



UNIVERSITÉ de CAEN/BASSE-NORMANDIE
U.F.R. : SCIENCES CAEN

Réseaux de Petri

Cours EL401T2

Master 1A Mention EEA Parcours AEII

G. Scorletti et G. Binet

Maîtres de conférences à l'Université de Caen

GREYC AUTO

6 bd du Maréchal Juin, F14050 Caen cedex

Tel : 02 31 45 27 12

e-mail : scorletti@unicaen.fr

©Gérard Scorletti, France, 2006

Page web : http://www.greyc.ensicaen.fr/EquipeAuto/Gerard_S/mait_Petri.html

20 juin 2006

Table des matières

1	Introduction	1
1.1	Systèmes et modèles	1
1.1.1	Notions générales sur les systèmes et modèles	1
1.1.2	Un modèle, pour quoi faire ?	4
1.2	Quelques classes de systèmes à variables discrètes	5
2	Modélisation par Réseaux de Petri	9
2.1	Modèle de base	9
2.1.1	Elements de base	9
2.1.2	Evolution d'un RdP	12
2.1.3	Exemples de Réseaux de Petri	15
2.1.4	Réseaux de Petri avec une structure particulière	17
2.2	Structures fondamentales pour la modélisation des systèmes	20
2.2.1	Elements de modélisation	20
2.2.1.1	Parallélisme	20
2.2.1.2	Synchronisation	22
2.2.1.3	Partage de ressources	25
2.2.1.4	Mémorisation	28
2.2.1.5	Lecture	28
2.2.1.6	Capacité limité	28
2.2.2	Modélisation structurée	30
2.2.2.1	Approche par affinements successifs	30
2.2.2.2	Approche par compositions de RdPs	31
3	Propriétés des Réseaux de Petri (ordinaires)	35
3.1	Notations et définitions	35
3.2	Propriétés	37
3.2.1	Réseau de Petri borné et Réseau sauf	37
3.2.2	Vivacité et blocage	37
3.2.3	Conflits	40
3.3	Invariants	41
3.3.1	Composante conservative	41
3.3.2	Composante répétitive	42
4	Etude des propriétés des Réseaux de Petri par graphes et arborescences de marquages	43
4.1	Introduction	43
4.2	Construction de l'arborescence de couverture	45
4.3	Propriétés et intérêts de l'arborescence (graphe) de couverture	45

5	Etude des propriétés des Réseaux de Petri par algèbre linéaire	47
5.1	Notations et définitions	47
5.1.1	Structure	47
5.1.2	Marquage, évolution	49
5.2	Equation fondamentale	52
5.3	Pondération des places et invariants de marquage	54
5.3.1	Pondérations	54
5.3.2	Invariants de marquage	55
5.4	Composantes répétitives	57
5.5	Remarque sur le lien entre RdP ordinaire et RdP généralisé	58
6	Etude des propriétés des Réseaux de Petri par réduction	59
6.1	Règles de réduction préservant la vivacité et la bornitude	59
6.1.1	Règle de réduction R1 : substitution de places	59
6.1.2	Règle de réduction R2 : place implicite	62
6.1.3	Règle de réduction R3 : transition neutre	64
6.1.4	Règle de réduction R4 : transitions identiques	64
6.1.5	Exemple de mise en œuvre des différentes règles R1 à R4	65
6.2	Règles de réduction préservant les invariants de marquage	66
6.2.1	Règle de réduction Ra : transition impure	66
6.2.2	Règle de réduction Rb : transition pure	67
6.2.3	Exemple de mise en œuvre des règles Ra et Rb	67
6.2.4	Remarques	67
7	Réseaux de Petri colorés	69
7.1	Introduction aux RdPs colorés à travers des exemples	70
7.2	Réseaux de Petri colorés (simplifiés)	75
7.2.1	Définition	75
7.2.2	Evolution du marquage d'un RdP coloré	76
7.2.3	Analyse d'un RdP coloré marqué	77
7.3	Un exemple d'application à la modélisation de systèmes de production	77
7.3.1	Modélisation d'un convoyeur de type FIFO	77
7.3.2	Modélisation d'une machine sur un convoyeur FIFO	79
8	Réseaux de Petri non autonomes	83
8.1	Réseaux de Petri synchronisés (RdPS)	83
8.1.1	Événements associés à une variable logique	83
8.1.2	Définition d'un Réseau de Petri Synchronisé	84
8.1.3	Propriétés des RdPs synchronisés	86
8.2	Réseaux de Petri temporisés	92

Documents de référence

Ce document de cours suit fortement la présentation et les résultats présentés dans le livre [2] disponible à la bibliothèque universitaire (BU) de Caen. Il est néanmoins aussi basé sur

- les livres [1, 4] et pour la partie Réseaux de Petri colorés [3] eux aussi disponibles à la BU;
- le document de cours de Robert Valette disponible sur la page WEB :
<http://www.laas.fr/~robert/>
- l'ensemble des documents disponibles sur la page WEB consacrée aux Réseaux de Petri :
<http://www.daimi.au.dk/PetriNets/>

Chapitre 1

Introduction

Les systèmes technologiques, de plus en plus souvent d'une grande complexité, qui ont envahi notre quotidien sont l'œuvre des ingénieurs. Pour l'ingénieur, se posent alors plusieurs problèmes importants : comment réussir à appréhender le comportement de ses systèmes afin de les concevoir, les réaliser et/ou les commander ? La description du comportement attendu du système constitue le cahier des charges. Celui-ci est en général défini par différents intervenants, intéressés par les aspects fonctionnels du produit, les besoins du consommateurs, les contraintes de coût, le marketing, etc... Du fait de la complexité de plus en plus forte des systèmes technologiques, il apparaît de plus en plus nécessaire de disposer de méthodes et d'outils de conception, de réalisation et/ou de commande qui soient particulièrement efficaces. Au centre de ces méthodes et de ces outils, se trouve en général la modélisation.

L'objectif de ce cours est de présenter la modélisation par Réseaux de Petri. Les Réseaux de Petri ont été développés pour permettre la modélisation de classes importantes de systèmes qui recouvrent des classes de systèmes de production, de systèmes automatisés, de systèmes informatiques et de systèmes de communication, pour n'en citer que quelques-uns, afin de permettre leur conception, leur évaluation et leur amélioration.

Dans ce chapitre d'introduction, les notions de système et de modèle sont rappelées : la modélisation par Réseaux de Petri est mise en perspective avec les outils de modélisation précédemment étudiés. L'intérêt de la modélisation est ensuite discutée. Le chapitre se termine sur une discussion succincte sur les systèmes auxquels les Réseaux de Petri peuvent être appliqués.

1.1 Systèmes et modèles

1.1.1 Notions générales sur les systèmes et modèles

Un **système** est une portion de la réalité définie par une *frontière* organisée en fonction d'un but. En général, un système est constitué par un ensemble d'éléments en interaction dynamique. Il possède des **entrées** et des **sorties**. Elles sont caractérisées par des **variables** de flux.

Exemple distributeur de boissons

Flux	Entrées	Sorties
Produit		
Energie		
Information		

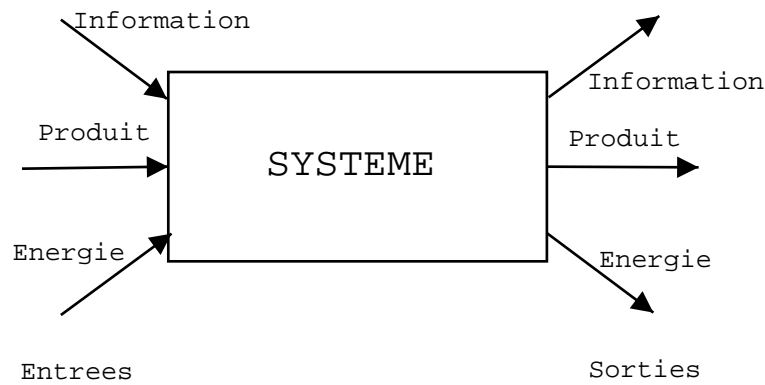


FIG. 1.1 – Système

Un **modèle** est une représentation, souvent en termes mathématiques, des relations qui existent entre les variables de sortie et les variables d'entrée d'un système. Cette représentation nécessite souvent l'introduction de variables supplémentaires appelées **variables d'état**. A un instant donné, la valeur des variables d'état caractérise un système : leur connaissance avec celle des valeurs dans le futur des variables d'entrée permet à l'aide du modèle de déterminer les valeurs des variables de sortie dans le futur.

Exemple distributeur de boissons (suite) Les variables d'état qui vont caractériser le distributeur de boissons sont

-
-
-

Un **scénario** est constitué par un instant initial t_0 , un instant final $t_1 > t_0$, la valeur des variables d'état à l'instant initial t_0 et par la valeur des variables d'entrée pour chaque instant appartenant à l'intervalle $[t_0, t_1]$.

Un modèle serait **idéal** s'il est capable de reproduire de façon exacte le comportement du système. Cela signifie que si, à un instant donné, on connaît la valeur des variables d'état du système et que si on connaît la valeur des variables d'entrée dans le futur (et le présent), les relations mathématiques qui constituent le modèle permettent de déterminer exactement les valeurs des variables de sortie qui peuvent être observées expérimentalement. Un des principes fondamentaux (et philosophiques) des Sciences de l'Ingénieur est qu'*il est impossible d'obtenir le modèle idéal d'un système*. Au mieux, on peut obtenir un "bon" modèle, c'est-à-dire, un modèle qui permette de déterminer de façon approchée les valeurs des variables de sortie qui peuvent être observées expérimentalement.

Exemple du Réservoir Le système Réservoir est représenté Figure 1.2. Le réservoir rempli d'une hauteur $h(t)$ de liquide est alimenté par un débit $d_{entree}(t)$ de liquide en haut du réservoir. Le liquide s'échappe à la base par une vanne pour laquelle il est possible de modifier la section de l'ouverture en agissant sur la valeur de la variable $v(t)$. Il s'échappe avec un débit $d_{sortie}(t)$.

- variables d'entrée :
- variables de sortie :
- variables d'état :

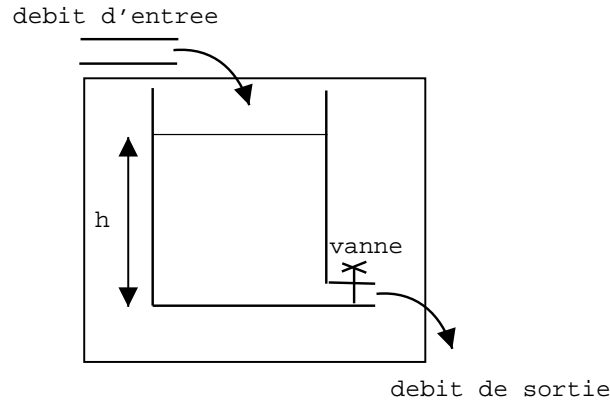


FIG. 1.2 – Système Réservoir

La variable de sortie, c'est-à-dire le débit qui s'échappe, peut s'exprimer en fonction de $h(t)$ et de $v(t)$:

$$d_{sortie}(t) = 2v(t)\sqrt{h(t)}. \quad (1.1)$$

La variation de la hauteur d'eau est reliée aux débits d'entrée et de sortie :

$$S \frac{dh(t)}{dt} = d_{entree}(t) - d_{sortie}(t) \quad (1.2)$$

où S est la section du réservoir. Les équations (1.1) et (1.2) constituent le modèle du système Réservoir.

Les variables d'un système peuvent être de nature différente, par exemple :

continues définies sur \mathbb{R} (par exemple, pour le réservoir,

discrètes définies sur \mathbb{N} (par exemple, pour le distributeur,

logiques définies sur $\{0, 1\}$ (par exemple, pour le distributeur,

Les variables varient en fonction du temps : ce sont donc des signaux. On peut définir différents types de signaux, par exemple :

signaux en temps continu :

$$\begin{aligned} \mathbb{R}^+ &\rightarrow \mathbb{R} \\ t &\mapsto s(t) \end{aligned}$$

signaux en temps discret : avec T_s la période d'échantillonnage

$$\begin{aligned} \mathbb{N} &\rightarrow \mathbb{R} \\ k &\mapsto s(kT_s) \end{aligned}$$

Suivant la nature des variables d'entrée, de sortie et d'état, les outils de modélisation utilisés sont différents. Dans le tableau qui suit, est présenté un ensemble non exhaustif d'outils de modélisation précédemment étudiés.

Les entrées, sorties, états sont des variables	Outils de modélisation	Cours / Diplôme
Continues en temps continu	Equations différentielles, Transformée de Laplace	Automatique Licence 3A
Continues en temps discret	Equations de récurrence, Transformée en Z	TNS, Automatique Master 1A
<i>Logiques</i>	Machines de Mealy & de Moore	Logique séquentielle Licence 3A
Discrètes	Réseaux de Petri	Réseaux de Petri Master 1A

Les Réseaux de Petri ont été inventés par Carl Maria Petri au début des années soixante. Des travaux ultérieurs ont permis de développer les Réseaux de Petri comme un outil de modélisation des systèmes à variables d'entrée, de sortie et d'état discrètes. C'est aussi un outil de modélisation pour les systèmes à variables logiques puisque ceux-ci sont un cas particulier des systèmes à variables discrètes.

Un caractère très intéressant est que les modèles Réseaux de Petri sont sous la forme d'une représentation mathématique *graphique*. Ce point est important car le fait d'écrire sous forme graphique un modèle plutôt que sous forme d'équations peut permettre de le rendre lisible par des personnes dont la formation scientifique n'est pas forcément poussée. Le modèle Réseaux de Petri a d'ailleurs donné naissance au *Grafcet*, un langage de spécification et de programmation d'automates industriels. Dans le langage Grafcet, le caractère graphique est ici fondamental car ce langage doit être accessible à la fois à l'industriel, l'ingénieur, le technicien et l'ouvrier car c'est un outil graphique. Les automates et le Grafcet seront étudiés en Master Professionnel 2A.

