text{id}$ où $a_{ij} \in Z$ et $\text{id} : [[D \rightarrow [D \rightarrow Z]]$ et $\text{id}(c) = c$, $\forall c \in D$

5) M_0 est le marquage initial de RPC_a .

Un tel réseau porte le nom de RPC avec fonction-identité. Il possède le même ensemble de couleurs pour les places et les transitions et c'est la fonction-identité qui est associée à tous ses arcs.

Nous pouvons générer plusieurs réseaux de Pétri à partir d'un RPC avec fonction-identité en créant une place pour chaque $C(p_i)$ et une transition pour tout $C(t_j)$. Nous pouvons obtenir ainsi autant de réseaux de Pétri que D n'a de couleurs. Tous ces réseaux de Pétri sont associés à la même matrice d'incidence

$$A' = [a_{ij}]_{1 \leq i \leq n, 1 \leq j \leq m'}$$

Définition.

Soit RPC_a , un réseau de Pétri coloré avec fonction-identité, le réseau de Pétri associé à RPC_a est

$$RP_{a'} = (P, T, A', M_0') \quad \text{où}$$

- 1) $P = \{p_1, \dots, p_n\}$
- 2) $T = \{t_1, \dots, t_m\}$
- 3) $A'(p_i, t_j) = a_{ij}$
- 4) M_0' est le marquage initial de $RP_{a'}$

Définition.

Soit $a' = (a_1, \dots, a_n)$ un invariant de A' et soit l'application

$$h : \text{Ker}(A') \rightarrow \text{Ker}(A) \quad \text{tq} \quad h(a') = \text{id} \cdot a'$$

on a :

$$h[(a_1, \dots, a_n)] = (a_1 \cdot \text{id}, \dots, a_n \cdot \text{id}) \\ \forall (a_1, \dots, a_n) \in \text{Ker}(A') \quad \text{où} \quad a_1, \dots, a_n \in \mathbb{Q}$$

$\text{Ker}(A')$ est un sous-espace de \mathbb{Q}^n et $\text{Ker}(A)$ est un sous-module de E , où E est le module des fonctions linéaires $\mathbb{Q}^{d \times d}$ et $d = \text{card}(D)$.

Proposition 3 : les images, sous h , de toute paire d'éléments linéairement indépendants $a, a' \in \text{Ker}(A')$ sont linéairement indépendantes.

$$\text{dém. : } f \cdot h(a) + f' \cdot h(a') = 0 \\ \Leftrightarrow (a_1 \cdot f, \dots, a_n \cdot f) + (a_1' \cdot f', \dots, a_n' \cdot f') = 0$$

que l'on peut réécrire :

$$a_1 \cdot f + a_1' \cdot f' = 0$$

$$a_n \cdot f + a_n' \cdot f' = 0$$

Or, a et a' sont, par hypothèse, linéairement indépendants

$$\Rightarrow f = f' = 0$$

$\Rightarrow h(a)$ et $h(a')$ sont linéairement indépendants.

Proposition 4 : les images des éléments d'une base de $\text{Ker}(A')$ par h constituent une base de $\text{Ker}(A)$.

dém. : soit $B_{a'} = \{v_1, \dots, v_q\}$ une base de $\text{Ker}(A')$.

D'après la proposition 2, nous déduisons que

$\{h(v_1), \dots, h(v_q)\}$ sont linéairement indépendants

$$\text{et } \dim [\text{Im} [h(v_1), \dots, h(v_q)]^T]$$

$$= \sum \dim [\text{Im} [h(v_i)]]$$

$$= \sum \dim [\text{Im} [v_i \cdot \text{id}]] = q \cdot d$$

il ne reste plus qu'à montrer que

$$\dim [\text{Ker}(A)] = q \cdot d$$

$$\dim [\text{Ker}(A)] = d \cdot n - \text{rang} (A)$$

$$= d \cdot n - d \cdot \text{rang} (A')$$

$$= d \cdot [n - \text{rang} (A')]$$

$$= d \cdot \dim [\text{Ker}(A')] = d \cdot q$$

Nous pouvons alors conclure que le calcul des invariants d'un RPC avec fonction-identité peut se réduire au calcul des invariants d'un réseau de Pétri traité précédemment.

2. Calcul des invariants symboliques d'un RPC.

2.1 Cas où W est factorisable.

Le calcul des invariants d'un RPC est basé sur l'équation matricielle :

$$x \cdot W = 0$$

$$\text{où } x : P \rightarrow [C(p) \rightarrow [U \rightarrow Z]]$$

U est un ensemble non vide de couleurs.

Si l'ensemble U se résume à un singleton, nous sommes ramenés au cas classique du calcul des invariants d'un réseau de Pétri. Outre les techniques citées, nous verrons que la programmation linéaire peut s'appliquer pour résoudre ce problème.

Le calcul est plus complexe lorsqu'il se rapporte à un réseau de Pétri coloré. De plus, l'analyse d'un RPC s'avère plus fructueuse lorsqu'elle est construite à partir d'un invariant basé sur un ensemble de couleurs.

Une manière simple de découvrir de tels invariants consisterait à chercher les invariants associés avec chaque couleur de U ainsi qu'à leur combinaison. De tels résultats pourraient être définis numériquement mais leur interprétation pourrait s'avérer pénible.

Il serait plus utile de déduire des invariants définis par un ensemble de fonctions

$$x_i \in [C(p_i) \rightarrow [U \rightarrow Z]]$$

tq les expressions $x_i \cdot m_i$ ont une signification claire pour le concepteur.

Cela peut être réalisé à condition que toutes les fonctions non nulles x_i puissent s'écrire comme composition de fonctions linéaires connues (somme et produit matriciels) : fonction identité, fonction de W et fonctions inverses généralisées de celles-ci.

Nous appellerons de tels invariants, des invariants symboliques. Ce nom s'applique notamment aux invariants d'un RPC avec fonction-identité obtenus à partir des images par h des invariants de la matrice A' du réseau de Pétri associé.

La matrice considérée ici, est factorisable, ce qui signifie qu'elle peut s'exprimer sous la forme :

$$W = F \cdot A$$

où A est la matrice d'un RPC avec fonction-identité.

2.1.1 Factorisation de W.

Définition.

Soit $RPC_W = (P, T, C, W, M_0)$ où

$$1) P = \{p_1, \dots, p_n\}$$

$$2) T = \{t_1, \dots, t_m\}$$

$$3) C(p_i) = D_i, \quad \forall p_i \in P \quad \text{et} \\ C(t_j) = D$$

$$4) W(p_i, t_j) = a_{ij} \cdot f_j, \quad \forall (p_i, t_j) \in P \times T \\ a_{ij} \in Z, \quad f_j \in [C(t_j) \rightarrow [C(p_i) \rightarrow Z]]$$

Ces quatre conditions définissent un réseau dans lequel la fonction couleur appliquée à une transition quelconque donne comme résultat le même ensemble de couleurs D. De plus, tous les arcs qui arrivent ou qui quittent une place p_i possèdent la même fonction f_j pondérée par des coefficients entiers a_{ij} différents.

Un tel réseau est caractérisé par une matrice d'incidence factorisable :

$$W = F \cdot A$$

où • F est une matrice diagonale
diag (F) = (f_1, \dots, f_n)

• $A = [a_{ij} \cdot \text{id}]_{1 \leq i \leq n, 1 \leq j \leq m}$ est la matrice d'incidence d'un RPC avec fonction-identité

Le calcul des invariants peut alors s'écrire :

$$x \cdot W = x \cdot F \cdot A = z \cdot A = 0 \quad \text{où} \quad z = x \cdot F$$

Il suffit de procéder en deux étapes :

1) calculer les invariants symboliques y de A qui seront de la forme :

$$y = (y_1 \cdot \text{id}, \dots, y_n \cdot \text{id})$$

2) déduire les invariants symboliques x qui satisfont

$$x \cdot F = z$$

$$\text{où } z = f \cdot y = (y_1 \cdot f, \dots, y_n \cdot f)$$

$$\text{avec } f \in [D \rightarrow [U \rightarrow Z]]$$

Le premier problème a déjà été résolu au paragraphe précédent et le deuxième est traité ci-dessous.

2.1.2 Calcul des invariants symboliques d'un RPC_w à partir des invariants symboliques d'un RPC_a

Nous allons montrer qu'il est possible de déduire des invariants symboliques d'un réseau de Pétri coloré à partir d'invariants du RPC avec fonction-identité associé, plus faciles à calculer.

Nous donnerons une méthode de calcul pour ces invariants.

Soit $y = (y^1 \cdot \text{id}, \dots, y^n \cdot \text{id})$ un invariant symbolique de A , où $\forall i \in [1, \dots, n], y^i \in Z$.

Définitions.

- Notons FS_y l'ensemble des fonctions $f_i \in \text{diag}(F)$ associé au support de y i.e.

$$FS_y = \{ f_i \in \text{diag}(F) \mid y_i \neq 0 \}$$

- Posons $\text{Im}(y)$ l'image de l'invariant symbolique y , i.e.

$$\text{Im}(y) = \{ (y^1 \cdot g, \dots, y^n \cdot g) \mid g \in [D \rightarrow [U \rightarrow Z]] \}$$

où U est un ensemble non vide de couleurs.

Proposition 5 : si $x \cdot F \in \text{Im}(y)$ alors

$$x \cdot W = 0$$

et il existe $g \in [D \rightarrow [U \rightarrow Z]]$ tq $x_i \cdot f_i = y^i \cdot g \quad \forall f_i \in \text{diag}(F)$

dém. : si $x \cdot F \in \text{Im}(y) \Leftrightarrow$ il existe g tq $x \cdot F = g \cdot y$
 $\Leftrightarrow (x_1 \cdot f_1, \dots, x_n \cdot f_n) = (y^1 \cdot g, \dots, y^n \cdot g)$
 D'autre part, $x \cdot W = x \cdot F \cdot A = (x_1 \cdot f_1, \dots, x_n \cdot f_n) \cdot A$
 $= (y^1 \cdot g, \dots, y^n \cdot g) \cdot A$
 $= g \cdot (y^1 \cdot \text{id}, \dots, y^n \cdot \text{id}) \cdot A$
 $= g \cdot y \cdot A = 0$

La fonction g joue un rôle capital dans le calcul des invariants de la matrice W . Nous pouvons caractériser cette fonction à l'aide de la proposition 5.

$$x_i \cdot f_i = y^i \cdot g, \quad \forall f_i \in \text{diag}(F) \Rightarrow \text{Im}(g) \subseteq \text{Im}(f_i) \quad \forall f_i \in \text{FS}_y.$$

Parmi toutes les fonctions g qui vérifient la dernière relation, celles dont $\text{Im}(g)$ est maximum sont d'un intérêt tout spécial pour nous.