1.1.2 Un modèle, pour quoi faire ?

L'utilisation de modèles prend une part de plus en plus importante dans les projets technologiques menés par les ingénieurs, que ce soit pour la définition de systèmes, leur conception, leur réalisation et même leur intégration (mise en œuvre), leur maintenance et leur exploitation.

- Par définition, *un modèle est une représentation d'un système* : il peut ainsi permettre de décrire de façon non équivoque un système technologique. De sa définition à son exploitation, un système technologique va être appréhendé par différentes personnes (ingénieurs, etc..) sous des points de vue différents (conception, réalisation, maintenance, exploitation, etc..). Le passage d'une personne à l'autre doit alors se faire sans perte d'information, ce que permet l'utilisation d'un modèle. Ce problème est suffisamment central pour qu'un énorme effort soit consacré au niveau de la définition et de la mise en œuvre de normes et de standards industriels. Par exemple, la modélisation du fonctionnement demandé à un automate industriel est possible par l'utilisation du Grafcet, qui est un outil de modélisation dérivant des Réseaux de Petri. Le langage Grafcet a fait l'objet de plusieurs normes.
- Lors de la conception d'un nouveau système technologique, le cahier des charges exprime le comportement attendu du système. La question fondamentale est de garantir que le système qui a été conçu (sur le papier) remplit bien le cahier des charges, voire possède un comportement "satisfaisant". L'approche traditionnelle de la conception d'un système consiste à le concevoir sur le papier, à le réaliser et à faire des expériences sur le système correspondant à des scénarios types afin de vérifier si son comportement est satisfaisant et s'il est nécessaire de l'améliorer voire de le re concevoir. Cette approche pose plusieurs problèmes. Dans certains cas, il est impossible de réaliser des expériences sur le système

(par exemple, systèmes spatiaux). Parfois, cela peut être dangereux si des informations manquent sur son comportement possible. De façon plus courante, une telle approche est longue et coûteuse, ce que permettent de moins en moins les contraintes économiques. Une alternative est de définir le système technologique à l'aide d'un modèle. En plus de l'avantage vu précédemment, pour un scénario donné, un modèle permet en général de calculer numériquement les valeurs des variables d'état et de sortie, ce qui est plus court et plus économique que de les mesurer au cours d'une expérience. Cela est désigné par le terme de *simulation*. La simulation permet ainsi dans une certaine mesure de tester si le comportement du système est satisfaisant et de mettre en évidence certains problèmes. Elle ne remplace pas complètement l'expérimentation qu'il est nécessaire d'effectuer si à l'issue de la simulation le comportement du système semble satisfaisant.

- Le problème de la simulation est que, même si le modèle est bon, on ne peut tester qu'un nombre limité de scénarios. Si le choix des scénarios types n'est pas pertinent par rapport à l'ensemble des scénarios auxquels sera confronté le système durant son existence, il est difficile de prévoir si le comportement du système sera satisfaisant dans tous les cas. Une alternative est de parfois pouvoir garantir qu'à partir d'une propriété du modèle, le modèle étant supposé bon, le comportement du système sera (ou non) satisfaisant pour une famille de scénarios. On parle d'*analyse*. L'analyse d'un système repose sur l'étude des propriétés mathématiques de son modèle.

Exemple d'analyse On considère un système en temps discret d'entrée u et de sortie y , u et y étant des variables continues. Le modèle reliant ces deux variables est donné par une fonction de transfert écrite dans le domaine en Z :

$$Y(z) = F(z)U(z).$$

On veut garantir que pour toute variable d'entrée bornée c'est-à-dire telle qu'il existe $M > 0$ tel que :

$$\forall k \geq 0, \quad |u(k)| \leq M,$$

la variable de sortie $y(k)$ est elle-même bornée. On pourrait essayer de simuler tous les scénarios possibles, c'est-à-dire de calculer les sorties $y(k)$ pour toutes les entrées $u(k)$ bornées. Or le nombre d'entrées possibles est infini, ce qui correspond à un nombre infini de scénarios... Une alternative est de calculer les racines du dénominateur de la fonction de transfert $F(z)$. Pour garantir cette propriété, il suffit de vérifier si les racines sont de module strictement inférieur à un. Ainsi, si c'est le cas, sans faire la moindre simulation, on peut garantir que les sorties seront bornées pour toutes entrées bornées. De plus, l'utilisation de la simulation n'aurait permis de vérifier que les sorties sont bornées que pour un nombre limité d'entrées et pour un intervalle de temps lui aussi limité.

Le rôle de la modélisation dans la conception des systèmes est résumé Figure 1.3.

1.2 Quelques classes de systèmes à variables discrètes

Les Réseaux de Petri permettent la modélisation de systèmes à variables d'entrée, d'état et de sortie discrètes. Quels types de systèmes cela recouvre-t-il ? On va simplement présenter succinctement quelques classes de systèmes ; cette présentation n'étant pas exhaustive.

Une première application est la modélisation de **processus de fabrication** de systèmes industriels manufacturiers. Un **processus** est une suite d'opérations effectuées dans un ordre bien

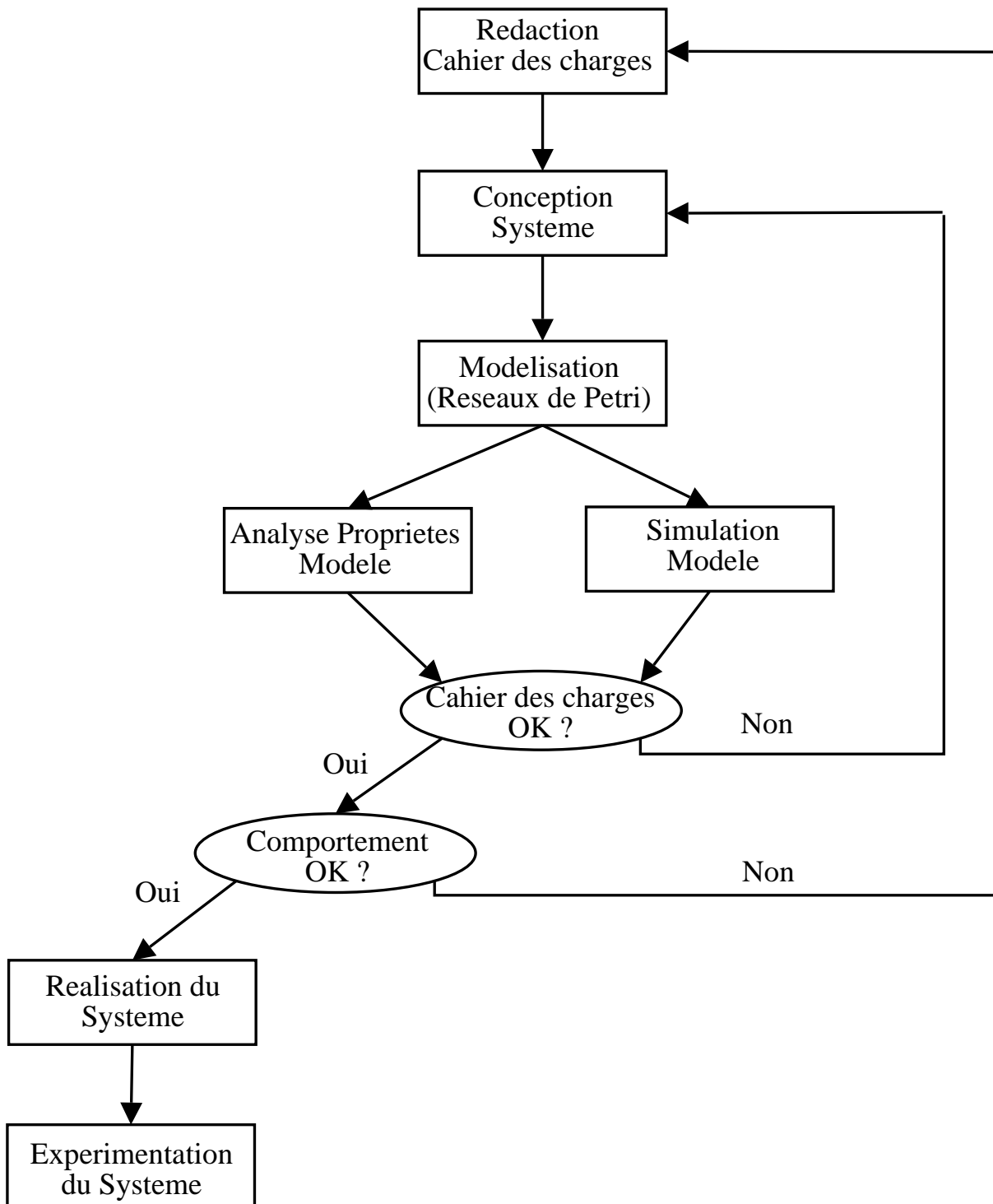


FIG. 1.3 – Rôle de la modélisation dans la conception des systèmes

déterminé. L'industrie manufacturière est caractérisée par la production de pièces en série (par exemple, production de véhicules), soit des variables discrètes représentant des nombres de pièces. On peut l'opposer à l'industrie lourde de transformation de la matière (par exemple, chimie lourde) qui est caractérisée par des flux continus de matière, d'énergie, soit des variables continues. Un processus de fabrication manufacturier utilise des ressources : les machines qui effectuent les différentes opérations, les stocks, les convoyeurs entre les différentes machines, etc... Il apparaît alors des problèmes d'*organisation* et plus particulièrement, des problèmes de *synchronisation* et de *coopération* entre des processus travaillant en *parallèle* et *partageant des ressources*. Ces problèmes ont une acuité de plus en plus forte. D'une part, la réduction des coûts a entraîné la diminution, voire la suppression des stocks (flux tendu). Par exemple, le moteur d'un camion est livré à l'usine d'assemblage à la date (et quasiment à l'heure) de son montage. D'autre part, l'adaptation aux goûts du public (pour un coût limité) a entraîné une grande diversification des pièces produites, voire une individualisation complète (exemple des cabines de camions). Pour garantir un fonctionnement efficace du système, ces problèmes doivent donc être abordés par des méthodes efficaces.

Si les systèmes industriels manufacturiers effectuent des transformations sur des pièces, les systèmes informatiques (par exemple un système d'exploitation) effectuent eux aussi des transformations mais sur de l'information. On peut définir un **processus informatique** de façon similaire à un processus manufacturier. Il utilise des ressources de même type : processeurs pour transformer l'information, mémoires pour la stocker, bus et réseaux pour la transporter, etc ... On est donc confronté aux mêmes problèmes. Il en est de même pour le fonctionnement de réseaux informatiques. Du fait de l'augmentation de la complexité des systèmes informatiques et de leurs interconnexions, ces problèmes sont très difficiles à aborder, sans méthodes particulièrement efficaces.

De façon plus générale, les problèmes d'organisation, de synchronisation et de coopération entre des processus travaillant en parallèle et partageant des ressources apparaissent dans de nombreux systèmes. Il peut s'agir de traiter du trafic dans des réseaux que ce soit des réseaux de communication ou des réseaux routiers. Il peut aussi s'agir d'organiser la diffusion de l'information au sein d'une entreprise (par la mise en place d'un système d'information supporté par un système informatique) ou la gestion d'un projet...

Objectif du cours L'objectif de ce cours n'est pas de présenter la méthode qui résout tous les problèmes succinctement évoqués ci-dessus ! L'étude de la modélisation par Réseaux de Petri permet d'introduire tous les concepts fondamentaux qui apparaissent généralement dans les méthodes traitant ces problèmes. Un étudiant ayant assimilé ce cours aura le bagage scientifique nécessaire pour la compréhension et la mise en œuvre de ces méthodes pour les systèmes particuliers auxquels il sera confronté.

Chapitre 2

Modélisation par Réseaux de Petri

2.1 Modèle de base

2.1.1 Elements de base

Un **graphe** peut être défini par un ensemble d'éléments appelés *noeuds* ou *sommets* et un ensemble de relations appelées *arrêtes* ou *arcs*. Un exemple est donné Figure 2.1 : les noeuds sont constitués par les carrés étiquetés A, B, C et D ; les arcs sont constitués par les droites étiquetées 1 (qui relie le sommet A au sommet B), 2 (qui relie le sommet A au sommet C), 3 (qui relie le sommet B au sommet C) et 4 (qui relie le sommet C au sommet D). Les arcs peuvent être orientés : on parle alors de **graphe orienté**. Les sommets peuvent être de plusieurs types.

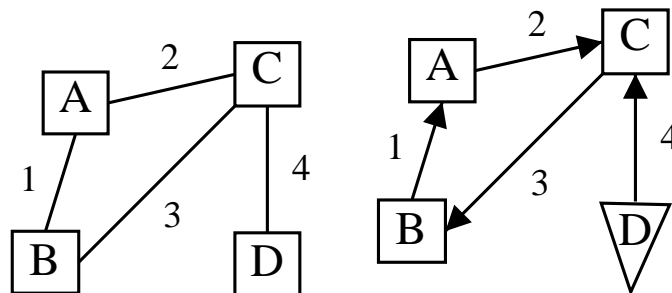
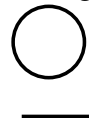



FIG. 2.1 – Exemple de Graphe (gauche) et Graphe orienté (droite)

Un **Réseau de Petri (RdP)** est un graphe orienté comprenant deux types de sommets :

- les **places**
- les **transitions**



Ils sont reliés par des arcs orientés . Un arc relie soit une place à une transition, soit une transition à une place **jamais** une place à une place ou une transition à une transition. Tout arc doit avoir à son extrémité un sommet (place ou transition). Un exemple de Réseaux de Petri est représenté Figure 2.2. A chaque place et transition, un nom peut être associé : par exemple sur le RdP de la Figure 2.2, les places sont nommées P_1, P_2, P_3 et P_4 et les transitions T_1, T_2 et T_3 . T_1 est reliée à P_1 par un arc orienté de T_1 vers P_1 : on dit que P_1 est *en sortie* ou *en aval* de T_1 . P_1 est reliée à T_2 par un arc orienté de P_1 vers T_2 : on dit que T_2 est *en sortie* de P_1 . De même, on peut dire que P_1 est *en entrée* ou *en amont* de T_2 . La place P_3 est *en entrée* de T_1 .

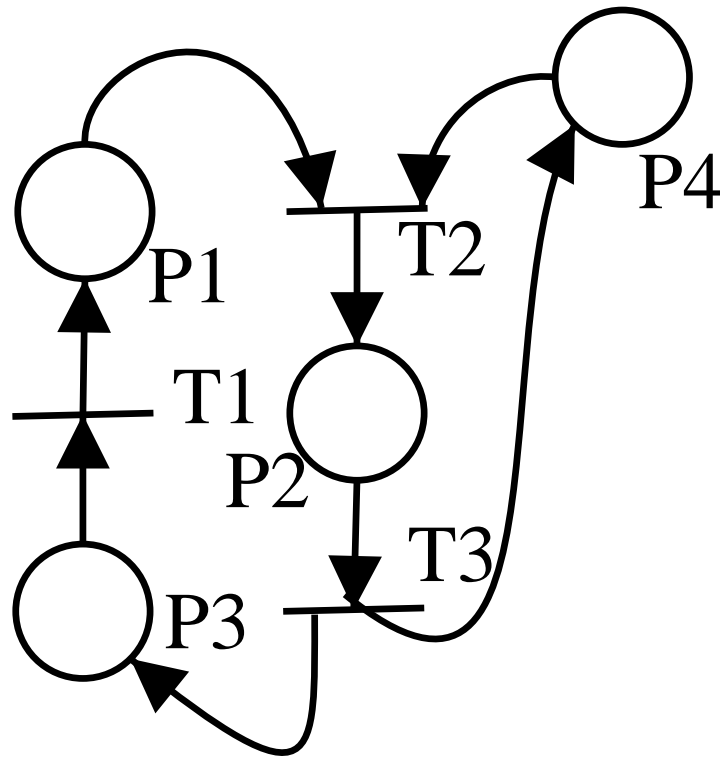
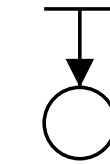


FIG. 2.2 – Exemple d'un Réseau de Petri

Cas particuliers

Transition source pas de place en entrée de la transition



Transition puits pas de place en sortie de la transition



Une place correspond à une *variable d'état* du système qui va être modélisé et une transition à un *événement et/ou une action* qui va entraîner l'évolution des variables d'état du système. A un instant donné, une place contient un certain nombre de **marques** ou **jetons** qui va évoluer en fonction du temps : il indique la valeur de la variable d'état à cet instant. Quand un arc relie une place à une transition, cela indique que la valeur de la variable d'état associée à la place influence l'occurrence de l'événement associé à la transition. Quand un arc relie une transition à une place, cela veut dire que l'occurrence de l'événement associé à la transition influence la valeur de la variable d'état associée à la place.

Exemple de l'atelier de coupe de bois Un atelier est constitué d'une machine de coupe et d'un stock. Quand une commande arrive et que la machine de coupe est disponible, la commande peut être traitée (découpe). Une fois le traitement terminé, la commande qui a été traitée est stockée. Sinon, la commande doit attendre que la machine de coupe se libère avant de pouvoir être traitée.

L'état du système est caractérisé par :

-
-
-
-

L'état du système va évoluer quand se produit :

-
-
-

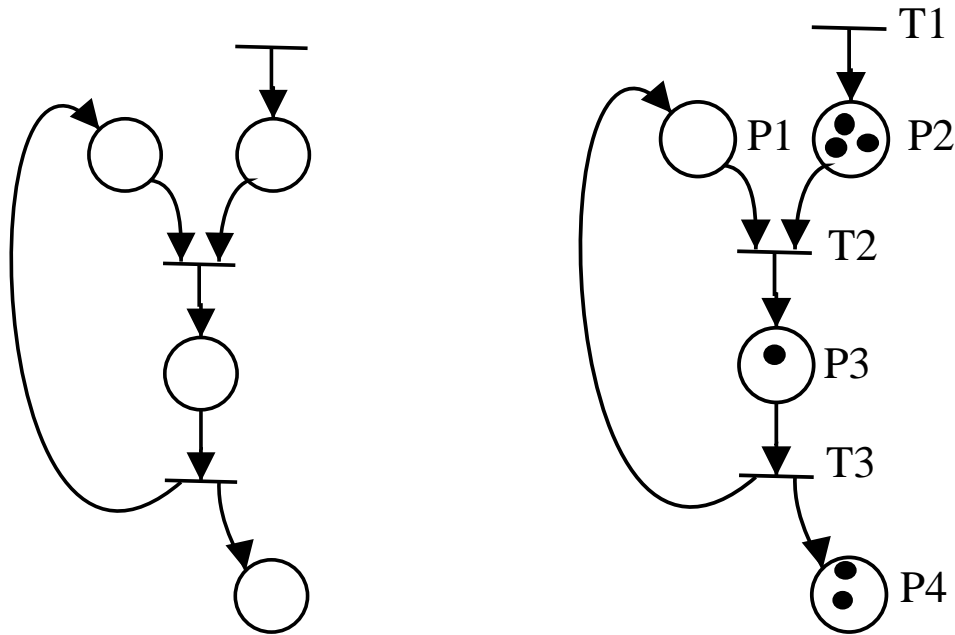


FIG. 2.3 – Atelier de coupe

Il faut noter les choses suivantes.

- Chaque place contient un nombre entier (positif ou nul) de **marques** ou **jetons**. Elle indique la valeur de la variable d'état associée à la place. Par exemple, les 3 marques dans P_2 représentent 3 commandes en attente.
- Le nombre de marques dans une place peut s'interpréter comme le nombre de *ressources* disponibles. Par exemple, dans la place P_1 , une marque indique qu'une machine est disponible, pas de marque indique que la machine n'est pas disponible. Ces ressources sont consommées ou/et produites au cours du temps. Par exemple, dans la place P_2 , le nombre de marques indique le nombre de commandes en attente d'être traitées par la machine de coupe, c'est-à-dire le nombre de "ressources" qui vont être consommées. Une marque dans la place P_3 indique qu'une commande est en train d'être traitée par la machine de coupe. Le nombre de marques dans la place P_4 indique le nombre de commandes qui ont été traitées et stockées. La fin de la coupe par la machine se traduit par la suppression de la marque dans la place P_3 et par la création d'une marque dans la place P_1 et dans la place P_4 .
- Les marques d'une même place n'ont *pas d'identité individuelle* : elles sont indiscernables. Par exemple, chaque commande en attente dans la place P_2 est traitée de la même façon que les autres.

- On appelle **marquage** M d'un Réseau de Petri le vecteur du nombre de marques dans chaque place : la $i^{\text{ème}}$ composante correspond au nombre de marques dans la $i^{\text{ème}}$ place. Il indique à un instant donné l'état du RdP. Par exemple, le marquage du RdP présenté figure 2.3 est donné par :

$$M = \begin{bmatrix} 0 \\ 3 \\ 1 \\ 2 \end{bmatrix}.$$

On appelle **marquage initial**, noté M_0 , le marquage à l'instant initial ($t = 0$).

On peut maintenant donner une définition plus formelle des RdPs.

Definition 2.1.1 On appelle **Réseau de Petri non marqué** le quadruplet $Q = \langle P, T, I, O \rangle$ où

- P est un ensemble fini non vide de places ;
- T est un ensemble fini non vide de transitions ;
- $P \cap T = \emptyset$;
- $I(T_i)$ est l'ensemble des places qui sont en entrée de la transition i ;
- $O(T_i)$ est l'ensemble des places qui sont en sortie de la transition i .

On appelle **Réseau de Petri marqué** $R = \langle Q, M_0 \rangle$ où M_0 est le marquage initial.

Q définit la **structure** du RdP. Le marquage à un instant donné définit la valeur des variables d'**état** du RdP à un instant donné.

Dans l'exemple de l'atelier de coupe, on a :

- $P = \{P_1, P_2, P_3, P_4\}$;
- $T = \{T_1, T_2, T_3\}$;
- $I(T_1) = \{\}$, $I(T_2) = \{P_1, P_2\}$, $I(T_3) = \{P_3\}$;
- $O(T_1) = \{P_2\}$, $O(T_2) = \{P_3\}$, $O(T_3) = \{P_1, P_4\}$.

2.1.2 Evolution d'un RdP

L'**évolution** d'un RdP correspond à l'évolution de son marquage au cours du temps (évolution de l'état du système) : il se traduit par un déplacement de marques ce qui s'interprète comme la consommation/production de ressources déclenchée par des événements ou des actions. Déterminer l'évolution d'un RdP correspond en fait à le *simuler*, terme plus généralement utilisé en modélisation.

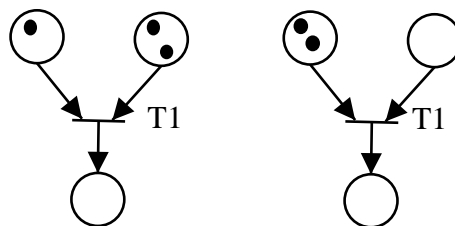


FIG. 2.4 – Transition validée (gauche), non validée (droite)

Transition validée Une transition est dite **validée** si toutes les places en amont (c'est-à-dire en entrée) de celle-ci possèdent au moins une marque. Une transition source est par définition toujours validée.

Deux exemples sont présentés Figure 2.4. Dans l'exemple de l'atelier de coupe, la transition T_3 est validée ainsi que la transition T_1 (transition source). La transition T_2 n'est pas validée.

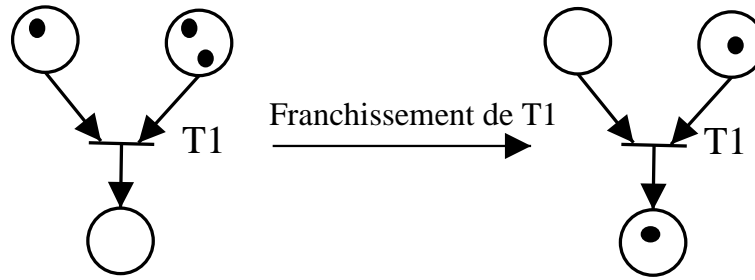


FIG. 2.5 – RdPs avant franchissement de T_1 (gauche) et après franchissement de T_1 (droite)

Franchissement ou tir Si la transition est validée, on peut effectuer le franchissement de cette transition : on dit alors que la transition est **franchissable**. Le franchissement consiste à

- retirer une marque dans chacune des places en entrée de la transition ;
- ajouter une marque à chacune des places en sortie.

Un exemple de franchissement est présenté Figure 2.5. Dans l'exemple de l'atelier de coupe, le franchissement de T_3 mène au marquage suivant :

$$M = \begin{bmatrix} 1 \\ 3 \\ 0 \\ 3 \end{bmatrix}.$$

Remarque Une transition validée est ici forcément franchissable : les deux termes peuvent être donc employés indifféremment. Ce n'est pas le cas pour les extensions des RdPs qui seront introduites chapitre 8.

Lorsqu'une transition est validée, cela n'implique pas qu'elle sera immédiatement franchie. Cela veut dire que les conditions nécessaires à son franchissement sont effectivement réunies : l'action et/ou l'événement associé à la transition sont alors possibles. Dans l'exemple de l'atelier de coupe, pour que la machine de coupe commence la commande (franchissement de la transition T_2), il est nécessaire que la machine soit disponible (une marque dans la place P_1) et qu'au moins une commande soit en attente (au moins une marque dans la place P_2). Cela ne veut pas dire que la machine va commencer à traiter immédiatement la commande.

L'évolution du RdP se fait par *le franchissement d'une seule transition à la fois*. Quand plusieurs transitions sont simultanément franchissables, on ne peut pas savoir dans quel ordre elles seront effectivement franchies. L'évolution n'est donc pas unique.

Dans le cas d'un réseau de Petri "de base" (par exemple pas d'utilisation de temporisation comme dans les RdP temporisés introduits chapitre 8), le franchissement, une fois décidé, est instantané. Dans l'exemple de l'atelier de coupe, les transitions T_2 et T_3 correspondent au début et à la fin du traitement de la commande par la machine de coupe.

Exemple d'évolution Un exemple d'évolution d'un RdP est présenté Figure 2.6.

Notion de RdP autonome et non autonome Un RdP **autonome** décrit le fonctionnement d'un système de façon autonome, c'est-à-dire dont l'évolution est régie par ses propres lois. La succession des quatre saisons peut être ainsi modélisée par un RdP autonome.