Définissons f comme une fonction avec une image maximale définie par l'intersection des images des fonctions f_i de FS_y :

$$\text{Im}(f) = \text{Im}(g)_{\max} = \bigcap_{f_i \in \text{FS}_y} \text{Im}(f_i)$$

Soit $\{h_i\}$ un ensemble de fonctions, où $\forall i$ tq $y^i \neq 0$, h_i est définie par :

$$h_i \in [D_i \rightarrow [U \rightarrow Z]]$$

et vérifie :

- 1) $h_i \cdot f_i = f$
- 2) $\text{Im}(f) = \bigcap_{f_i \in \text{FS}_y} \text{Im}(f_i)$

Proposition 6 : $x = (y^1 \cdot h_1, \dots, y^n \cdot h_n)$ est un invariant symbolique de W .

dém. : $x \cdot W = x \cdot F \cdot A = (y^1 \cdot h_1 \cdot f_1, \dots, y^n \cdot h_n \cdot f_n) \cdot A$
 $= f \cdot (y^1 \cdot \text{id}, \dots, y^n \cdot \text{id}) \cdot A \quad (h_i \cdot f_i = f)$
 $= f \cdot y \cdot A = 0 \quad (y \text{ est un invariant de } A)$

La proposition 7 nous permet d'exprimer les fonctions h_i à partir de fonctions connues. Plus particulièrement, si une condition spéciale est satisfaite, le

corollaire 2 nous permet de les écrire à partir des fonctions de W et de leurs inverses généralisées.

Proposition 7 : considérons $h_i = f \cdot f_i^*$. Nous pouvons affirmer que :

$$x = (y^1 \cdot f \cdot f_1^*, \dots, y^n \cdot f \cdot f_n^*) \text{ est un invariant de } W$$

où f_i^* désigne la fonction inverse généralisée de f_i , $\forall i \in [1, \dots, n]$

dém. : $x \cdot W = x \cdot F \cdot A = (y^1 \cdot f \cdot f_1^* \cdot f_1, \dots, y^n \cdot f \cdot f_n^* \cdot f_n) \cdot A$

En considérant que :

$$1) f = h_i \cdot f_i \text{ alors } f \cdot f_i^* \cdot f_i = h_i \cdot f_i \cdot f_i^* \cdot f_i = h_i \cdot f_i = f \\ \forall f_i \in FS_y, \text{ et}$$

$$2) y^i = 0, \forall f \in FS_y.$$

Nous pouvons écrire :

$$x \cdot W = (y^1 \cdot f, \dots, y^n \cdot f) \cdot A = f \cdot (y^1 \cdot \text{id}, \dots, y^n \cdot \text{id}) \cdot A \\ = f \cdot y \cdot A = 0$$

Corollaire 1 : les invariants suivants sont des invariants symboliques de W .

$$a) x_1 = (y^1 \cdot h_m \cdot f_m \cdot f_1^*, \dots, y^m \cdot h_m \cdot f_m \cdot f_m^*, \dots, y^n \cdot h_m \cdot f_m \cdot f_n^*)$$

$$b) x_2 = (0, \dots, 0, \text{id}_m - f_m \cdot f_m^*, 0, \dots, 0)$$

$$c) x_3 = (y^1 \cdot h_m \cdot f_m \cdot f_1^*, \dots, y^m \cdot h_m, \dots, y^n \cdot h_m \cdot f_m \cdot f_n^*)$$

dém. : $x = (y^1 \cdot f \cdot f_1^*, \dots, y^n \cdot f \cdot f_n^*)$ est un invariant de W .

$$\Rightarrow a) \text{ pour } f = h_m \cdot f_m \text{ alors } x_1 = x \Rightarrow x_1 \cdot W = 0$$

$$b) (\text{id}_m - f_m \cdot f_m^*) \cdot f_m = 0 \Rightarrow x_2 \cdot F = 0 \\ \Rightarrow x_2 \cdot W = 0$$

$$c) x_3 = x_1 + y^m \cdot h_m \cdot x_2 \Rightarrow x_3 \cdot W = 0$$

Corollaire 2 : s'il existe $f_m \in FS_y$ tq $\text{Im}(f_m) \subset \text{Im}(f_i)$,
 $\forall f_i \in FS_y$ alors

$$x = (y^1 \cdot f_m \cdot f_1, \dots, y^m \cdot \text{id}_m, \dots, y^n \cdot f_m \cdot f_n^*)$$

est un invariant de W .

dém. : d'après le corollaire 1,

$$x_3 = (y^1 \cdot h_m \cdot f_m \cdot f_1^*, \dots, y^m \cdot h_m, \dots, y^n \cdot h_m \cdot f_m \cdot f_n^*)$$

est un invariant symbolique de W .

En prenant $f = f_m$ et $h_m = \text{id}_m$, on peut conclure que

$$x = (y^1 \cdot f_m \cdot f_1, \dots, y^m \cdot \text{id}_m, \dots, y^n \cdot f_m \cdot f_n^*)$$

est aussi un invariant de W .

Ces résultats fournissent une méthode de calcul pour les invariants symboliques de W . Illustrons celle-ci à l'aide de la fig. 22 qui reprend trois cas particuliers que l'on peut rencontrer lors de la recherche de l'intersection des images des fonctions de FS_y . Supposons

$$y = (y^1 \cdot \text{id}, y^2 \cdot \text{id}, y^3 \cdot \text{id}, 0, \dots, 0)$$

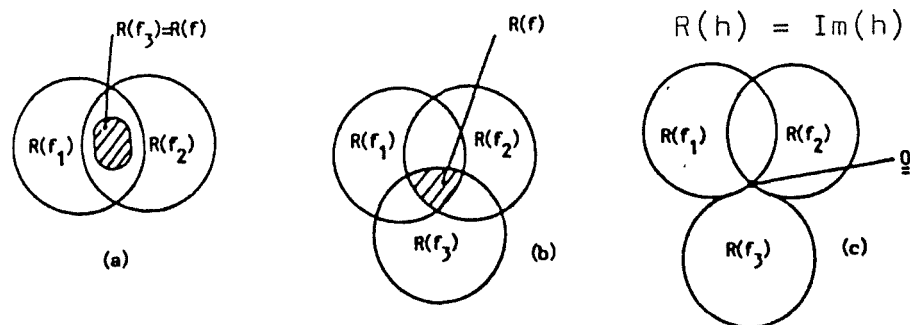


fig. 22 : intersection des images des fonctions de FS_y : trois cas possibles

cas a : $\text{Im}(f) = \bigcap_{f_i \in FS_y} \text{Im}(f_i) = \text{Im}(f_3)$, i.e. l'image de f_3 est incluse dans les images des autres fonctions de FS_y . En appliquant le corollaire 2 et en considérant que :

$$f = h_3 \cdot f_3 = f_3 \quad \text{i.e.} \quad h_3 = \text{id},$$

nous pouvons écrire l'invariant :

$$x = (y^1 \cdot f_3 \cdot f_1^*, y^2 \cdot f_3 \cdot f_2^*, y^3 \cdot \text{id}, 0, \dots, 0).$$

cas b : $\text{Im}(f) = \bigcap_{f_i \in \text{FS}_y} \text{Im}(f_i) \neq \text{Im}(f_k) \quad \forall f_k \in \text{FS}_y$, i.e. $\text{Im}(f)$ ne correspond pas à l'image d'une des fonctions de FS_y . L'invariant calculé sera :

$$x = (y^1 \cdot h_3 \cdot f_3 \cdot f_1^*, y^2 \cdot h_3 \cdot f_3 \cdot f_2^*, y^3 \cdot h_3, 0, \dots, 0)$$

où h_3 doit vérifier $f = h_3 \cdot f_3$.

cas c : $\text{Im}(f) = 0$, i.e. l'intersection des images n'est autre que l'ensemble vide. Dans ce cas,

$$h_3 = 0$$

et donc, seul l'invariant suivant peut être établi :

$$x = (0, \dots, 0).$$

Finalement, supposons que les r fonctions de FS_y puissent être ordonnées de la façon suivante :

$$\text{FS}_y = \{g_1, g_2, \dots, g_r\}$$

où la fonction g_j vérifie la propriété de commutativité énoncée ci-dessous :

$$\prod_{(j=i, \dots, r)} g_j = \left(\prod_{(j=i+1, \dots, r)} g_j \right) \cdot g_i \quad \forall i \in [1, \dots, r]$$

ce qui signifie que g_i commute avec le produit $g_{i+1} \cdot \dots \cdot g_r$.

Corollaire 3 : dans de telles conditions,

$$x = (y^1 \cdot h_1, \dots, y^n \cdot h_n)$$

est un invariant symbolique de W ,
où $h_i = \prod g_j$ ($j \in [1, \dots, r]$ et $g_j \neq f_j$).

2.1.3 Calcul d'un ensemble d'invariants symboliques de RPC_W .

Comme nous l'avons illustré au début de ce chapitre, l'analyse d'un réseau ne se base pas que sur un seul invariant mais sur tout un groupe. Un ensemble adéquat d'invariants est même capital pour accomplir un bon travail de validation.

Une base pourrait, par exemple, s'avérer très utile pour l'analyste qui serait ainsi capable de générer tous les invariants nécessaires à des résultats fructueux.

Ce paragraphe est consacré à la recherche de tels ensembles.

Définitions. [8]

- Le support d'un vecteur f de N^n est l'ensemble des éléments i de $\{1, \dots, n\}$ tel que la $i^{\text{ième}}$ composante de f soit non nulle.
- Soient G un sous-ensemble de N^n et f un vecteur non nul de G . On dit que f est à support minimal dans G , si et seulement si il n'existe pas dans G un vecteur g non nul tel que son support soit strictement contenu dans celui de f .