L'état du système est caractérisé par :

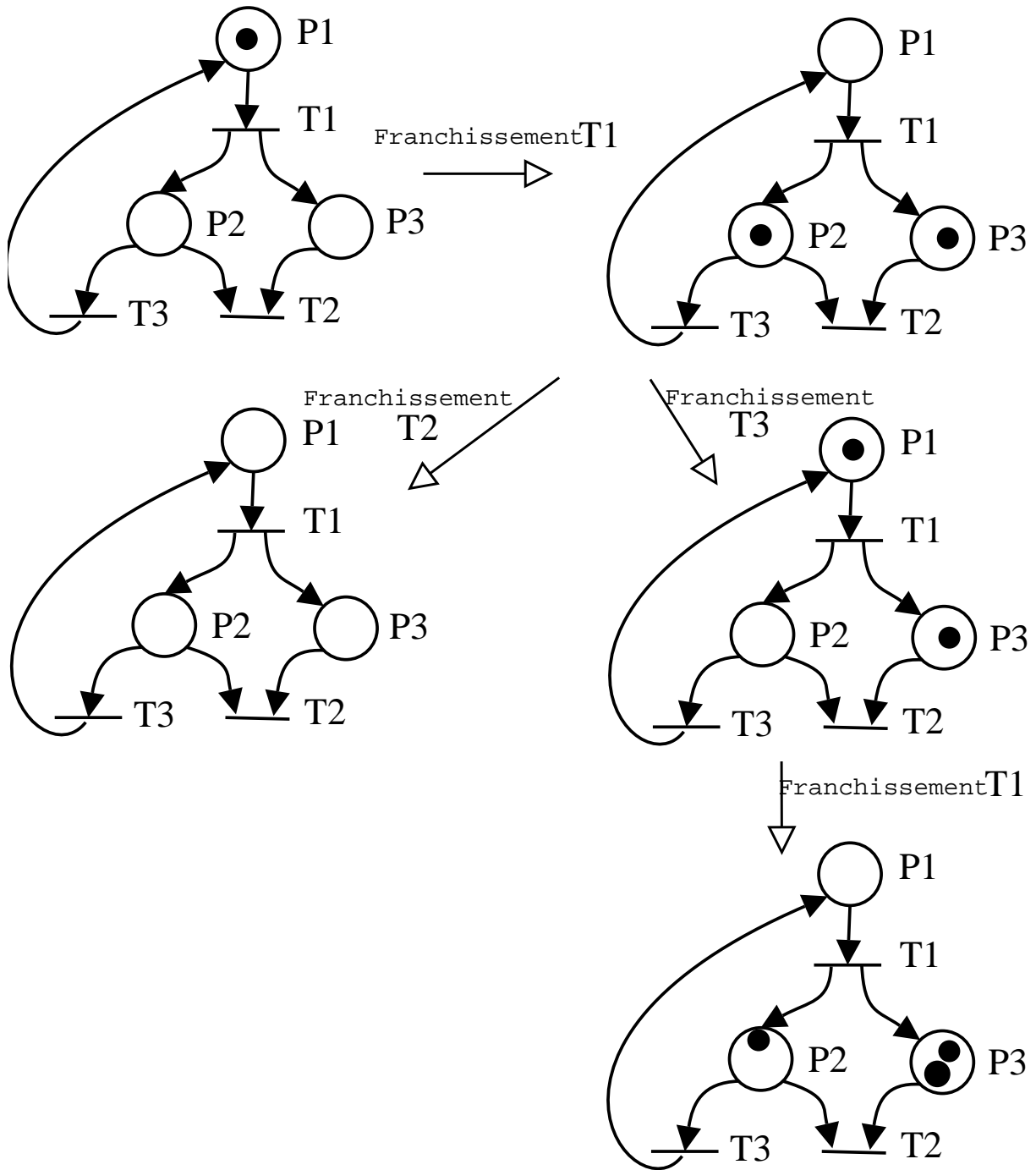


FIG. 2.6 – Exemple d'évolution d'un RdP

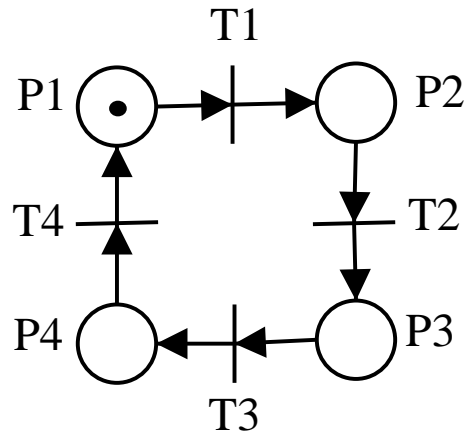


FIG. 2.7 – Quatre saisons

–
–
–
–

L'état du système va évoluer quand se produit :

–
–
–
–

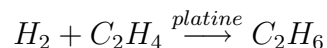
On suppose qu'à l'instant initial $t = 0$, la saison est l'été. On obtient alors le RdP représenté Figure 2.7.

Il s'agit ici de donner une description qualitative du cycle des saisons. Si le franchissement d'une transition est conditionné par un événement, un ordre extérieur ou encore par le temps, on parle alors de RdP non autonome. Le RdP décrivant le fonctionnement de l'atelier de coupe est-il autonome ? Si non, comment le rendre autonome ?

La première partie du cours va être consacrée aux Réseaux de Petri autonomes : les RdPs non autonomes seront abordés dans le chapitre 8.

2.1.3 Exemples de Réseaux de Petri

Modèle d'une réaction chimique On considère maintenant la réaction chimique suivante qui se produit en présence de catalyseur (platine) :



L'état du système est caractérisé par :

- La quantité de H_2 (nombre de marques dans la place P_1)
- La quantité de C_2H_4 (nombre de marques dans la place P_2)
- La quantité de platine (nombre de marques dans la place P_3)
- La quantité de C_2H_6 (nombre de marques dans la place P_4)

L'état du système va évoluer quand :

- La réaction chimique se produit (franchissement de la transition T_1)

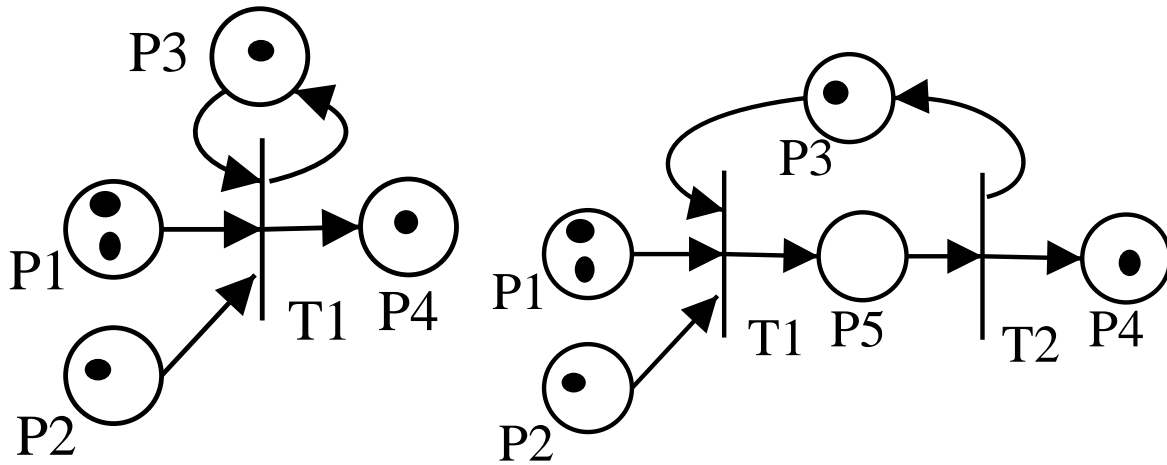


FIG. 2.8 – Réaction chimique

On suppose qu'au départ on a deux unités d' H_2 , une unité de C_2H_4 , une unité de C_2H_6 et que le platine est libre. On obtient alors le RdP représenté Figure 2.8, gauche. Le nombre de marques dans chaque place correspond ici à la quantité de chaque produit chimique. La présence de platine est absolument nécessaire à la réaction chimique : la place P_3 doit être impérativement en entrée de la transition T_1 . Cependant, quand la réaction chimique se produit, le platine étant un catalyseur, il n'est pas consommé. Par suite, la place P_3 doit être en sortie de la transition T_1 de façon à restituer la ressource après que la réaction se soit produite.

La réaction chimique a été supposée être un événement instantané. En réalité, la réaction a une certaine durée caractérisée par un début (où les réactifs se fixent sur le catalyseur afin de réagir) et par une fin (les réactifs ont fini de réagir et libère le catalyseur). On introduit donc une place supplémentaire P_5 qui indique si oui (une marque dans P_5) ou non (pas de marque dans P_5) les produits réagissent à la surface du catalyseur. Une transition supplémentaire T_2 est introduite : le franchissement de T_1 correspond maintenant au début de la réaction chimique et le franchissement de T_2 à la fin de celle-ci. On obtient alors le RdP représenté Figure 2.8, droite.

Modèle d'opérations sur deux entiers naturels Soient n_1 et n_2 deux entiers naturels. On veut modéliser l'*addition* de ses deux entiers par un RdP.

L'état du système est caractérisé par :

- Le nombre n_1 à l'instant initial (nombre de marques dans la place P_1)
- Le nombre n_2 à l'instant initial (nombre de marques dans la place P_2)
- La somme $n_1 + n_2$ à l'instant final (nombre de marques dans la place P_3)

L'état du système va évoluer quand :

- Une marque est enlevée de P_1 pour être placée dans P_3 (franchissement de la transition T_1)
- Une marque est enlevée de P_2 pour être placée dans P_3 (franchissement de la transition T_2)

On obtient alors le RdP représenté Figure 2.9, gauche.

On veut maintenant modéliser la *soustraction* de ces deux entiers par un RdP. La soustraction est définie de la façon suivante : si $n_1 \geq n_2$ alors le résultat est $n_1 - n_2$, sinon le résultat est 0.

L'état du système est caractérisé par :

- Le nombre n_1 à l'instant initial (nombre de marques dans la place P_1)

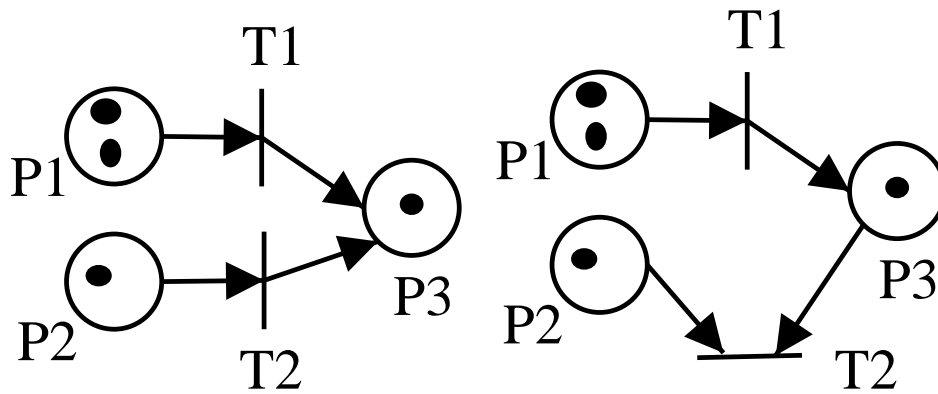


FIG. 2.9 – Opérations sur des entiers naturels, addition (à gauche) et soustraction (à droite)

- Le nombre n_2 à l'instant initial (nombre de marques dans la place P_2)
- La résultat de l'opération à l'instant final (nombre de marques dans la place P_3)

L'état du système va évoluer quand :

- Une marque est enlevée de P_1 pour être placée dans P_3 (franchissement de la transition T_1)
- Une marque est enlevée de P_2 et de P_3 (franchissement de la transition T_2)

On obtient alors le RdP représenté Figure 2.9, droite.

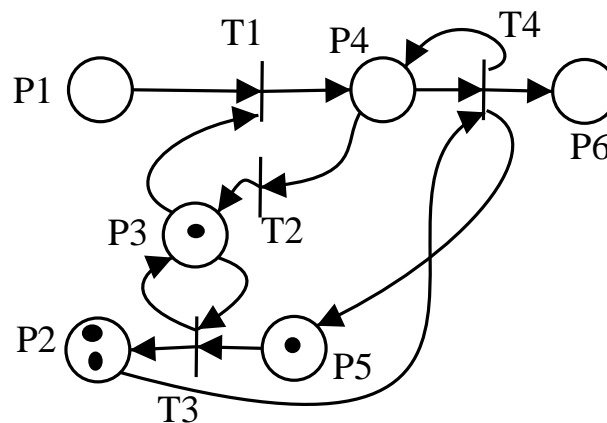


FIG. 2.10 – Opérations sur des entiers naturels : multiplication

On considère maintenant la *multiplication* de ces deux entiers. Le RdP obtenu est représenté Figure 2.10. A l'instant initial, la place P_1 contient n_1 marques et la place P_2 n_2 marques. A l'instant final, la place P_6 contient $n_1 \times n_2$ marques. Vérifier que ce modèle RdP représente bien la multiplication des deux entiers.

2.1.4 Réseaux de Petri avec une structure particulière

La structure d'un Réseau de Petri est définie par le RdP non marqué qui a été noté Q dans la définition 2.1.1, page 12. Les RdPs avec une structure particulière ont des propriétés que n'ont pas forcément les RdPs en général.

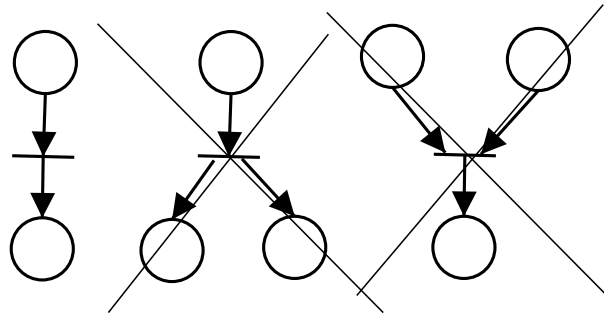


FIG. 2.11 – Graphe d'états ou pas

Graphe d'états Un réseau de Petri est un **graphe d'états** si et seulement si toute transition a *exactement une* place d'entrée et *une* place de sortie (voir Figure 2.11). Si l'ensemble des places du graphe d'états ne contient qu'une seule marque, on retrouve le graphe d'état classique qui a été introduit en licence pour la description et la conception des machines logiques séquentielles.

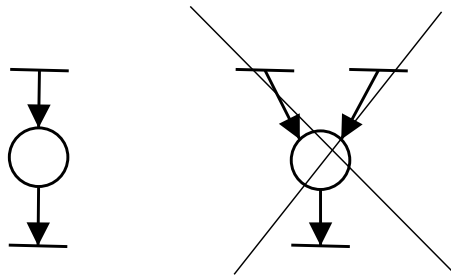


FIG. 2.12 – Graphe d'événements ou pas

Graphe d'événements Un réseau de Petri est un **graphe d'événements** si et seulement si toute place a *exactement une* transition d'entrée et *une* transition de sortie (voir Figure 2.12).

RdP sans conflit Un réseau de Petri est dit **sans conflit** si et seulement si toute place a *au plus une* transition de sortie (voir Figure 2.13). Un **conflit (structurel)** correspond à l'existence d'une place P_i qui a *au moins deux* transitions de sortie T_j, T_k, \dots . Notation $\langle P_i, \{T_j, T_k, \dots\} \rangle$. Sur le RdP de droite de la figure 2.13, on a le conflit $\langle P_1, \{T_2, T_3\} \rangle$. Quand la place P_1 contient une marque, les transitions T_1 et T_2 sont franchissables. Seule une des deux transitions peut être franchie : il est nécessaire de prendre une décision pour savoir laquelle des deux le sera effectivement. L'absence ou la présence d'un conflit est une propriété importante d'un réseau de Petri.

RdP à choix libre Un réseau de Petri est dit **à choix libre** si et seulement si les transitions de sortie de tous ses conflits n'admettent qu'une seule place d'entrée.

RdP à choix libre étendu Un réseau de Petri est dit **à choix libre étendu** si et seulement si pour chaque conflit, toutes les transitions de sortie de celui-ci admettent les mêmes places d'entrée.

Exemple : soit un réseau de Petri à choix libre étendu qui admet le conflit $\langle P_1, \{T_2, T_3\} \rangle$. Si T_2 admet aussi pour place d'entrée P_2 alors forcément T_3 admet P_2 comme place d'entrée.

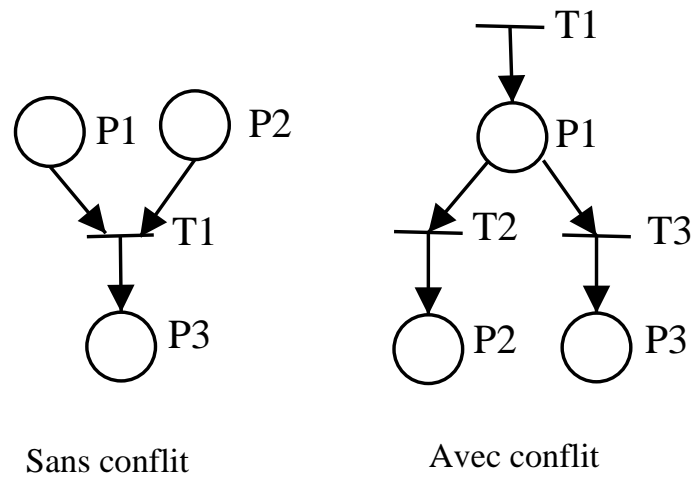


FIG. 2.13 – Conflit ou pas

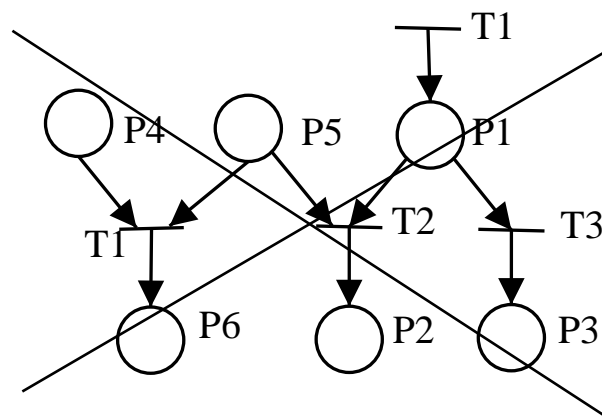


FIG. 2.14 – RdP qui n'est pas simple

RdP simple Un réseau de Petri est dit **simple** si et seulement si toutes ses transitions n'interviennent que dans un seul conflit au maximum. Le RdP représenté Figure 2.13 est simple ; celui représenté Figure 2.14 n'est pas simple.

RdP pur Un réseau de Petri est dit **pur** si et seulement s'il n'existe pas de transition ayant une place d'entrée qui est aussi place de sortie. Le RdP représenté Figure 2.8 n'est pas pur car la place P_3 est place d'entrée et place de sortie de la transition T_1 . On parle alors de RdP **impur**.

Propriété 2.1.1 *Tout RdP impur peut être transformé en un RdP pur.*

Le RdP représentant une réaction chimique Figure 2.8, à gauche n'est pas pur puisque la place P_3 est à la fois place en entrée et en sortie de la transition T_1 . Par contre, une description plus fine de la réaction chimique a donné le RdP Figure 2.8, à droite. Ce RdP est pur. Cependant, il est plus complexe. Cet exemple montre que des "transitions impures" peuvent ainsi être introduites pour simplifier le modèle RdP d'un système. Un autre exemple est donné par les deux RdPs représentés Figure 2.15 qui sont équivalents.

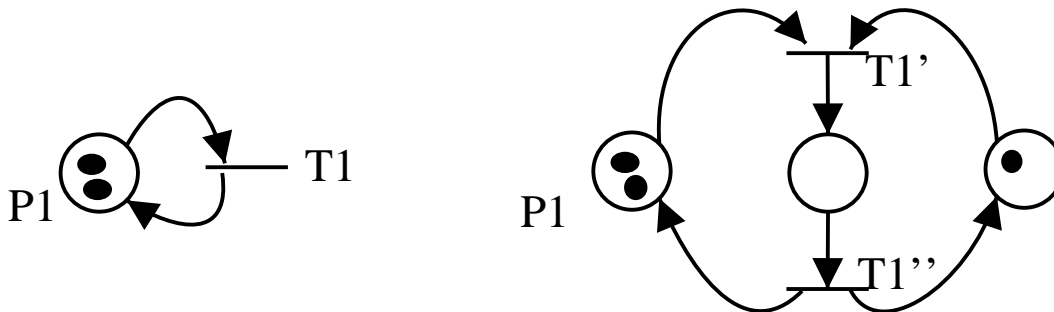


FIG. 2.15 – RdP impur équivalent à un RdP pur

2.2 Structures fondamentales pour la modélisation des systèmes

Les RdPs permettent de modéliser un certain nombre de comportements importants dans les systèmes : le parallélisme, la synchronisation, le partage de ressources, la mémorisation et la lecture d'information, la limitation d'une capacité de stockage. Dans cette section, sont présentées les différentes structures apparaissant dans un réseau de Petri reproduisant ce type de comportements.

2.2.1 Elements de modélisation

2.2.1.1 Parallélisme

Le parallélisme représente la possibilité que plusieurs processus évoluent simultanément au sein du même système. On peut provoquer le départ simultané de l'évolution de deux processus à l'aide d'une transition ayant plusieurs places de sortie. Pour cela, le RdP doit contenir la structure présentée Figure 2.16, gauche. Par convention, lorsqu'un carré grisé apparaît dans une figure, cela indique que seule une partie du RdP a été représentée : pour la compréhension de l'explication, le reste du RdP est supposé ne pas être important. Le franchissement de la transition T_1 met une marque dans la place P_2 (ce qui marque le déclenchement du processus 1) et une marque dans la place P_3 (ce qui marque le déclenchement du processus 2). Il est ensuite

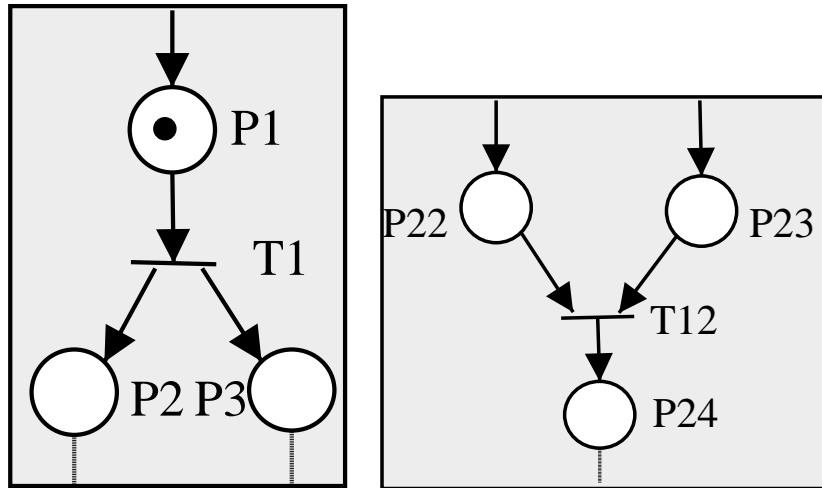


FIG. 2.16 – Parallélisme

possible de synchroniser l'achèvement des deux processus, voir Figure 2.16, droite. La place P_{22} correspond à la fin du processus 1 et la place P_{23} à la fin du processus 2. Le RdP évoluera par franchissement de la transition T_{12} . Pour cela, il est nécessaire que les places P_{22} et P_{23} contiennent chacune au moins un jeton, c'est-à-dire que les processus 1 et 2 soient terminés. Le RdP total est représenté Figure 2.18.

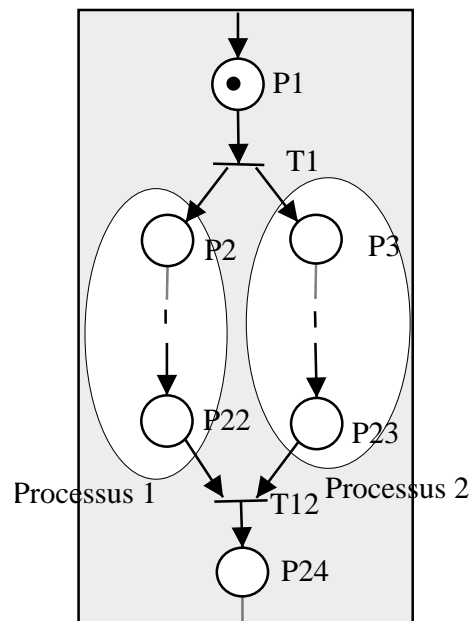


FIG. 2.17 – Parallélisme

La machine à remplir et à boucher La machine à remplir et à boucher des bouteilles est composée de trois postes travaillant en parallèle.

- Le poste 1 sert au transfert et au chargement. Dans on premier temps, on sort le vérin de transfert B pour à décaler le convoyeur d'une position vers la droite. Ensuite, le vérin A sert au chargement d'une nouvelle bouteille vide.
- Le poste 2 sert au remplissage des bouteilles à l'aide de la vanne D.

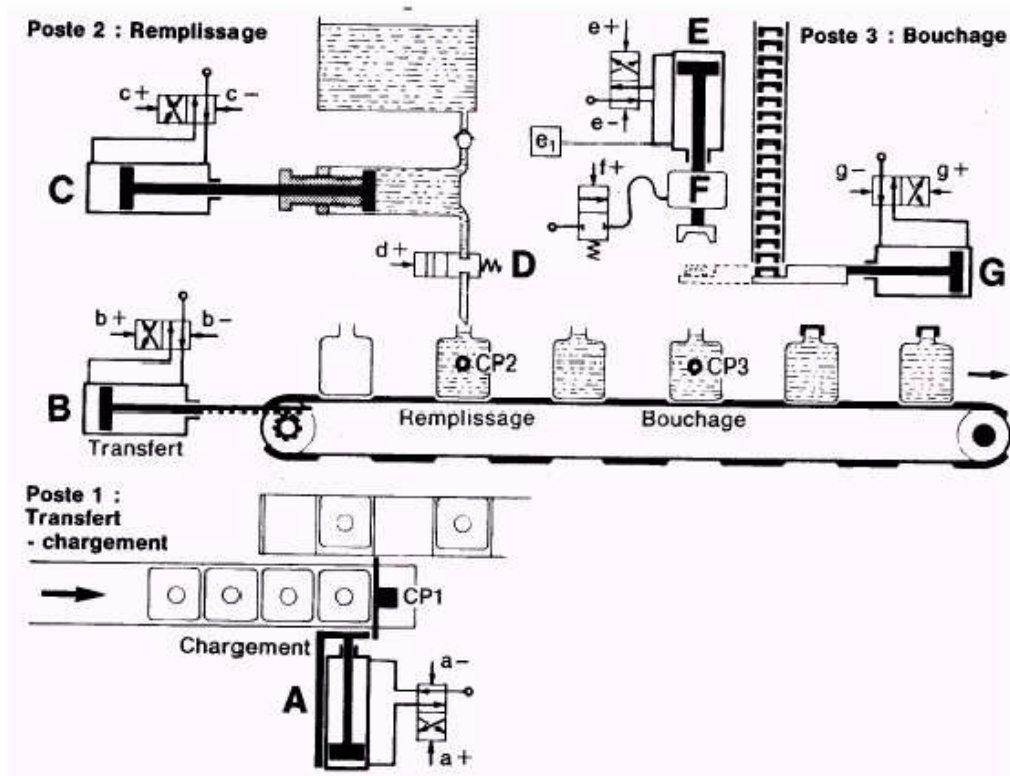


FIG. 2.18 – Parallélisme

– Le poste 3 est le poste de bouchage.

Les actions de chargement d'une bouteille, remplissage d'une bouteille et bouchage d'une bouteille sont effectuées en parallèle. Le transfert par le vérin B n'est effectué que lorsque ces trois opérations sont terminées.