Considérons alors $y = (y^1 \cdot \text{id}, \dots, y^n \cdot \text{id})$ un invariant de A avec son support non minimal. Il est toujours possible de trouver une base d'invariants de A où chaque y_i est un invariant à support minimal. Le vecteur y peut s'exprimer comme suit :

$$k \cdot y = \sum k_i \cdot y_i$$

$$\text{où } \begin{cases} \forall i \in [1, \dots, q] : k_i \in \mathbb{Z} \\ k \in \mathbb{Z} \end{cases}, \quad y_i = (y_i^1 \cdot \text{id}, \dots, y_i^n \cdot \text{id})$$

Soient $x = (y^1 \cdot f \cdot f_1^*, \dots, y^n \cdot f \cdot f_n^*)$ et $x_i = (y_i^1 \cdot g_i \cdot f_1^*, \dots, y_i^n \cdot g_i \cdot f_n^*)$ les invariants de W calculés en appliquant la proposition 7 à y et à y_i , respectivement. Il est facile de vérifier que :

$$\text{Im}(f) \subseteq \text{Im}(g_i) \quad \forall i \in [1, \dots, q]$$

$$\Rightarrow \text{il existe } h_i \text{ tq } f = h_i \cdot g_i$$

Proposition 8 : tout invariant symbolique, x , d'une matrice factorisable peut s'exprimer comme une combinaison linéaire des invariants à support minimal x_i :

$$k \cdot x = \sum k_i \cdot h_i \cdot x_i$$

$$\begin{aligned} \text{dém. : } & \sum k_i \cdot h_i \cdot x_i \\ &= (\sum k_i \cdot h_i \cdot y_i^1 \cdot g_i \cdot f_1^*, \dots, \sum k_i \cdot h_i \cdot y_i^n \cdot g_i \cdot f_n^*) \\ &= (\sum k_i \cdot y_i^1 \cdot f \cdot f_1^*, \dots, \sum k_i \cdot y_i^n \cdot f \cdot f_n^*) \\ &= k \cdot (y^1 \cdot f \cdot f_1^*, \dots, y^n \cdot f \cdot f_n^*) \\ &= k \cdot x \end{aligned}$$

Nous pouvons ainsi conclure que le processus de calcul des invariants symboliques de RPC_W peut se restreindre aux invariants à support minimal.

Nous allons étudier à nouveau le système de bases de données. Sa matrice d'incidence n'est pas factorisable et donc, en principe, la méthode de calcul proposée ne peut pas lui être appliquée.

Considérons pour le moment le sous-réseau qui contient toutes ses places mais seulement les transitions t_1 et t_2 . Appelons sa matrice d'incidence W_1 . Celle-ci est factorisable, $W_1 = F_1 \cdot A_1$

Comme

$$\text{Im (ID) = Im (MES)}$$

$$\text{IM (ABS) } \subset \text{ Im (ID)}$$

nous pouvons calculer les douze invariants symboliques de la table 2 en appliquant le corollaire 2.

	I	A	Ac	E	In	En	R	C	
v1	ID								
v2								ID	
v3	ID	ID							
v4	MES					ID		ID	
v5	ABS		ID						
v6				ID	ABS.MES*		ABS.MES*		
v7	MES			ID					
v8					ID	ID	ID		
v9	- MES					ID	ID		
v10	- ABS	ID							
v11	- MES	ID							
v12	ID - ABS.MES*								

table 2 : invariants symboliques à support minimal de W_1

A cet ensemble d'invariants symboliques de W_1 , nous devons ajouter ceux de F_1 . Ces derniers sont présentés à la table 3 dans laquelle k_f désigne une fonction linéaire de $\text{Ker}(f)$ i.e. $k_f \cdot f = 0$ ($k_f = \text{id} - f \cdot f^*$ est une solution à l'équation en question).

	I	A	Ac	E	In	En	R	C
k1					k _{mes}			
k2						k _{mes}		
k3								k _{mes}

table 3 : invariants de W_1 qui sont aussi ceux de F_1

Détaillons la manière d'obtenir deux invariants de la table 2.

- v_6 : le point de départ est l'invariant a_6 de A_1 . Le support de a_6 et l'ensemble des fonctions de F qui sont associées à ce support sont :

$$\text{support}(a_6) = \{E, En, C\}$$

$$FS_{a_6} = \{f_E, f_{En}, f_C\}$$

$$\text{où } f_E = \text{ABS} \quad , \quad f_C = f_{En} = \text{MES}$$

Nous pouvons appliquer le corollaire 2 en choisissant $f_m = \text{ABS}$ puisque

$$\text{Im}(\text{ABS}) \subset \text{Im}(\text{MES})$$

Nous obtenons ainsi :

$$v_6 = (0 \ 0 \ 0 \ x_E \ 0 \ x_{En} \ 0 \ x_C)$$

$$\text{où } x_E = \text{ID}$$

$$x_{En} = f_E \cdot f_{En}^* = \text{ABS} \cdot \text{MES}^*$$

$$x_C = f_E \cdot f_C^* = \text{ABS} \cdot \text{MES}^*$$

- v_4 : considérons maintenant l'invariant a_4 de A_1 :

$$\text{support}(a_4) = \{I, En, C\}$$

$$\text{où } f_I = \text{ID}$$

$$f_{En} = f_C = \text{MES}$$

Ces trois fonctions peuvent jouer le rôle de f_m dans le corollaire 2 puisque

$$\text{Im}(\text{ID}) = \text{Im}(\text{MES})$$

Si nous choisissons $f_m = f_I$, nous obtenons :

$$v4 = (x_I \ 0 \ 0 \ 0 \ 0 \ x_{E_n} \ 0 \ x_C)$$

$$\text{où } x_I = \text{ID}$$

$$x_{E_n} = f_I \cdot f_{E_n}^* = \text{ID} \cdot \text{MES}^* = \text{MES}^*$$

$$x_C = f_I \cdot f_C^* = \text{ID} \cdot \text{MES}^* = \text{MES}^*$$

D'autre part, si notre choix se porte sur f_{E_n} , l'invariant résultant s'écrit :

$$v4' = (x_I \ 0 \ 0 \ 0 \ 0 \ x_{E_n} \ 0 \ x_C)$$

$$\text{où } x_I = f_{E_n} \cdot f_I^* = \text{MES} \cdot \text{ID}^* = \text{MES}$$

$$x_{E_n} = \text{ID}$$

$$x_C = f_{E_n} \cdot f_C^* = \text{MES} \cdot \text{MES}^*$$

Les différentes formes de l'invariant symbolique $v4$ proviennent du fait que l'invariant $k2$ a été intégré dans $v4'$. En effet, réécrivons $k2$:

$$k2 = (0 \ 0 \ 0 \ 0 \ 0 \ 0 \ \text{ID} - \text{MES} \cdot \text{MES}^* \ 0 \ 0)$$

Il est facile de montrer que :

$$v4' = \text{MES} \cdot v4 + k2$$

Etant donné que $f_C = f_{E_n}$, le corollaire 1 nous permet d'écrire :

$$v4' = (x_I \ 0 \ 0 \ 0 \ 0 \ x_{E_n} \ 0 \ x_C)$$

$$\text{où } x_I = \text{MES}$$

$$x_{E_n} = \text{ID}$$

$$x_C = \text{ID}.$$

qui est justement l'invariant qui apparaît dans la table 2. Il faut finalement remarquer que k_2 peut être exprimé sous forme d'une combinaison linéaire de v_4 et k_3 :

$$k_2 = k_{mes} \cdot v_4 - k_3$$

Considérons maintenant les tables 2 et 3. Elles contiennent des informations redondantes puisqu'il s'y trouve des invariants qui sont des combinaisons linéaires d'autres.

Il est évidemment inutile pour l'analyse du réseau d'avoir une surpopulation d'invariants. Un ensemble plus restreint est même préférable. Celui-ci peut éventuellement constituer une base. Une autre possibilité intéressante serait de générer tous les invariants symboliques non négatifs. Cet intérêt est dû à la signification physique spéciale que comportent habituellement les invariants non négatifs d'un réseau.

Voyons comment de tels invariants peuvent être établis.

Dans la table 2, les invariants v_1, v_2, \dots, v_9 sont non négatifs et comme nous le verrons plus tard, constituent un ensemble qui génère tous les invariants de W_1 .

Pour calculer une base d'invariants symboliques de W_1 , nous ne pouvons pas choisir n'importe quelle base de A_1 . Par exemple, considérons B_{A_1} , une base d'invariants de A_1 :

$$B_{A_1} = \{a_1, a_2, a_5, a_6, a_{10}, a_{12}\}$$

A partir de ces éléments, nous pouvons déduire l'ensemble d'invariants de W_1

$$SW_1 = \{v_1, v_2, v_5, v_6, v_{10}, v_{12}\}$$

Soit maintenant, B_{F_1} un ensemble d'invariants de F_1 qui constitue une base :

$$B_{F_1} = \{k_1, k_2, k_3\},$$

il est aisé de montrer que dans ce cas, $SW_1 \cup B_{F_1}$ ne forme pas un ensemble qui génère tous les invariants de W_1 . Il suffit pour cela de vérifier que v_3 ou v_4 ne peuvent pas s'exprimer comme une combinaison linéaire des vecteurs de $SW_1 \cup B_{F_1}$. Par contre, si nous choisissons :

$$B_{A_1'} = \{a_1, a_2, a_3, a_4, a_5, a_7\}$$

$$\text{d'où } SW_1' = \{v_1, v_2, v_3, v_4, v_5, v_7\}$$

il est possible de prouver que :

$$GW_1' = SW_1' \cup B_{F_1}$$

constitue un ensemble qui génère tous les invariants de A et que

$$BW_1 = GW_1' - \{k_1, k_2\}$$

est une base.

Nous avons illustré ainsi la difficulté qui se présente lors de la construction d'une base d'invariants de W . Il existe cependant une possibilité de calculer directement une base. Pour cela, considérons $B_A = \{a_i\}_{1 \leq i \leq q}$ une base d'invariants symboliques de A qui vérifie les deux conditions suivantes :

1) B_A est de la forme suivante :

$$B_A^T = \begin{bmatrix} a_1 & \pi_1 & \pi_2 & \dots & \pi_q & \dots & \pi_n \\ a_2 & a_{11} \cdot \text{id} & \dots & & & & \\ \dots & & a_{22} \cdot \text{id} & \dots & & & \\ a_q & & & & a_{qq} \cdot \text{id} & & \end{bmatrix}$$

○

□ $a_j \cdot \text{id}$

où π représente une permutation des places colorées $p_j \in \mathbb{C}^P$.

2) $\forall a_i \in B_A$, $\forall f_j \in FS_{a_i}$, $\text{Im}(f_j) \subset \text{Im}(f_i)$. Nous pouvons donc appliquer le corollaire 2 à a_i .

Nous supposons, pour simplifier l'écriture, que π est la permutation identité.

Proposition 9 : l'image par F d'un invariant x de W peut s'exprimer par rapport à la base B_A de la façon suivante :

$$y = x \cdot F = (u_1 \cdot f_1, \dots, u_q \cdot f_q) \cdot B_A^T$$

$$\text{où } x_j \in [C(p_j) \rightarrow [U \rightarrow Z]] \text{ , } \forall j \in [1, \dots, n] \text{ et}$$

$$u_i \in [C(p_i) \rightarrow [U \rightarrow Q]] \text{ , } \forall i \in [1, \dots, q].$$

Nous allons présenter une démonstration constructive de cette proposition en introduisant une méthode de calcul des fonctions u_i ($i \in [1, \dots, q]$).

dém. : comme y est l'image de x par F , nous pouvons écrire :

$$y = x \cdot F = (x_1 \cdot f_1, \dots, x_n \cdot f_n) \quad (1)$$

or, x est un invariant de W , i.e.

$$x \cdot W = 0 \Leftrightarrow x \cdot F \cdot A = y \cdot A = 0$$

$$\Rightarrow y \in \text{Ker}(A).$$

y peut donc s'exprimer de la façon suivante :

$$y = (y^1, \dots, y^q) \cdot B_A^T \quad (2)$$

Les équations (1) et (2) nous amènent à écrire :

$$\begin{aligned} a_{11} \cdot y_1 &= x_1 \cdot f_1 \\ a_{12} \cdot y_1 + a_{22} \cdot y_2 &= x_2 \cdot f_2 \\ \dots & \\ a_{1j} \cdot y_1 + \dots &+ a_{jj} \cdot y_j = x_j \cdot f_j \\ \dots & \\ a_{1q} \cdot y_1 + \dots &+ a_{qq} \cdot y_q = x_q \cdot f_q \end{aligned}$$

La connaissance de y_1, \dots, y_{j-1} peut alors conduire à celle de y_j . En effet,

$$\left. \begin{array}{l} y_1 = u_1 \cdot f_1 \\ y_1 = x_1 \cdot f_1 / a_{11} \end{array} \right\} \Rightarrow u_1 = x_1 / a_{11}$$

$$\begin{aligned} y_j &= u_j \cdot f_j \\ y_j &= (x_j \cdot f_j - a_{1j} \cdot y_1 \dots - a_{j-1j} \cdot y_{j-1}) / a_{jj} \\ y_j &= (x_j - a_{1j} \cdot u_1 \cdot f_1 \dots - a_{j-1j} \cdot y_{j-1} \cdot f_{j-1}) / a_{jj} \\ y_j &= (x_j - a_{1j} \cdot u_1 \cdot h_{1j} \dots - a_{j-1j} \cdot u_{j-1} \cdot h_{j-1j}) \cdot f_j \\ &\quad \text{où } h_{ij} \text{ est une fonction telle que } f_i = h_{ij} \cdot f_j \end{aligned}$$

$$\begin{aligned} \Rightarrow u_j &= (x_j - \sum_{i \in [1, \dots, j-1]} a_{ij} \cdot u_i \cdot h_{ij}) \\ &\quad (i \in [1, \dots, j-1] \text{ et } a_{ij} \neq 0) \\ &\quad \text{où } h_{ij} \text{ vérifie } f_i = h_{ij} \cdot f_j \end{aligned}$$

Notons $G_W = \{g_i\}_{1 \leq i \leq q}$ l'ensemble des invariants de W calculés en appliquant le corollaire 2 aux éléments de la base B_A .

$$\begin{aligned} g_i \cdot F &= (0, \dots, 0, a_{ij} \cdot \text{id}, \dots, a_{ij} \cdot f_i \cdot f_j^*, \dots) \cdot F \\ &= (0, \dots, 0, a_{ij} \cdot f_i, \dots, a_{ij} \cdot f_i \cdot f_j^* \cdot f_j, \dots) \end{aligned}$$

Comme $\text{Im}(f_i) \subset \text{Im}(f_j) \Rightarrow f_i \cdot f_j^* \cdot f_j = f_i$, nous pouvons écrire :

$$g_i \cdot F = f_i \cdot (0, \dots, 0, a_{ij} \cdot \text{id}, \dots, a_{ij} \cdot \text{id}, \dots) = f_i \cdot a_i$$

et

$$G_W^T \cdot F = \begin{bmatrix} f_1 \\ \dots \\ f_q \end{bmatrix} \cdot B_A^T$$

Soient F_Q et F_C des matrices diagonales de fonctions linéaires définies de la façon suivante :

- $\forall i \in [1, \dots, k] : \text{RPC}_i = (P, T_i, C_i, W_i, M_0)$ tq
- 1) $\cup T_i = T$ et $T_i \cap T_j = \emptyset, \forall j \neq i$
 - 2) $C_i(p) = C(p) \forall p \in P$ et $C_i(t) = C(t) \forall t \in T_i$
 - 3) $W_i(p, t) = W(p, t), \forall (p, t) \in P \times T$
 - 4) $W_i = F_i \cdot A_i$

En général, les trois premières conditions sont vérifiées par un grand nombre de partitions. Seule, la dernière restreint les possibilités. Elle contient implicitement la relation d'incompatibilité entre les transitions d'un RPC énoncée ci-dessous.

Définition.

Deux transitions $t_j, t_k \in T$ sont incompatibles et ne peuvent pas appartenir en même temps à l'ensemble des transitions T_i de RPC_i si il existe $p \in P$ tq $W(p, t_j) \neq 0$ et $W(p, t_k) \neq 0$ et il n'existe pas $m, n \in \mathbb{Z}$ tq $m \cdot W(p, t_j) = n \cdot W(p, t_k)$.

Une solution simpliste à la condition de incompatibilité consiste à scinder W en matrices W_i composées d'une seule transition i.e.

$$W_i = [t_i]$$

La matrice d'incidence du RPC modélisant le système de bases de données peut être décomposée à l'aide de la table 4 qui reprend les transitions incompatibles de ce réseau.

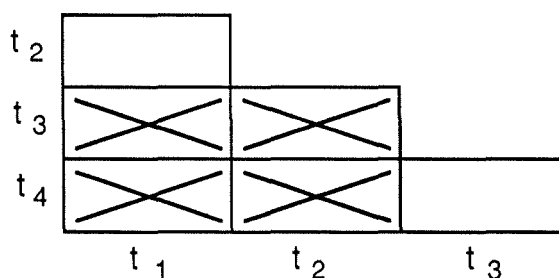


table 4 : transitions incompatibles du RPC associé au système de B. D..

L'examen de cette table nous conduit immédiatement à la partition :

$$W = [W_1 \ W_2]$$

$$\text{avec } W_1 = \{t_1, t_2\}$$

$$W_2 = \{t_3, t_4\}$$

Nous avons décomposé W en matrices factorisables et nous pouvons ainsi nous référer au paragraphe précédent. Nous y avons expliqué comment obtenir une base d'invariants d'une matrice factorisable.

En appliquant ce procédé à W_1 , nous pouvons déduire B_{W_1} . Cependant, puisque

$$\text{Ker}(W) \subsetneq \text{Ker}(W_1)$$

étant donné que

$$\text{Ker}(W) = \text{Ker}[W_1 \ W_2] \subsetneq \text{Ker}(W_1)$$

nous devons restreindre le sous-espace des invariants de W_1 pour déterminer celui de W .

Nous présentons ici un algorithme qui permet le calcul d'une base d'invariants de W en profitant de la technique de partition.

Considérons :

1) I_0 une matrice identité dont la dimension est telle que

$$I_0 \cdot W = W$$

2) $B = \{b_1, \dots, b_k\}$ une base d'invariants d'une matrice H de fonctions linéaires à laquelle nous associons la matrice B définie par

$$B = (b_1, \dots, b_k)$$

Ainsi, $B^T \cdot H = 0$.

L'algorithme est alors le suivant :

1. $Q := I_0$; $i := 1$; $H^i := W$;
2. while $H^i \neq []$ do { [] désigne la matrice nulle}
 - 2.1 partitionner la matrice H^i en vérifiant la relation d'incompatibilité des transitions :

$$H^i = [H^i_1 \dots H^i_q] \quad \text{\{où } q \text{ désigne une variable\}}$$
 - 2.2 calculer la matrice B_i^T associée à la base B_i des invariants de H^i_1
 - 2.3 $Q := B_i^T \cdot Q$; $H^{i+1} = [B_i^T \cdot H^i_2 \dots B_i^T \cdot H^i_q]$;
 - $i := i + 1$
- end of while

Il peut être démontré [4] que la matrice Q résultante est la matrice associée à une base d'invariants symboliques de W .

Pour terminer, nous présentons le résultat de l'application de cet algorithme au système de bases de données à la table 5.

	I	A	Ac	E	In	En	R	C
w1		ABS		ID				
w2		MES			ID			
w3			-ID				DEST	
w4	MES					ID	MES.DEST+ID	ID
w5	ID	ID	ID					

table 5 : base d'invariants de $W = [W_1 \ W_2]$

Nous pouvons réécrire ces invariants sous la forme :

i1) $|M(\text{attente})| + M(\text{exclusion}) = 1$

i2) $MES(M(\text{attente})) + M(\text{inutilisé}) = \sum BM$

i3) $M(\text{accomplit}) = DEST(M(\text{reçu}))$

$$i4) \text{MES}(M(\text{attente})) + M(\text{envoyé}) + (\text{MES} \cdot \text{DEST} + \text{ID}) (M(\text{reçu})) + M(\text{confirmé}) \\ = \sum \text{BM}$$

$$i5) M(\text{inactif}) + M(\text{attente}) + M(\text{accompli}) = \sum \text{GBD}$$

Ceux-ci constituent une base des invariants de la matrice d'incidence du système considéré.

L'ensemble des invariants étudiés au paragraphe précédent diffère quelque peu de celui-ci. En effet, les deux relations :

$$M(\text{inutilisé}) + M(\text{envoyé}) + M(\text{reçu}) + M(\text{confirmé}) = \sum \text{BM} \quad (1)$$

$$\text{MES}(M(\text{attente})) = M(\text{envoyé}) + M(\text{reçu}) + M(\text{confirmé}) \quad (2)$$

ne sont pas reprises ci-dessus. Cependant, il est facile de s'apercevoir que ces deux invariants peuvent s'exprimer à partir des éléments de la base découverte par l'application de l'algorithme. Ainsi,

$$(1) = w_2 + w_4 - \text{MES}(w_5) - \text{MES}(w_3)$$

$$(2) = \text{MES}(w_5) - w_4 + \text{MES}(w_3)$$

VI. Conclusion.

Après avoir montré l'avantage de la méthode de l'invariant, nous avons défini un ensemble de règles de transformation qui peuvent être utilisées pour simplifier la matrice d'incidence.

Nous avons illustré leur application à l'aide de deux exemples. Le degré de simplification obtenu par cette technique est important. L'ordre de grandeur pour les deux exemples est repris à la fig. 23.

	lignes		colonnes		éléments		élé. non nuls	
	avant	après	avant	après	avant	après	avant	après
réseau de bases de données	14	6	18	2	252	12	47	8
système du téléphone	13	11	11	4	143	44	52	14

fig. 23 : degré de simplification obtenu à l'aide des règles de transformation

Ces règles de transformation peuvent être utilisées pour découvrir les invariants mais elles peuvent aussi montrer que certaines places ne peuvent pas être couvertes par ceux-ci. Dans [9], on montre que l'existence de places non couvertes dans un réseau de Pétri implique que ce dernier est soit non borné, soit non vivant. Il serait intéressant d'étendre ces conclusions aux RPC.