L'état du système est caractérisé par :

-
-
-
-
-
-
-

L'état du système va évoluer quand se produit :

-
-
-
-
-

2.2.1.2 Synchronisation

Mutuelle La *synchronisation mutuelle* ou *rendez-vous* permet de synchroniser les opérations de deux processus. Un exemple est donné Figure 2.20. Le franchissement de la transition T_7 ne peut se faire que si la place P_{12} du processus 1 et la place P_6 du processus 2 contiennent chacun au moins une marque. Si ce n'est pas le cas, par exemple la place P_{12} ne contient pas de marque,

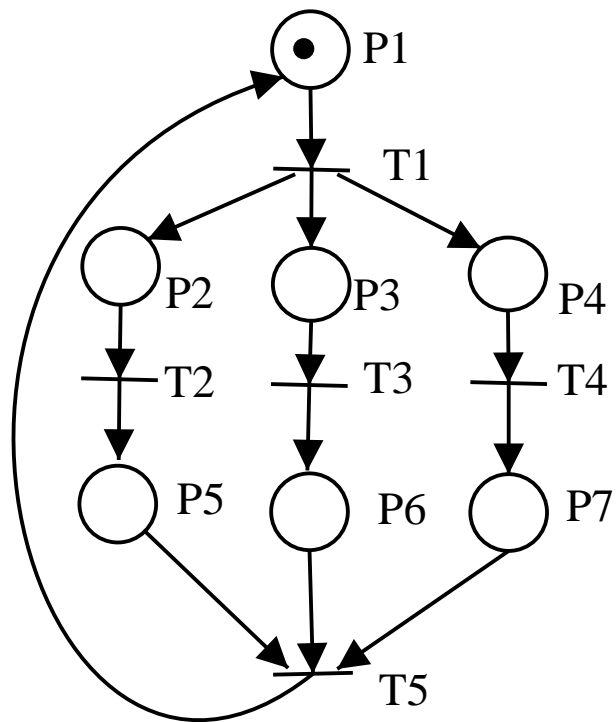


FIG. 2.19 – RdP machine à remplir et à boucher

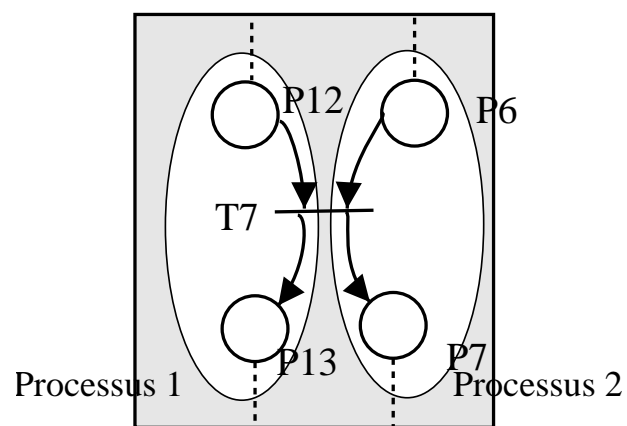


FIG. 2.20 – Synchronisation mutuelle

le processus 2 est “bloqué” sur la place P_6 : il attend que l’évolution du processus 1 soit telle qu’au moins une marque apparaisse dans la place P_{12} .

Exemple On considère un système magasin/consommateur. Le magasin vend un seul type de produits entreposés dans un stock de capacité limité : son état est caractérisé par le nombre de produits en stock et par le nombre d’emplacements libres dans le stock. Le nombre de produits disponibles dans le magasin augmente quand le magasin est livré et diminue quand le consommateur effectue un achat. Le consommateur a deux états : soit il consomme, soit il va faire ses courses.

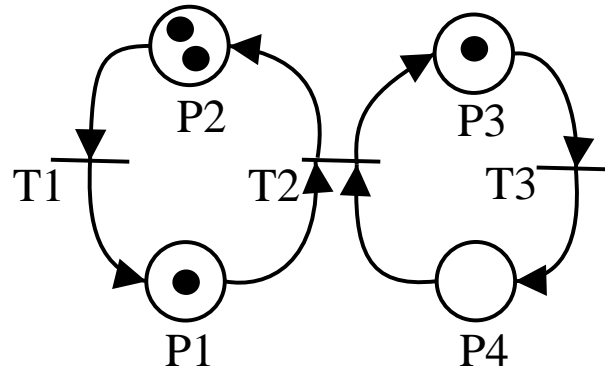


FIG. 2.21 – RdP système magasin/consommateur

L'état du système est caractérisé par :

-
-
-
-

L'état du système va évoluer quand :

-
-
-

On suppose qu'à l'instant initial, le magasin a deux emplacements de libre et un seul produit stocké et que le consommateur est en train de consommer. Le RdP obtenu est représenté Figure 2.21.

Sémaphore Les opérations du processus 2 ne peuvent se poursuivre que si le processus 1 a atteint un certain niveau dans la suite de ses opérations. Par contre, l'avancement des opérations du processus 1 ne dépend pas de l'avancement des opérations du processus 2. D'après Figure 2.22, le processus 2 ne peut franchir la transition T_8 que si la place P_0 contient au moins une marque. Une marque est ajoutée dans la place P_0 lorsque l'évolution du processus 1 amène le franchissement de la transition T_{17} . L'évolution du processus 2 va donc dépendre de l'évolution du processus 1.

Exemple On désire effectuer deux opérations :

$$a \times b + e \quad \text{et} \quad a \times b + c \times d$$

On dispose de deux processeurs pouvant travailler en parallèle. Afin de gagner du temps, on décide que le premier processeur effectuera l'opération $a \times b$ tandis que parallèlement le second

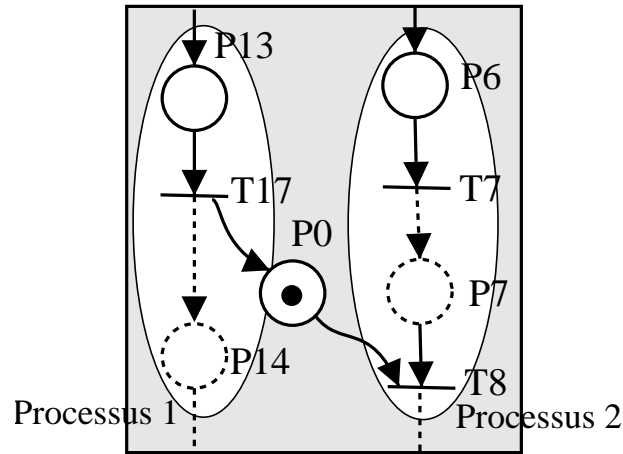


FIG. 2.22 – Sémaphore

effectuera l'opération $c \times d$. Quand le processeur 1 aura terminé le produit, il y ajoutera e . Quand le processeur 2 aura terminé l'opération $c \times d$, si le processeur 1 a terminé le produit $a \times b$, il fera la somme des deux produits, sinon il attendra que le produit $a \times b$ soit terminé pour pouvoir l'effectuer.

L'état du système est caractérisé par :

- le processeur 1 effectue l'opération $a \times b$ (une marque dans la place P_1)
- le processeur 2 effectue l'opération $c \times d$ (une marque dans la place P_2)
- le résultat de $a \times b$ est disponible (une marque dans la place P_3)
- le résultat de $c \times d$ est disponible (une marque dans la place P_4)
- le processeur 1 effectue l'addition $(a \times b) + e$ (une marque dans la place P_5)
- le processeur 2 effectue l'addition $(a \times b) + (c \times d)$ (une marque dans la place P_6)

L'état du système va évoluer quand :

- fin de l'opération $a \times b$ (franchissement de la transition T_1)
- fin de l'opération $c \times d$ (franchissement de la transition T_2)
- début l'addition $(a \times b) + (c \times d)$ (franchissement de la transition T_3)

A l'instant initial, le processeur 1 effectue l'addition $(a \times b) + e$, que le résultat de $a \times b$ est disponible et que le processeur 2 effectue l'opération $c \times d$. Le RdP obtenu est représenté Figure 2.23.

2.2.1.3 Partage de ressources

Cette structure va modéliser le fait qu'au sein du même système plusieurs processus partagent une même ressource. Figure 2.24, la marque dans la place P_0 représente une ressource mise en commun entre le processus 1 et le processus 2. Le franchissement de la transition T_{17} lors de l'évolution du processus 1 entraîne la "consommation" de la marque présente dans la place P_0 . La ressource que constitue cette marque n'est alors plus disponible pour l'évolution du processus 2 puisque le franchissement de la transition T_7 n'est plus possible. Lors de l'évolution du processus 1, lorsque la transition T_{18} est franchie, une marque est alors "redonnée" à la place P_0 : la ressource redevient alors disponible pour l'évolution des deux processus.

Partage d'une mémoire par deux programmes informatiques Deux programmes informatiques, programme 1 et programme 2 partagent un espace mémoire unique. Quand un des deux programmes l'utilise, l'autre ne peut pas y avoir accès.

L'état du système est caractérisé par :

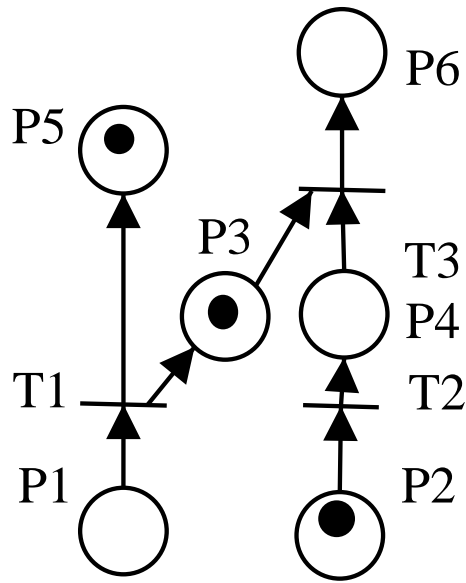


FIG. 2.23 – Calculs parallèles

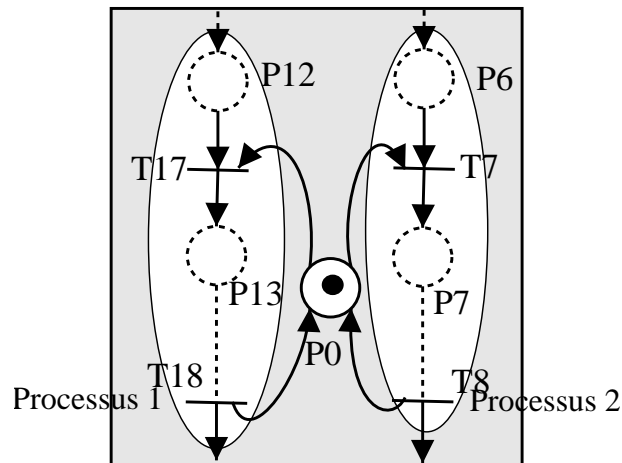


FIG. 2.24 – Partage de ressource

-
-
-
-
-
-
-

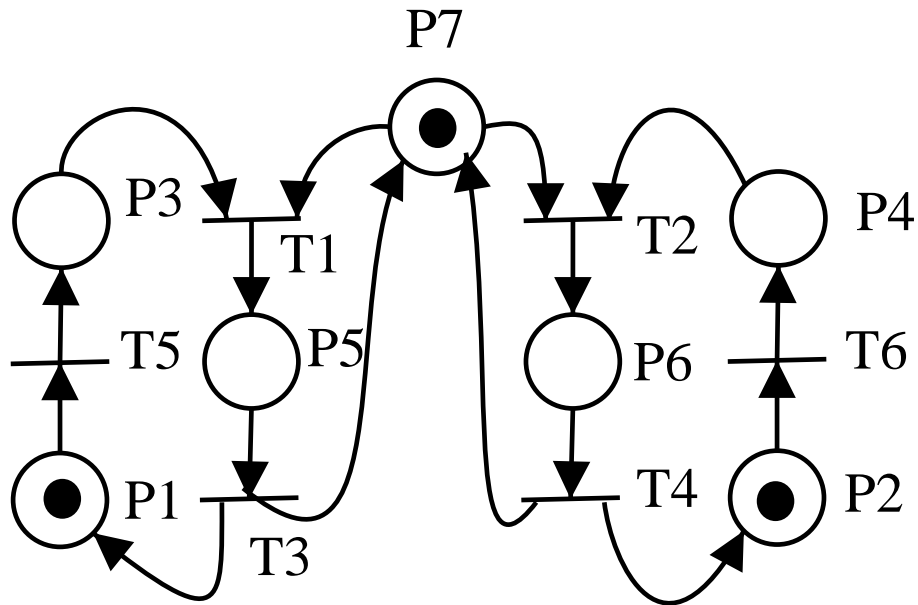


FIG. 2.25 – Partage d’une mémoire par deux programmes

L’état du système va évoluer quand :

-
-
-

A l’instant initial, les deux programmes n’ont pas besoin de la mémoire qui est donc disponible. On obtient alors le RdP représenté Figure 2.25.

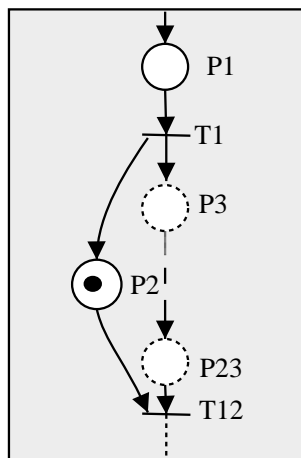


FIG. 2.26 – Mémorisation

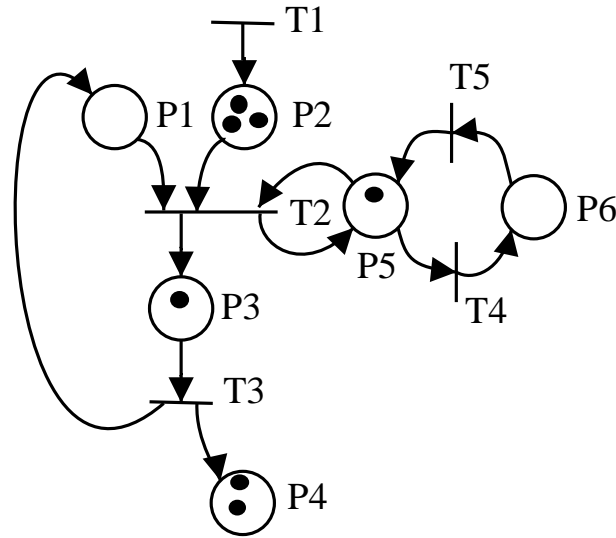


FIG. 2.28 – Atelier de coupe 2

franchissable si le franchissement de la transition T_4 permet de mettre des marques dans la place P_5 . Au total, la place P_4 ne pourra pas contenir plus de 3 marques. Cette partie de RdP peut modéliser un stock de capacité total égale à 3 : d'où le nom **capacité limitée**. Le marquage de P_5 indique le nombre d'emplacements libres dans le stock, le marquage de P_4 le nombre d'emplacements occupés. Le franchissement de la transition T_3 s'interprète alors comme la mise d'un élément dans le stock ; le franchissement de la transition T_4 comme le retrait d'un élément du stock.