La seconde méthode présentée est celle de M. Silva et al.. Ils ont montré qu'il était possible d'exprimer à partir de fonctions connues les invariants d'une matrice factorisable à partir du calcul des invariants d'un RPC simplifié.

De plus, ils présentent un algorithme qui permet d'appliquer cette technique aux matrices non factorisables.

CHAPITRE 3 - LA PROGRAMMATION LINEAIRE.

I. Introduction.

Une des caractéristiques importantes d'un réseau de Pétri se traduit par son équation fondamentale :

$$M' = M + C \cdot s$$

Dans un premier temps, nous pouvons constater l'analogie avec "l'équation d'état" bien connue en théorie des systèmes linéaires.

Cependant, cette théorie n'apporte qu'une aide assez limitée à l'analyse des réseaux de Pétri. Murata [10] a pu livrer quelques résultats concernant la contrôlabilité et l'accessibilité des réseaux de Pétri mais le fruit de ses recherches n'est pas suffisant et ne peut être exploité pratiquement.

Nous avons abordé au chapitre précédent l'algèbre linéaire qui prouve un ensemble de propositions intéressantes à propos de certaines caractéristiques des réseaux de Pétri. Mais, le fait que l'on travaille avec des entiers rend la programmation coûteuse.

La programmation linéaire a été proposée par Peterson [11] pour calculer les invariants d'un réseau. J. B. Lasserre et P. Mahey ont étendu cette technique et ont [12] prouvé leur validité pour détecter les réseaux bornés et les réseaux vivants.

Nous terminerons ce chapitre en essayant d'appliquer ces résultats aux réseaux de Pétri colorés. Nous allons profiter de la structure particulière des matrices d'incidence des RPC et appliquer la méthode de décomposition de Dantzig et Wolfe.

II. Rappels de programmation linéaire.

Le problème général de la programmation linéaire peut s'écrire :

$$\begin{array}{l} \text{opt } z = \sum c_j x_j \\ \text{sous des contraintes} \\ \text{(s.c.)} \end{array} \left| \begin{array}{l} \sum a_{ij} x_j \geq b_i \quad , i=1, \dots, p \\ \sum a_{ij} x_j \leq b_i \quad , i=p+1, \dots, q \\ \sum a_{ij} x_j = b_i \quad , i=q+1, \dots, m \\ x_j \geq 0 \quad , j=1, \dots, r \\ x_j \text{ quelconque} \quad , j=r+1, \dots, m \end{array} \right.$$

où les a_{ij} , b_i et c_j ($i=1, \dots, m$; $j=1, \dots, n$) sont des constantes données. z est appelé la fonction économique.

On peut toujours amener un programme linéaire sous la forme standard suivante :

$$\begin{array}{l} \min z = cx \quad (1) \\ \text{s.c.} \left| \begin{array}{l} Ax = b \quad (2) \\ x \geq 0 \end{array} \right. \end{array}$$

avec $c = (c_1, \dots, c_n)$
 $x = (x_1, \dots, x_n)^{-1}$
 $A = (a_{ij})$

En supposant le système (2) incompatible et indépendant, on peut écrire A sous la forme

$$A = (B \ N) \quad \text{où} \cdot B_{m \times m} \text{ est constitué de vecteurs} \\ \text{colonnes de } A \text{ linéairement} \\ \text{indépendants} \\ B = (P_1, P_2, \dots, P_m) \\ \text{avec } P_i = (a_{1i}, a_{2i}, \dots, a_{mi})^{-1} \\ \cdot N_{m \times (n-m)}$$

et transformer l'équation (2) qui devient :

$$B x_B + N x_N = b$$

$$\text{où } x_B = (x_1, \dots, x_m)^{-1} \quad \text{et} \quad x_N = (x_{m+1}, \dots, x_n)^{-1}$$

En multipliant les deux membres de l'équation par B^{-1} (qui existe puisque B est une base), on obtient :

$$x_B + B^{-1}N x_N = B^{-1}b$$

En posant,

$$\begin{aligned} \bar{b} &= B^{-1}b \\ \bar{P}_j &= B^{-1}P_j = (\bar{a}_{1j}, \bar{a}_{2j}, \dots, \bar{a}_{mj})^{-1} \end{aligned}$$

on établit l'équivalent canonique du système $Ax = b$ correspondant à la base B :

$$\begin{array}{rcl} x_1 + \dots & + \sum \bar{a}_{1j} x_j & = \bar{b}_1 \\ x_2 + \dots & + \sum \bar{a}_{2j} x_j & = \bar{b}_2 \\ & \dots & \\ & x_m + \sum \bar{a}_{mj} x_j & = \bar{b}_m \end{array} \quad (3)$$

Il lui est équivalent dans le sens où il possède les mêmes solutions.

Définitions

- Un vecteur $x \geq 0$ solution du système $Ax = b$ est appelée solution réalisable du programme linéaire (P. L.).
- Une solution réalisable du P. L. satisfaisant à (1) est appelée solution réalisable optimale du P. L..
- La solution

$$\begin{aligned} x_1 = \bar{b}_1, x_2 = \bar{b}_2, \dots, x_m = \bar{b}_m \\ x_{m+1} = 0, \dots, x_n = 0 \end{aligned}$$
 ou $(x_B = \bar{b} \text{ et } x_N = 0)$

est appelée solution de base du système $Ax = b$.

- Une variable x_i dont la valeur a été fixée à zéro est appelée variable indépendante.
Une variable x_i tel que $x_i = \bar{b}_i$ est appelée variable de base.
- Une solution de base est dégénérée si la valeur d'une ou de plusieurs variables de base est nulle.
- Une solution de base réalisable est une solution de base satisfaisant à $x \geq 0$.

- Une solution de base réalisable optimale (ici, minimale) est une solution de base réalisable qui fournit une valeur optimale (ici, minimale) finie de la fonction économique.

1. Rappels mathématiques.

Définitions.

- Un hyperplan de \mathbb{R}^n est un ensemble de points de la forme

$$H(a, d) = \{ x \in \mathbb{R}^n \mid \langle a, x \rangle = d \}$$
 où $a \neq 0$; a est son vecteur normal et $\langle a, x \rangle = d$ est son équation.

L'hyperplan $H(a, d)$ détermine deux demi-espaces fermés :

$$H^+(a, d) = \{ x \mid \langle a, x \rangle \geq d \} \quad \text{et} \\ H^-(a, d) = \{ x \mid \langle a, x \rangle \leq d \}.$$

- Une combinaison linéaire convexe de points $x^1, x^2, \dots, x^p \in \mathbb{R}^n$ est un point $x \in \mathbb{R}^n$ tel que

$$x = \sum k_j x_j, \quad k_j \geq 0, \quad \sum k_j = 1.$$
- Un sous ensemble X de \mathbb{R}^n est convexe ssi toute combinaison linéaire convexe de deux quelconques de ses points x^1 et $x^2 \in X$ se trouvent aussi dans X :

$$\forall x^1, x^2 \in X, \forall k \in [0, 1] : kx^1 + (1-k)x^2 \in X$$

On peut démontrer facilement qu'un hyperplan est un convexe. D'autre part, on peut remarquer que l'équation d'un hyperplan

$$\langle a, x \rangle = d \iff \sum a_j x_j = d \quad \text{où } a = (a_1, a_2, \dots, a_p) \\ x = (x_1, x_2, \dots, x_p)^{-1}$$

est similaire à l'équation (1) où d jouerait le rôle de z . De plus, la variation de z (de d) engendre une série d'hyperplans parallèles entre eux.

Théorème 4 : l'intersection d'un nombre quelconque d'ensembles convexes est elle-même convexe.

2. Algorithme du simplexe.

En reprenant l'équation d'un P. L. sous forme standard

$$\begin{aligned} \min z &= \sum c_j x_j \\ \sum a_{ij} x_j &= b_j & i = 1, \dots, m \\ x_j &\geq 0 & j = 1, \dots, n \end{aligned}$$

on constate que chacune des contraintes de ce P. L. définit un ensemble convexe.

En appliquant le théorème 4 et en notant X l'ensemble des points satisfaisant aux contraintes de ce P. L., on peut déduire que X est un ensemble convexe.

Résoudre un programme linéaire consiste donc à trouver dans la famille d'hyperplans celui qui a encore des valeurs dans X et qui donne à z sa valeur minimale.

C'est sur cette hypothèse qu'est basé l'algorithme du simplexe dont les différentes étapes sont données ci-dessous.

Etape 1 : mettre l'équivalent canonique associé à la base B sous forme d'un tableau, appelé tableau du simplexe (fig. 23).

La base B est supposée réalisable, d'où $\bar{b}_i \geq 0$ pour $i=1, \dots, m$.

Etape 2 : si tous les $\bar{c}_j \geq 0$, arrêter les calculs car la solution de base réalisable actuelle est un programme minimal. Sinon, aller à l'étape 3.

Etape 3 : sélectionner l'un des $\bar{c}_j < 0$, par exemple

$$\bar{c}_s = \min \bar{c}_j \quad (\text{critère d'entrée})$$

et aller à l'étape 4.

	1	x_1	x_2	\dots	x_m	x_{m+1}	\dots	x_n
-Z	$-\bar{z}_0$	0	0	\dots	0	\bar{c}_{m+1}	\dots	\bar{c}_n
x_1	\bar{b}_1	1	0	\dots	0	$\bar{a}_{1,m+1}$	\dots	\bar{a}_{1n}
x_2	\bar{b}_2	0	1	\dots	0	$\bar{a}_{2,m+1}$	\dots	\bar{a}_{2n}
.	.	.	.	\dots	.	.	\dots	.
.	.	.	.	\dots	.	.	\dots	.
.	.	.	.	\dots	.	.	\dots	.
x_m	\bar{b}_m	0	0	\dots	1	$\bar{a}_{m,m+1}$	\dots	\bar{a}_{mn}

fig. 23 : tableau du simplexe

Etape 4 : si tous les $\bar{a}_{js} \leq 0$, arrêter les calculs car la fonction économique n'a pas de minimum fini. Sinon, aller à l'étape 5.

Etape 5 : pour tous les $\bar{a}_{js} > 0$, calculer

$$\bar{b}_r / \bar{a}_{rs} = \min \bar{b}_i / \bar{a}_{is} \quad (\text{critère de sortie})$$

et aller à l'étape 6.

Etape 6 : pivoter le tableau du simplexe avec \bar{a}_{rs} pour pivot et retourner à l'étape 2.

3. Interprétation géométrique de l'algorithme du simplexe.

Définitions.

- Un point extrémal (ou point anguleux) d'un convexe $X \subset \mathbb{R}^n$ est un point $s \in X$ (à distance finie) qui ne peut être représenté comme combinaison linéaire convexe, à coefficients non nuls, de deux points distincts dans X . Autrement dit, $x \in X$ est un point extrémal de X s'il est impossible de trouver deux points $x^1, x^2 \in X$ ($x^1 \neq x^2$) vérifiant :

$$x = k x^1 + (1-k) x^2, \quad 0 < k < 1.$$

- Un tronçon de \mathbb{R}^n est l'intersection d'un nombre fini de demi-espaces fermés :

$$T = \{ x \in \mathbb{R}^n \mid \langle a^i, x \rangle \geq b_i, i=1, \dots, m \}$$

dont les hyperplans générateurs $\langle a^i, x \rangle = b_i$, ne passent pas tous par l'origine (au moins un $b_i \neq 0$).

Les contraintes d'un P. L. constituent un tronçon et il est possible de démontrer que lors d'un changement de solution de base réalisable, on se ballade sur ce tronçon en passant d'un point extrémal à un autre sauf dans le cas d'une dégénérescence.

4. Méthode lexicographique.

Lors d'une itération de l'algorithme du simplexe, il est possible que l'on soit en présence d'une dégénérescence. Dans ce cas, le changement de base ne provoquera pas de diminution de la valeur de la fonction économique puisque

$$\tilde{z}_0 = \bar{b}_r / \bar{a}_{rs} + \bar{z}_0$$

où $\bar{b}_r / \bar{a}_{rs} = \min \bar{b}_i / \bar{a}_{is}$
 \tilde{z}_0 est la nouvelle valeur de la fonction
 économique
 \bar{z}_0 sa valeur précédente

On risque alors de retomber sur une base réalisable déjà obtenue.

Pour éviter ce problème de cyclage, G. B. Dantzig, A. Order et P. Wolfe ont imaginé une méthode connue sous le nom de méthode lexicographique. Elle permet de classer les bases réalisables d'un programme linéaire selon un ordre strict de façon que celles-ci n'apparaissent qu'une seule fois au cours des itérations successives.

Définitions.

- Un vecteur v est lexicographiquement positif ($v \succ 0$) si $v \neq 0$ et si sa première composante non nulle est positive.

- Un vecteur est lexicographiquement supérieur (respectivement inférieur) à un vecteur $w \in \mathbb{R}^p$ si $v - w \succ 0$ (respectivement $w - v \succ 0$).

La méthode lexicographique est identique à l'algorithme du simplexe à ceci près que :

- toutes les lignes du tableau du simplexe, sauf éventuellement la première (correspondant à la fonction économique), constituent des vecteurs lexicographiquement positifs dans \mathbb{R}^{n+1} ;
- le critère de sortie est remplacé par le suivant : s étant l'indice de la variable entrant dans l'ensemble des variables de base, celle dont l'indice r correspond au vecteur qui, parmi les vecteurs

$$(\bar{b}_i/\bar{a}_{is}, \bar{a}_{i1}/\bar{a}_{is}, \bar{a}_{i2}/\bar{a}_{is}, \dots, \bar{a}_{in}/\bar{a}_{is}) \quad \text{avec} \begin{cases} a_{is} > 0 \\ i \in \{1, \dots, m\} \end{cases}$$

est lexicographiquement inférieur à tous les autres.

Il est possible de démontrer que cette méthode fournit une solution de base optimale après un nombre fini d'itérations et donc, empêche tout cyclage.

5. Forme révisée de la méthode du simplexe.

La méthode du simplexe possède un inconvénient évident : la place énorme requise pour stocker tous les tableaux successifs. Mais en examinant l'algorithme, on peut observer que seule la mémorisation de c , b , A et B^{-1} est nécessaire.

Cette constatation a amené G. B. Dantzig et W. Orchard-Hays à proposer la forme révisée du simplexe. Cette méthode peut s'énoncer comme suit :

Etant donné l'inverse B^{-1} de la base actuelle et la solution de base réalisable actuelle ($x_B = B^{-1}b = \bar{b}$, $x_N = 0$), effectuer les étapes suivantes :

Etape 1 : calculer le vecteur des multiplicateurs du simplexe $\Pi = c_B B^{-1}$ et le vecteur $\bar{c}_N = c_N - \Pi N$ des coûts relatifs correspondant aux variables indépendantes.

Etape 2 : si $\bar{c}_N \geq 0$, arrêter les calculs car $(x_B = \bar{b}, x_N = 0)$ est une solution optimale. Sinon, aller à l'étape 3.

Etape 3 : sélectionner, dans \bar{c}_N , un coût relatif $\bar{c}_s < 0$ (le plus négatif, par exemple) et calculer le vecteur :

$$\bar{P}_s B^{-1} P_s = (\bar{a}_{1s}, \dots, \bar{a}_{ms}).$$

Aller à l'étape 4.

Etape 4 : si $\bar{a}_{is} \leq 0$ pour tout $i \in \{1, \dots, m\}$, arrêter les calculs car la fonction économique n'a pas de minimum fini. Sinon, déterminer l'indice r tel que

$$\bar{b}_r / \bar{a}_{rs} = \min \bar{b}_i / \bar{a}_{is}.$$

Aller à l'étape 5.

Etape 5 : calculer l'inverse \tilde{B}^{-1} de la nouvelle base \tilde{B} , ainsi que la solution de base réalisable $(x_{\tilde{B}} = \tilde{B}^{-1} \bar{b} = b, x_N = 0)$. Puis retourner à l'étape 1.

III. Application de la programmation linéaire aux réseaux de Pétri.

J. B. Lasserre et P. Mahey [12] ont montré que certaines particularités des réseaux de Pétri peuvent être analysées en appliquant la programmation linéaire (de complexité polynomiale) et non de la programmation en nombres entiers. Ils ont pu prouver sa validité pour détecter les réseaux bornés et les réseaux vivants.

Leur raisonnement se base sur deux lemmes qui étendent les relations en nombres entiers aux réels.

Lemme 1 : soit W une matrice dont les coefficients sont entiers. Nous avons :

$$\begin{aligned} \Omega_0 &= \{ f : f \in \mathbb{N}^m, W^T f = 0 \} \neq \{0\} \\ &\Leftrightarrow \\ \Omega'_0 &= \{ f : f \in \mathbb{R}^m, f \geq 0, W^T f = 0 \} \neq \{0\} \end{aligned}$$

Lemme 2 : soit W une matrice dont les coefficients sont entiers. Alors

$$\begin{aligned}\Omega_1 &= \{ f : f \in \mathbb{N}^m, f > 0, W^T f \leq 0 \} \neq \emptyset \\ &\Leftrightarrow \\ \Omega'_1 &= \{ f : f \in \mathbb{R}^m, f \geq e, W^T f \leq 0 \} \neq \emptyset\end{aligned}$$

où e est un vecteur de "1".

1. Cas des réseaux bornés.

On peut montrer que :

Théorème 5 : soit $R = (P, T, \text{Pré}, \text{Post})$ un réseau de Pétri dont la matrice d'incidence est W .

R est borné \Leftrightarrow il existe $f \in \mathbb{N}^m : f > 0$ et $W^T f \leq 0$.

Si un tel f existe, nous avons alors :

$$M(p) \leq f^T M_0 / f(p) \quad , \quad \forall p \in P$$

A partir du lemme 2, on peut alors déduire :

Proposition 10 : R est borné \Leftrightarrow il existe $f \in \mathbb{R}^m : f \geq e$ et $W^T f \leq 0$

Ce qui nous conduit au corollaire suivant.

Corollaire : une borne au nombre de marques dans la place p est donnée par :

$$\begin{array}{l} \text{Min } M_0^T f \\ \left| \begin{array}{l} W^T f \leq 0 \\ f \geq e \\ f(p) = 1 \end{array} \right. \end{array}$$

qui est un programme linéaire.

2. Calcul des invariants.

L'ensemble des invariants est en fait celui défini par Ω_0 dans le lemme 1.

Théorème 6 : soit $\Omega_0 = \{ f \in N^m \text{ tq } W^T f = 0 \}$

si $f \in \Omega_0$ alors

$$\forall M \in A(R, M_0) : f^T M = F^T M_0$$

De plus, si $f(p) \neq 0$,

$$M(p) \leq M_0^T f / f(p) \quad \forall f \in \Omega'_0$$

Cette propriété sur Ω_0 est également valable sur Ω'_0 . Ainsi, une borne sur $M(p)$ peut être obtenue à l'aide du programme linéaire suivant :

$$\begin{array}{l} \text{Min } M_0^T f \\ \left| \begin{array}{l} W^T f = 0 \\ f(p) = 1 \\ f \geq 0 \end{array} \right. \end{array}$$

3. Cas des réseaux vivants.

Une condition nécessaire de vivacité peut être étendue à Ω'_0 comme pour les réseaux bornés.

Proposition 11 : si (R, M) est vivant, alors

$$\forall f \in \Omega'_0, f^T M \geq \text{Pré}(\cdot, t) \quad \forall t = 1, \dots, n$$

Pour vérifier cette condition, il suffira de résoudre les n programmes linéaires :

$$\begin{array}{l} \text{Min } f^T (M - \text{Pré}(\cdot, t)) \quad t = 1, \dots, n \\ \left| \begin{array}{l} W^T f = 0 \\ f \geq 0 \end{array} \right. \end{array}$$

Comme nous l'avons vu, certaines propriétés des réseaux de Pétri peuvent être étudiées à l'aide de la programmation linéaire. L'intérêt de cette technique réside dans le fait qu'elle permet de manipuler des réseaux de Pétri de grande taille.

IV. Principe de décomposition de Dantzig et Wolfe. [13]

1. Introduction.

L'aide à la résolution de problèmes fournie par la programmation linéaire n'est plus à démontrer. Cependant le volume des calculs naissants croît rapidement en fonction du nombre de contraintes, du nombre de coefficients de la matrice du problème et de la structure de cette matrice.

Heureusement, parmi les matrices de grande taille qui sont à considérer, un grand nombre est composé de blocs de coefficients le long de la diagonale principale ainsi que de lignes et de colonnes reliant ces blocs, les autres parties de la matrice étant composées de coefficients nuls. Une telle structure est représentée à la fig. 24.