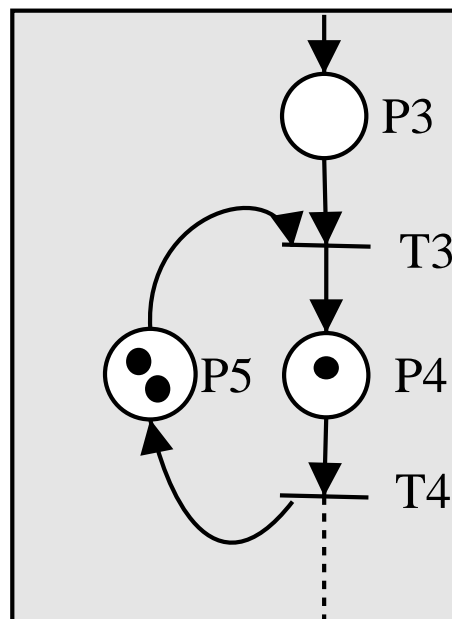


FIG. 2.29 – Capacité limitée

Une capacité limitée a été utilisée dans la modélisation de l'atelier de coupe par le RdP Figure 2.3, page 11. Elle est constituée par l'ensemble transition T_2 , place P_3 , transition T_3 , place P_1 . Par contre, le stock des commandes effectuées est simplement représenté par la place

P_4 : on n'a donc pas modéliser le fait que le stock peut être de capacité limitée. Modifier le RdP de façon à prendre en compte que la capacité du stock est limitée par 4.

2.2.2 Modélisation structurée

Lors de la construction du modèle RdP d'un système d'une grande complexité, il est nécessaire de procéder par étapes. Deux approches sont ainsi possibles.

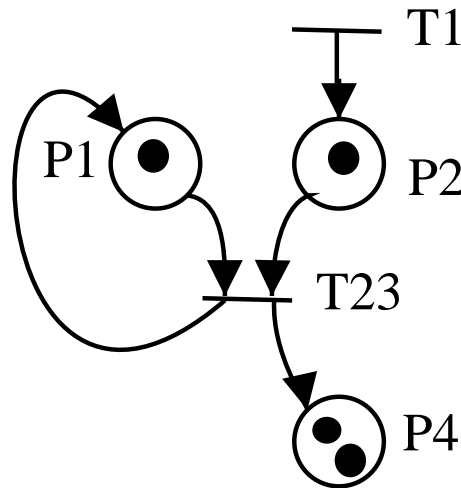


FIG. 2.30 – Atelier suite

2.2.2.1 Approche par affinements successifs

Dans une première étape, le système est décrit par un RdP “simplifié”. Pour cela, les transitions sont associées à des actions/événements complexes. Dans une seconde étape, les transitions sont alors remplacées par les Réseaux de Petri décrivant ces actions/événements complexes.

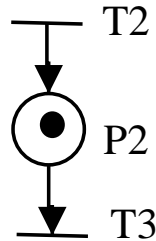
Dans l'exemple de la réaction chimique, section 2.1.3, le premier modèle RdP est un modèle simplifié : la transition T_1 est associée à un événement complexe puisqu'il recouvre la fixation du H_2 et du C_2H_4 sur le catalyseur (platine), la réaction qui se produit sur celui-ci et le départ du C_2H_6 ainsi formé. Le second modèle RdP est un modèle affiné où la transition T_1 a été remplacée par l'ensemble transition T_1 , arc, place P_5 , arc, transition T_2 .

Dans l'exemple de l'atelier, page 10, on aurait pu modéliser de façon plus simple le système en considérant que le début de la découpe par la machine, la découpe elle-même et la fin de la découpe ne constituent qu'un seul et même événement (complexe). L'état du système est alors simplement caractérisé par :

- La machine de coupe est en attente (une marque dans la place P_1)
- Le nombre de commande en attente (nombre de marques dans la place P_2)
- Le nombre de commande traitées et stockées (nombre de marques dans la place P_4)

L'état du système va évoluer quand :

- Arrivé d'une commande (franchissement de la transition T_1)
- La découpe de la commande par la machine de coupe s'est produite (franchissement de la transition T_{23})

FIG. 2.31 – RdP équivalent à T_{23}

Le RdP correspondant est représenté Figure 2.30. La transition T_{23} est associée à un événement complexe : la découpe débute, une commande est dans la machine de découpe, la découpe est terminée. Par suite la transition T_{23} correspond au Réseau de Petri représenté Figure 2.31. En remplaçant T_{23} par celui dans le RdP Figure 2.30, on obtient le RdP Figure 2.3.

2.2.2.2 Approche par compositions de RdPs

Un système complexe peut être décomposé en sous systèmes. Chaque sous système est alors modélisé par un RdP. Le RdP du système complet est alors obtenu en assemblant les RdPs des différents sous systèmes. L'assemblage peut se faire selon deux approches :

1. par fusion des places communes aux différents RdPs : cela traduit par exemple que les sous systèmes partagent des ressources ;
2. par fusion des transitions communes aux différents RdPs : cela traduit par exemple que les sous systèmes évoluent sous l'occurrence de même événements.

Un exemple de la première approche est donné par l'exemple du partage de mémoire précédemment introduit. On modélise l'utilisation d'une mémoire par un programme par un RdP. En écrivant les RdPs pour les deux programmes et en fusionnant les deux places représentant la disponibilité de la mémoire on obtient le RdP de l'utilisation de la mémoire par deux programmes, voir Figure 2.32.

Un exemple de fusion de transitions est donné par le système producteur/magasin/consommateur. Il est constitué de trois sous systèmes qui peuvent chacun être modélisé par un RdP, voir Figure 2.33.

Par fusion des transitions communes, le RdP du système complet est obtenu, voir Figure 2.34.

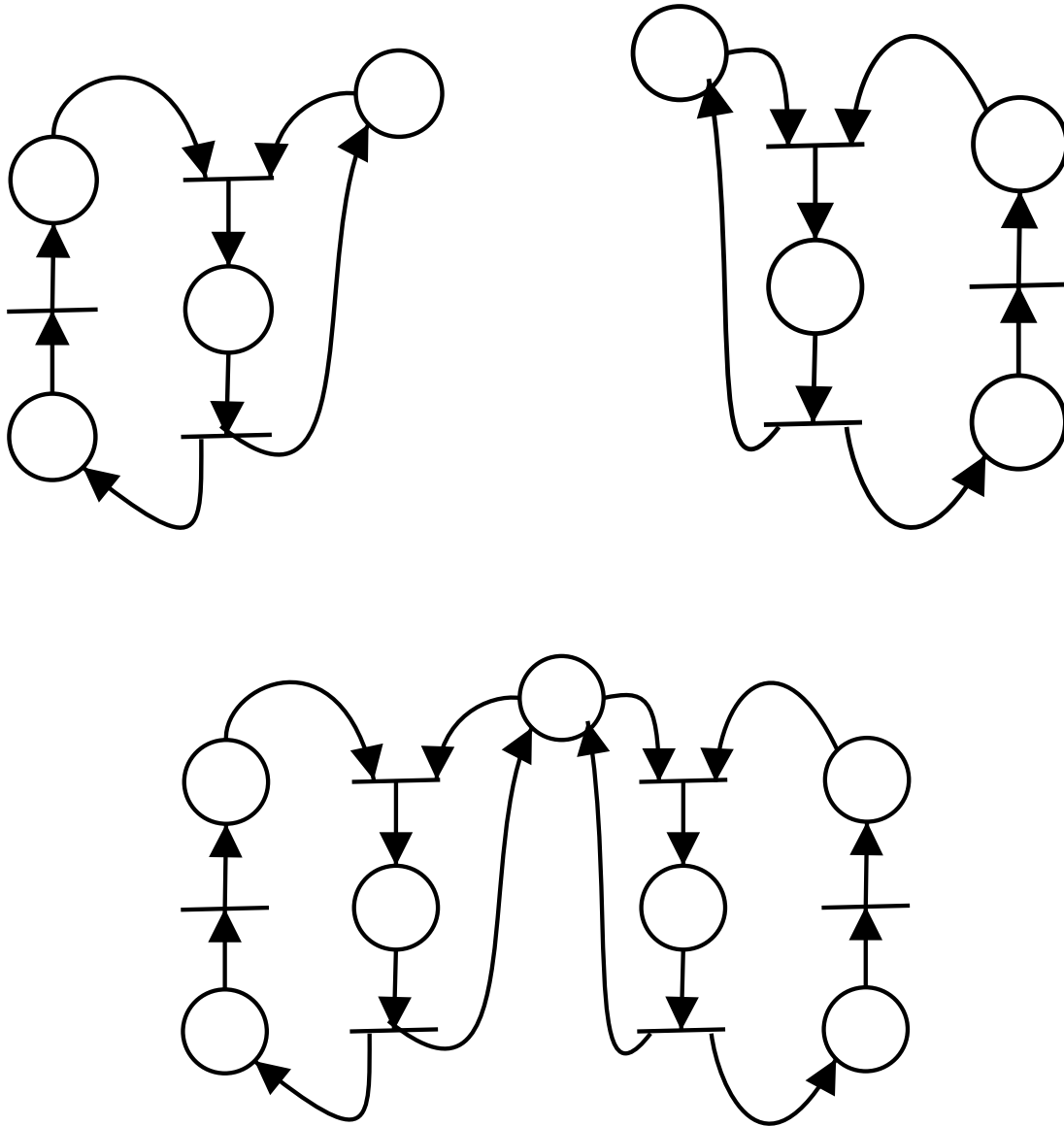


FIG. 2.32 – Exemple de fusion de places

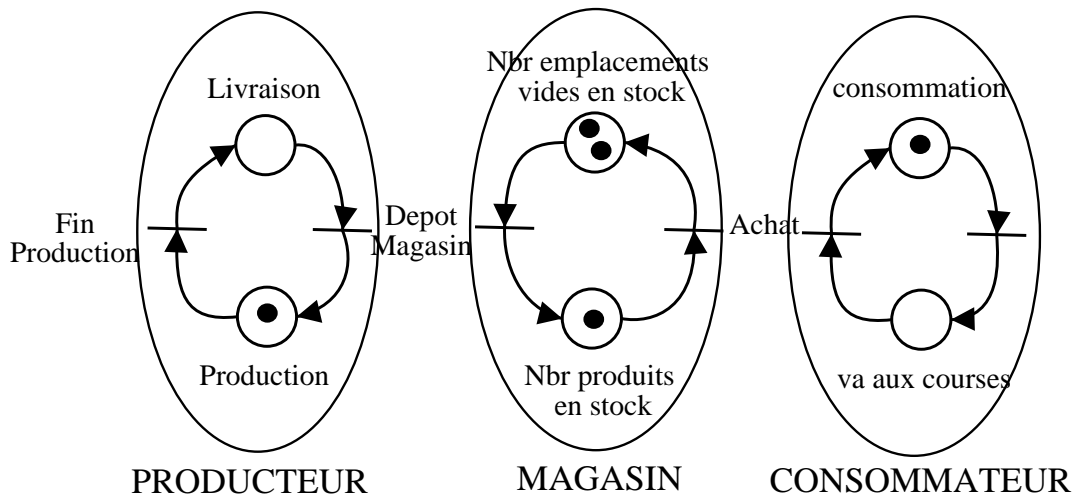


FIG. 2.33 – Producteur magasin consommateur

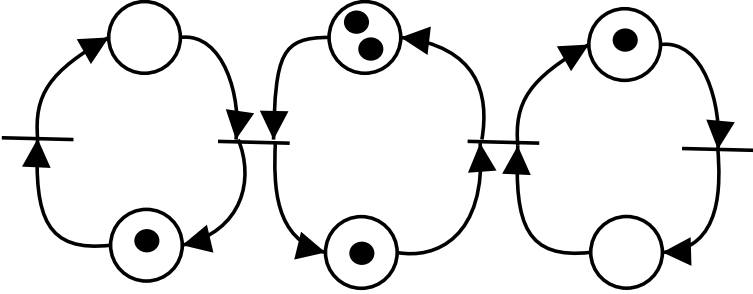


FIG. 2.34 – Producteur magasin consommateur

Chapitre 3

Propriétés des Réseaux de Petri (ordinaires)

3.1 Notations et définitions

Marquage Le marquage d'un Réseau de Petri à un instant donné est un vecteur colonne dont la valeur de la $i^{\text{ème}}$ composante est le nombre de marques dans la place P_i à cet instant.

Le franchissement d'une transition conduit à un changement du marquage. Le passage du marquage M_k au marquage M_l par franchissement de la transition T_j est noté : $M_k | T_j > M_l$. Le nombre de marques dans la place P_i pour le marquage M_k est noté $M_k(P_i)$.

A partir d'un même marquage, il peut être possible de franchir plusieurs transitions, menant ainsi à des marquages différents. L'**ensemble des marquages accessibles** à partir du marquage M_0 est l'ensemble des marquages obtenus à partir de M_0 par franchissements successifs d'une ou plusieurs transition(s). Cet ensemble est noté $*M_0$.