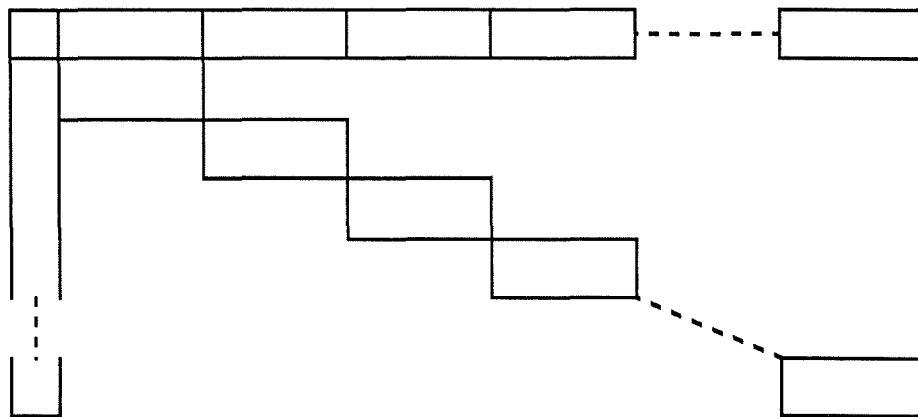


fig. 24 : structure d'une matrice angulaire par blocs

Les blocs situés sur la diagonale principale sont appelés les contraintes de liaison et ceux constituant les lignes et les colonnes de la matrice, les variables de liaison. On dit que la matrice est angulaire par blocs.

Deux approches différentes ont été proposées pour résoudre les problèmes de programmation linéaire comportant ce type de matrices. Il s'agit du partitionnement et de la décomposition.

Dans la première technique, comme dans celle du simplexe, on se ballade d'un point extrémal à un autre du tronçon des programmes alors que dans la deuxième, on reste à l'intérieur du tronçon.

C'est cette dernière approche qui a été développée par Dantzig et Wolfe et qui va être exposée dans les paragraphes suivants.

2. Domaine d'application.

Ce principe peut être appliqué aux problèmes dont l'ensemble des contraintes peut être partitionné en $(p+1)$ sous-ensembles. Parmi ceux-ci, p sont des systèmes mutuellement indépendants, c'est-à-dire qu'ils font intervenir des inconnues différentes et les contraintes du $(p+1)^{\text{ième}}$ sont des contraintes de liaison affectant toutes les variables.

On remplace ainsi la résolution d'un problème par celle de $(p+1)$ problèmes. Cette technique est surtout efficace quand elle engendre des systèmes dont la structure est simple.

Le cas intéressant où $p = 1$ montre que ce principe de décomposition peut s'appliquer à n'importe quel problème de programmation linéaire et le scinder en deux. On peut obtenir ainsi des systèmes aussi petits que l'on veut. Ceci est non seulement intéressant si la taille du problème initial interdit sa résolution d'un seul bloc mais aussi si on aboutit à un sous-problème possédant une structure simple.

3. Formulation du problème original et d'un problème équivalent.

Les matrices considérées au paragraphe précédent sont composées de blocs d'éléments non nuls disposés selon la figure qui suit.

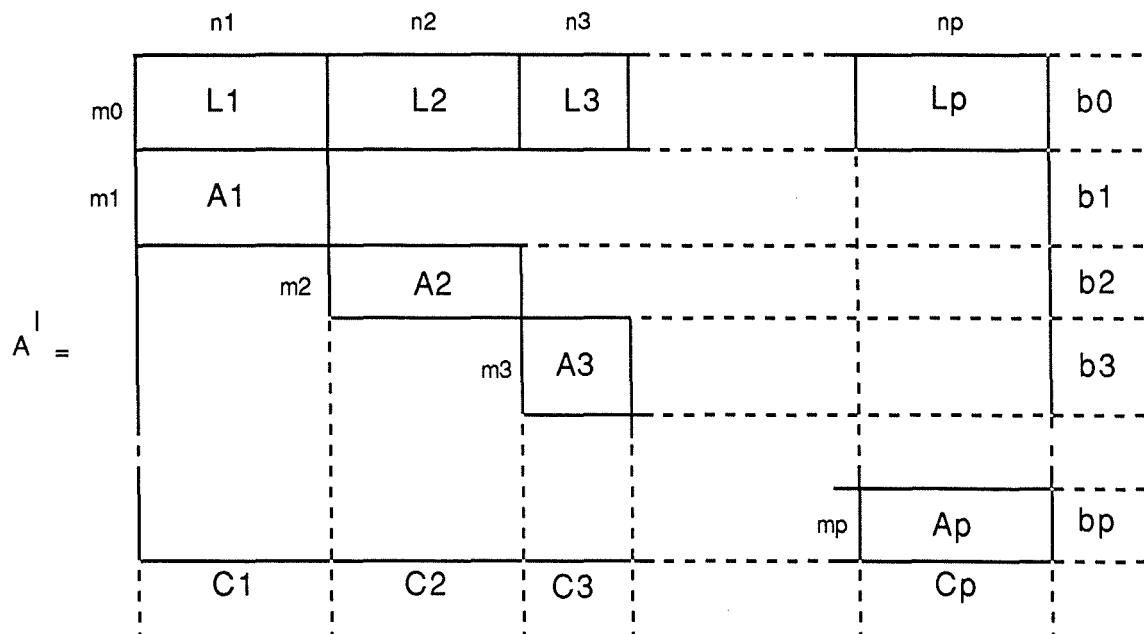


fig. 25 : composition d'une matrice angulaire par blocs

La matrice "de liaison" L correspond aux m_0 premières lignes de la matrice des contraintes. Elle est constituée de p sous-matrices L_j ($j=1, \dots, p$) ayant les mêmes indices de colonne que les matrices A_j des p problèmes indépendants. Certaines des A_j peuvent ne pas exister et la matrice L peut être partiellement creuse.

Les vecteurs b et c sont partitionnés respectivement horizontalement et verticalement comme la matrice A^I .

En décomposant, on obtient le système suivant :

$$\begin{aligned} \min z &= \sum c_j x_j \\ &\left| \begin{array}{l} \sum L_j x_j = b_0 \\ A_j x_j = b_j \\ x_j \geq 0 \end{array} \right. \quad j=1, \dots, p \end{aligned} \quad (1)$$

qui comporte $\sum m_j$ équations à $\sum n_j$ inconnues. On peut observer que chacun des systèmes

$$\begin{aligned} A_j x_j &= b_j \\ x_j &\geq 0 \end{aligned}$$

définit un tronçon S_j dans l'espace R_j^n des solutions. Si l'on suppose que tous ces tronçons S_j sont bornés, ce qui signifie qu'il s'agit de polyèdres convexes, on peut appliquer le théorème de représentation énoncé ci-dessous.

Théorème de la représentation : tout point d'un ensemble convexe fermé et borné $X \in R^n$ peut s'exprimer comme combinaison linéaire convexe d'un nombre fini de points extrémaux de X .

En notant x_{ji} ($i=1, \dots, s_j$) les points extrémaux de S_j , nous pouvons alors écrire :

$$\forall x_j \in S_j : x_j = \sum k_{ji} x_{ji} \quad , \quad k_{ji} \geq 0 \quad , \quad \sum k_{ji} = 1$$

Et en posant, $\forall j$

$$\begin{aligned} l_{ij} &= L_j x_{ji} \\ c_{ij} &= C_j x_{ji} \end{aligned} \quad i = 1, \dots, s_j$$

nous pouvons transformer le système (1) et obtenir :

$$\begin{aligned} \min z &= \sum \sum c_{ji} k_{ji} \\ &\left\{ \begin{array}{l} \sum \sum k_{ji} l_{ji} = b_0 \\ \sum k_{ji} = 1 \\ k_{ji} \geq 0 \end{array} \right. \quad j = 1, \dots, p \end{aligned} \quad (2)$$

Ce système ne comporte plus que $(m_0 + p)$ équations. Le coût à payer pour cette diminution est une augmentation du nombre de variables qui passe de $\sum n_j$ à $\sum s_j$. Cependant, le volume des calculs ne dépend qu'assez peu du nombre de variables à condition que l'évaluation de toutes les quantités l_{ji} et c_{ji} ne soit pas nécessaire.

Nous allons montrer qu'à chaque itération, seul le calcul du vecteur l_{ji} (et le coefficient c_{ji}) qui "entre" à cette itération doit être effectué.

La matrice du problème (2) est représentée à la fig. 26.

	s1	s2	s3	...	sp	
m0	l11 l22...l1s1	l21 l22...l2s2	l31...l3s3		lp1...lpsp	b0
1	1 1 ... 1					1
A ⁻¹ =		1	1 1 ... 1			1
			1 ... 1			1
					1	1 ... 1
	c11...c1s1	c21...c2s2	c31...c3s3		cp1...cpsp	

fig. 26 : matrice associée au système (2)

4. Algorithme de décomposition.

Nous utilisons la forme révisée du simplexe qui ne nécessite que la mémorisation de l'inverse de la base et des données du problème initial.

Décrivons une itération de la résolution.

Soient B la base correspondant à l'itération précédente, ($x_B = B^{-1}\bar{b} = \bar{b}$, $x_N = 0$) la solution de base réalisable actuelle et $c^B = (c_{ij})$ le vecteur des coûts dont les composantes ont les indices des colonnes de B .

A l'aide des éléments conservés en mémoire, nous pouvons calculer $\Pi = c_B B^{-1}$ et le décomposer en distinguant les sous-vecteurs Π^0 formé par les m_0 premières composantes de Π et $\Pi^1 = (\Pi^1_1, \Pi^1_2, \dots, \Pi^1_p)$ formé par les p dernières composantes de Π . Nous pouvons ensuite évaluer le vecteur $\bar{c}_N = c_N - \Pi N$ des coûts relatifs correspondant aux variables indépendantes qui peut se réécrire :

$$\bar{c}_{ji} = c_{ji} - \Pi^0 l_{ji} + \Pi^1$$

Si $\bar{c}_{ji} \geq 0$, nous pouvons arrêter les calculs car ($x_B = \bar{b}$, $x_N = 0$) est une solution optimale. Sinon, nous devons effectuer un changement de base en substituant à un vecteur de B l'un des l_{ji} vérifiant $c_{ji} < 0$. En pratique, le vecteur l_{kr} "entrant" dans la base est déterminé par la relation :

$$\bar{c}_{kr} = \min \bar{c}_{ji} < 0$$

Le critère de sortie usuel du simplexe permet de déterminer le vecteur qui "sort" de B pour être remplacé par I_{kr} .

Nous pouvons déduire de ce qui précède qu'à chaque itération de l'algorithme du simplexe, il suffit de connaître le minimum de \bar{c}_{ji} sur l'ensemble des colonnes j_i de la matrice A'' .

De plus, déterminer

$$\min (c_j - \Pi^0 L_j) x_{ji}$$

revient à résoudre le programme linéaire suivant :

$$\begin{array}{l} \min (c_j - \Pi^0 L_j) x_j \\ \left\{ \begin{array}{l} A_j x_j = b_j \\ x_j \geq 0 \end{array} \right. \end{array} \quad (3)$$

Nous savons en effet, que le minimum est toujours atteint en un point extrémal x_{ji} de S_j .

La solution de ce programme est tel que $r_j \in \{1, \dots, s_j\}$ et que

$$(c_j - \Pi^0 L_j) x_{jr_j} = \min [(c_j - \Pi^0 L_j) x_{ji}]$$

Il suffit ensuite de déterminer l'indice k tel que :

$$(c_k - \Pi^0 L_k) x_{kr_k} + \Pi^1_k = \min [(c_j - \Pi^0 L_j) x_{jr_j} + \Pi^1_j]$$

Nous obtenons alors la valeur de \bar{c}_{kr_k} et il ne reste plus qu'à la comparer avec la valeur nulle.

Si $\bar{c}_{kr_k} \geq 0$, la solution actuelle x_{ji} est minimale et nous pouvons en déduire une solution x_j au programme initial. Dans le cas inverse, il est nécessaire d'effectuer un changement de base.

L'algorithme de décomposition se résume en fait, à une application du simplexe (sous forme révisée) au problème (2) dont les coefficients (l_{ji}, c_{ji}) restent en majorité inconnus, seuls étant calculés, à chaque itération, ceux qui sont nécessaires pour le choix du vecteur qui entre dans la base.

5. Extension au cas où les S_j ne sont pas tous bornés.

Définitions. [14]

- Soit S un ensemble convexe fermé de \mathbb{R}^n . Un vecteur non nul de \mathbb{R}^n est appelé direction de S si

$$\forall x \in S : x + k d \in S, \quad \forall k \geq 0$$

- Deux directions d_1 et d_2 de S sont distinctes si

$$d_1 \neq k d_2 \quad \forall k > 0$$

- Une direction d de S est appelée direction extrême si elle ne peut pas être décrite sous forme d'une combinaison linéaire positive de deux directions distinctes i.e. si

$$d = v d_1 + w d_2 \quad \text{avec } v, w > 0$$

alors

$$d_1 = k d_2 \quad \forall k > 0.$$

Théorème 7 : soit $S = \{x \in \mathbb{R}^n \text{ tq } Ax = b ; x \geq 0\} \neq \emptyset$

où A est une matrice $m \times n$ de rang m .

Soient x_1, \dots, x_k les points extrémaux de S

d_1, \dots, d_l les directions extrémales de S

alors

$$x \in S \Leftrightarrow x = \sum k_j x_j + \sum m_j d_j$$

avec $\sum k_j = 1$

$$k_j \geq 0 \quad j = 1, \dots, k$$

$$m_j \geq 0 \quad j = 1, \dots, l$$

Ainsi, tout point x_j de S_j peut s'écrire sous la forme :

$$x_j = \sum k_{ji} x_{ji} + \sum q_{ji} d_{ji}$$

$$\text{avec } \sum k_{ji} = 1$$

$$k_{ji} \geq 0$$

$$q_{ji} \geq 0$$

En posant, $\forall j$

$$r_{ji} = L_j d_{ji} \quad \text{et} \quad v_{ji} = c_j d_{ji} \quad i = 1, \dots, t_j$$

le programme (2) s'écrit maintenant

$$\begin{array}{l} \min z = \sum \sum c_{ji} k_{ji} + \sum \sum v_{ji} d_{ji} \\ \left| \begin{array}{l} \sum \sum k_{ji} l_{ji} + \sum \sum q_{ji} r_{ji} = b_0 \\ \sum k_{ji} = 1 \\ k_{ji} \geq 0 \\ q_{ji} \geq 0 \end{array} \right. \begin{array}{l} j = 1, \dots, p \\ j = 1, \dots, p ; i = 1, \dots, s_j \\ j = 1, \dots, p ; i = 1, \dots, t_j \end{array} \end{array} \quad (2')$$

On peut montrer [13] que , moyennant une règle supplémentaire, l'algorithme de décomposition reste valable. Cette règle peut s'énoncer comme suit :

si l'optimisation d'un sous-problème (3) conduit à une solution optimale infinie définissant une direction extrême t_{ji} , introduire dans la base du problème (2') la colonne r_{ji} et le coût v_{ji} correspondant définis par :

$$r_{ji} = L_j d_{ji} \quad \text{et} \quad v_{ji} = c_j d_{ji} \quad i = 1, \dots, t_j$$

6. Application aux réseaux colorés.

Lorsque l'on désire appliquer l'algorithme exposé ci-dessus aux réseaux de Pétri colorés, il est nécessaire, dans un premier temps, de transformer la matrice d'incidence en une matrice angulaire par blocs. Il est d'ailleurs évident que la symétrie présente dans les RPC le permet.

Il est alors possible de réduire le nombre de contraintes des trois programmes linéaires proposés pour étudier les propriétés des réseaux. Réexaminons ces trois cas.

6.1 Cas des réseaux bornés.

Le programme proposé pour déterminer une borne au nombre de marques de la place p était le suivant :

$$\begin{array}{l} \text{Min } M_0^T f \\ \left| \begin{array}{l} W^T f \leq 0 \\ f \geq e \\ f(p) = 1 \end{array} \right. \end{array}$$

Nous pouvons observer qu'une des variables du problème étant fixée, nous allons travailler avec une matrice de taille réduite. La ligne correspondant à la place p pourra effectivement être supprimée.

Comme nous pouvons le remarquer, les contraintes de ce PL constituent un ensemble convexe non borné. Il sera donc nécessaire d'appliquer la règle supplémentaire de l'algorithme nécessaire dans de telles circonstances.

6.2 Calcul des invariants.

Dans ce cas, les contraintes définissent un ensemble convexe fermé et borné. Il suffit donc d'utiliser l'algorithme introduit par Dantzig et Wolfe dans sa version générale avec une matrice dont la dimension sera également diminuée.

6.3 Cas des réseaux vivants.

A nouveau, nous pouvons observer que les contraintes de ce PL constituent un ensemble convexe fermé. L'algorithme dans sa première version pourra donc être appliqué sans oublier de travailler avec une matrice angulaire par blocs.

7. Conclusion.

Il est évident que cet algorithme n'est conseillé que pour des matrices de grande taille ou pour des modèles utilisés un grand nombre de fois. Dans le cas contraire, le faible gain occasionné par cette technique risque d'être compensé par l'effort supplémentaire requis par sa mise en oeuvre.

Comme les programmes linéaires proposés se rapportent à des propriétés locales des réseaux (une place ou une transition), on peut penser que pour des

réseaux de taille importante, c'est-à-dire ceux pour lesquels les RPC ont été notamment suggérés, cette méthode est tout indiquée.

CONCLUSION.

Les réseaux de Pétri constituent un outil de modélisation utile. Malheureusement, lorsque la taille du système s'accroît, le réseau devient dense, peu clair et difficile à analyser.

Les réseaux de Pétri colorés complètent ce modèle et permettent l'étude de systèmes plus complexes. Ils conduisent à des représentations beaucoup plus claires et plus compactes. D'un autre côté, leur analyse s'avère difficile.

Le but de ce travail était de parcourir quelques méthodes d'analyse des réseaux de Pétri et d'étudier leurs extensions possibles aux réseaux colorés. Les deux techniques basées sur l'algèbre linéaire appliquées à des exemples simples semblent efficaces. Malheureusement, les règles de transformation paraissent complexes et ne permettent pas d'établir tous les invariants. Même si M. Silva et al. affirment que leur méthode a été appliquée avec succès à la validation de réseaux de taille plus importante, la technique la plus prometteuse nous semble être la programmation linéaire.

Comme nous l'avons montré, cette approche est tout indiquée pour découvrir les propriétés d'un réseau de Pétri. De plus, nous pouvons profiter de la structure particulière des matrices d'incidence des réseaux colorés pour diminuer le nombre de contraintes.

Cependant, même si le gain d'espace mémoire s'annonce important, il n'est pas évident que les temps de calcul soient inférieurs. En effet, certaines fonctions complexes pourraient rendre les calculs longs et difficiles.

Il serait intéressant d'approfondir ce problème. Dans un premier temps, on pourrait penser à quelques applications pratiques afin de comparer la résolution de programmes linéaires construits à partir d'un réseau de Pétri et d'un réseau coloré correspondant au même système.

REFERENCES.

- [1] G. W. Brams, Réseaux de Pétri : théorie et pratique, Masson (1983)
- [2] J. L. Peterson, A note on Colored Petri nets, *Information Processing Letters*, **11** (1) (1980), 40-43
- [3] K. Jensen, Coloured Petri Nets and the Invariant Method, *Theoretical Computer Science*, **14** (1981), 317-336
- [4] M. Silva, J. Martinez, P. Ladet and H. Alla, Generalized Inverses and the Calculation of Symbolic Invariants for Coloured Petri Nets, *Technique et Science Informatiques*, **4** (1) (1985), 113-126
- [5] K. Jensen, How to Find Invariants for Coloured Petri Nets, *Mathematical Foundations of Computer Science*, Lectures notes in Computer Science, **118**, Springer Verlag, Berlin (1981), 327-338
- [6] J. L. Peterson, Petri Nets, *Computing Surveys*, **9** (3) (1977), 223-252
- [7] S. L. Campbell and C. D. Meyer, Generalized Inverses of Linear Transformations, Pitman, London (1979)
- [8] H. Alaiwan, J. M. Toudic, Recherche des Semi-Flots, des Verrous et des Trappes dans les Réseaux de Pétri, *Technique et Science Informatiques*, **4** (3), (1985), 103-112
- [9] G. Memmi and R. Roucairol, Linear Algebra in Net Theory, *Advanced Course on General Net Theory of Processes and Systems*, Lectures Notes in Computer Science, **84**, Springer Verlag (1980), 213-223
- [10] T. Murata, State Equations, Controllability and Maximal Matchings of Petri Nets, *I.E.E.E. Trans. Automatic Control*, **22** (3) (1977), 412-416
- [11] J. L. Peterson, Petri Net Theory and the Modeling of Systems, Prentice Hall (1981)

- [12] J. B. Lasserre and P. Mahey, Using Linear Programming in Petri Net Analysis, *Recherche Opérationnelle*, **23** (1) (1989), 43-50
- [13] M. Simmonard, *Programmation Linéaire*, Dunod (1962)
- [14] Mokhtar S. Bazaraa and C. M. Shetty, *Non Linear Programming*, John Wiley and sons (1979).