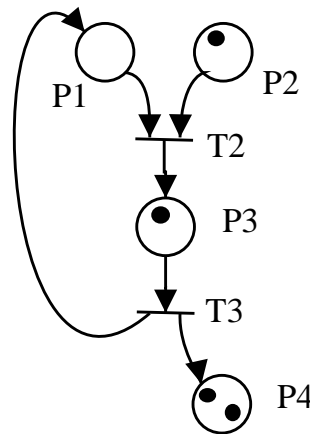


FIG. 3.1 – Atelier de coupe simplifié

Le RdP représenté Figure 3.1 représente une version simplifiée de l'atelier de coupe.

L'ensemble des marquages accessibles est donc défini par : $*M_0 = \{M_1, M_2, M_3\}$.

franchissement de la transition T_1 ou de la transition T_2	$T_1 + T_2 = T_2 + T_1$
franchissement de la transition T_1 puis de la transition T_2	$T_1T_2 \neq T_2T_1$
franchissement de la transition T_1 puis de la transition T_1	$T_1^2 = T_1T_1$
séquence de longueur 2	T_1T_2
séquence de longueur nulle	λ
répétition du franchissement de T_1 un nombre quelconque de fois	$T_1^* = (\lambda + T_1 + T_1^2 + \dots)$

TAB. 3.1 – Notation des séquences de franchissement

Séquence de franchissements

Definition 3.1.1 *Une séquence de franchissement est une suite de transitions qui sont successivement franchies pour passer d'un marquage à un autre.*

Dans l'exemple de l'atelier précédent :

$$M_0 | S > M_3 \quad \text{avec} \quad S = T_3T_2T_3.$$

Les séquences de franchissement sont décrites à l'aide des notations présentées dans le tableau 3.1. Par exemple, le franchissement de la transition T_1 suivi du franchissement de la transition T_2 ou du franchissement de la transition T_3 se note :

$$T_1(T_2 + T_3) = T_1T_2 + T_1T_3.$$

Le franchissement de la transition T_1 suivi du franchissement de la transition T_3 ou le franchissement de la transition T_2 suivi du franchissement de la transition T_3 se note :

$$(T_1 + T_2)T_3 = T_1T_3 + T_2T_3.$$

L'ensemble des séquences de franchissement des 4 saisons (RdP représenté Figure 2.7) se note :

Couverture

Definition 3.1.2 *Un marquage M_k couvre un marquage M_l si, pour chaque place, le nombre de marques de M_k est supérieur ou égal au nombre de marques de M_l :*

$$\forall i, M_k(P_i) \geq M_l(P_i).$$

Notation : $M_k \geq M_l$. La **couverture** est **stricte** si de plus :

$$\exists m, M_k(P_m) > M_l(P_m).$$

Notation : $M_k > M_l$.

Propriété 3.1.1 *Pour un Réseau de Petri non marqué Q , soit $L(Q, M_0)$ l'ensemble des séquences de franchissement à partir du marquage initial M_0 . Si $M'_0 \geq M_0$ alors $L(Q, M_0) \subset L(Q, M'_0)$.*

3.2 Propriétés

3.2.1 Réseau de Petri borné et Réseau sauf

Definition 3.2.1 Une place P_i est *bornée pour un marquage initial* M_0 si pour tout marquage accessible à partir de M_0 , le nombre de marques dans P_i reste borné. Elle est dite *k-bornée* si le nombre de marques dans P_i est toujours inférieur ou égal à k . Un RdP marqué est (*k*) borné si toutes ses places sont (*k*) bornées.

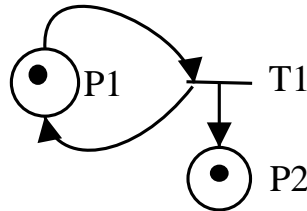


FIG. 3.2 – RdP non borné

Un RdP marqué peut ne pas être borné : sur l'exemple représenté figure 3.2, la transition T_1 admet la place P_1 comme unique place d'entrée. La place P_1 a une marque : la transition T_1 est franchissable. Comme P_1 est aussi place de sortie de T_1 , le franchissement de T_1 ne change pas le marquage de P_1 . La transition T_1 est donc franchissable en permanence et peut donc être franchie un nombre de fois infini. Chaque franchissement de T_1 ajoutant une marque dans la place P_2 , le marquage de celle-ci peut donc tendre vers l'infini.

Definition 3.2.2 Un RdP marqué est *sauf* ou *binnaire* pour un marquage initial M_0 s'il est 1-borné.

Dans le cas où un RdP marqué modélise un système logique, chaque place du RdP correspond à un élément du vecteur d'état du système. Celui-ci ne peut prendre comme valeur que 0 ou 1, soit zéro ou une marque dans la place correspondante. Tester si le système logique est cohérent revient alors à vérifier que son modèle RdP est sauf.

struct ?

Remarque Il faut bien noter que les propriétés précédentes dépendent du marquage initial. Par exemple, pour le RdP représenté Figure 3.2, avec ${}^*M_0 = \begin{bmatrix} 0 \\ N \end{bmatrix}$, le RdP marqué serait borné.

Propriété 3.2.1 Si un RdP marqué n'est pas borné pour le marquage initial M_0 alors il n'est pas borné pour le marquage initial $M'_0 \geq M_0$.

3.2.2 Vivacité et blocage

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

Definition 3.2.3 Une *transition* T_j est *vivante pour un marquage initial* M_0 si pour tout marquage accessible M_k , il existe une séquence de franchissements à partir de M_k contenant T_j :

$$\forall M_k \in {}^*M_0, \exists S, M_k | S > \text{ et } S = \dots T_j \dots$$

Si une transition T_j est vivante alors, à tout instant, on sait que T_j peut être franchie dans le futur. Dans le cas d'un Rdp modélisant un système fonctionnant en permanence, si une transition n'est pas vivante et si une fonction du système est associée au franchissement de cette transition, cela veut dire qu'à partir d'un certain instant, cette fonction ne sera plus disponible dans le futur, ce qui peut traduire une erreur ou une panne.

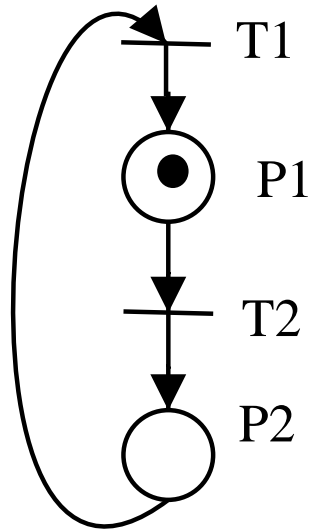


FIG. 3.3 – Rdp vivant

Exemples Les transitions T_2 et T_3 du Rdp marqué Figure 3.1 ne sont pas vivantes. Les transitions T_1 et T_2 du Rdp marqué Figure 3.3 sont vivantes : pour ce Rdp, on a :

Definition 3.2.4 Un *Rdp marqué est vivant pour un marquage initial* M_0 si toutes ses transitions sont vivantes pour ce marquage initial. Un Rdp est dit *conforme* s'il est sauf et vivant.

Le Rdp marqué Figure 3.3 est vivant et conforme.

Definition 3.2.5 Une *transition* T_j est *quasi vivante pour un marquage initial* M_0 s'il existe une séquence de franchissements à partir de M_0 contenant T_j :

$$\exists S, M_0 | S > \text{ et } S = \dots T_j \dots$$

Un *Rdp marqué est quasi vivant pour un marquage initial* M_0 si toutes ses transitions sont quasi vivantes pour ce marquage initial.

Le Rdp marqué Figure 3.1 est quasi vivant.

Définition 3.2.6 Un **blocage** (ou **état puits**) est un marquage pour lequel aucune transition est validée. Un RdP marqué est dit **sans blocage pour un marquage initial** M_0 si aucun marquage accessible est un blocage.

Le RdP marqué Figure 3.1 a pour blocage le marquage :

$$M_3 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 4 \end{bmatrix}.$$

Remarque Il faut bien noter que, pour la bornitude, les propriétés précédentes dépendent du marquage initial.

Propriété 3.2.2

1. Si la transition T_j est quasi vivante pour un marquage initial M_0 alors elle est quasi vivante pour $M'_0 \geq M_0$.
2. Si la transition T_j est vivante pour un marquage initial M_0 alors elle n'est pas nécessairement vivante pour $M'_0 \geq M_0$.
3. Si un RdP marqué est sans blocage pour un marquage initial M_0 alors il n'est pas nécessairement sans blocage pour $M'_0 \geq M_0$.

Démonstration

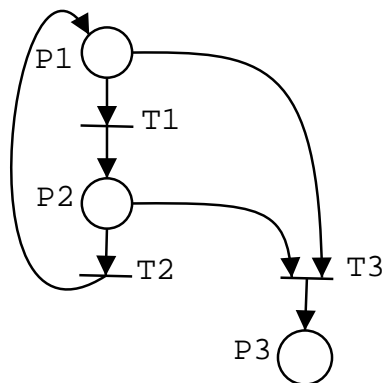


FIG. 3.4 – Couverture et vivacité

1. Elle découle du fait que toute séquence de franchissement à partir de M_0 est aussi séquence de franchissement à partir de M'_0 .

2. et 3. Pour $M_0 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$ et le RdP représenté Figure 3.4 : les transitions T_1 et T_2 sont

vivantes et le RdP marqué est sans blocage. Pour $M'_0 = \begin{bmatrix} 2 \\ 0 \\ 0 \end{bmatrix} \mid T_1 T_3 > M_1$: la

transition T_2 n'est plus vivante et le marquage $M_1 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$ est un blocage.

□

Definition 3.2.7 Un RdP marqué a un **état d'accueil** M_a pour un marquage initial M_0 si pour tout marquage accessible M_k à partir de M_0 , il existe une séquence de franchissements permettant d'atteindre le marquage M_a :

$$\forall M_k \in {}^*M_0, \exists S_j, M_k | S_j > M_a.$$

Si un RdP présente un état d'accueil, il est aisé de vérifier s'il est sans blocage et d'étudier sa vivacité.

Propriété 3.2.3 Soit un Réseau de Petri $\langle Q, M_0 \rangle$ présentant un état d'accueil M_a .

1. Il est sans blocage si et seulement si il existe une transition franchissable à partir du marquage d'accueil M_a .
2. Une transition est vivante pour $\langle Q, M_0 \rangle$ si et seulement si elle est quasi vivante pour $\langle Q, M_a \rangle$.

Definition 3.2.8 Un RdP est réinitialisable pour un marquage initial M_0 si M_0 est un état d'accueil.

Exemple Les 4 saisons.

3.2.3 Conflits

Un conflit structurel a précédemment été défini comme l'existence d'une place P_i qui a au moins deux transitions de sortie T_j, T_k , etc.. Notation : $\langle P_i, \{T_j, T_k, \dots\} \rangle$.

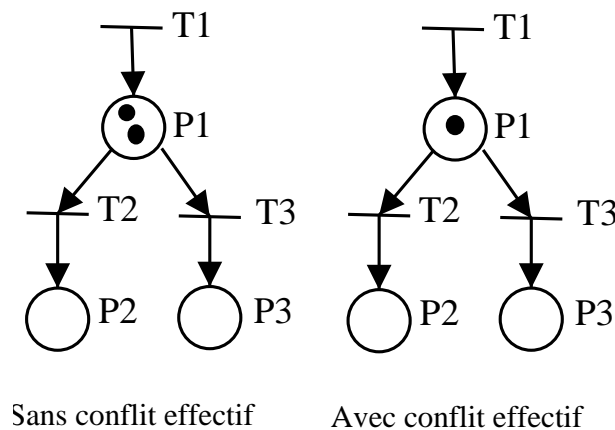


FIG. 3.5 – Conflit effectif ou pas

Definition 3.2.9 Un **conflit effectif** est l'existence d'un conflit structurel $\langle P_i, \{T_j, T_k, \dots\} \rangle$ et d'un marquage M tels que le nombre de marques dans la place P_i est strictement inférieur au nombre des transitions de sortie de P_i validées par le marquage M (voir Figure 3.5).

Lors qu'un conflit effectif se produit, il est nécessaire de choisir la transition qui va être effectivement franchie.

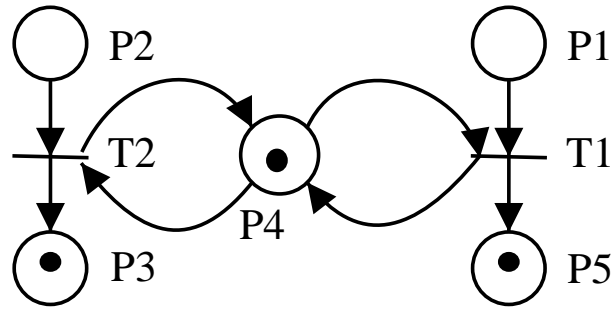


FIG. 3.6 – RdP persistant

Definition 3.2.10 *Un RdP est **persistant** pour un marquage initial M_0 si pour tout marquage M_i accessible à partir de M_0 , on a : si T_j et T_k sont validées par le marquage M_i alors $T_j T_k$ ($T_k T_j$) est une séquence de franchissement à partir de M_i .*

Si le RdP est persistant alors il n'est pas nécessaire d'effectuer un choix lors d'un conflit effectif. Un exemple est donné Figure 3.6.

3.3 Invariants

A partir d'un marquage initial, le marquage d'un RdP évolue par franchissements de transitions. En l'absence de blocage, le nombre de franchissements de transitions et le nombre de marquages effectivement réalisées sont illimitées. Il est donc difficile d'étudier les séquences de transition et les marquages accessibles simplement par exemple en entreprenant une énumération. On est donc amené à définir des invariants caractérisant certaines propriétés des séquences de transitions et des marquages accessibles quelque soit l'évolution.

3.3.1 Composante conservative

Lorsqu'on considère l'exemple des 4 saisons, modélisé par le RdP représenté Figure 2.7, on peut noter que la somme du nombre de marques présentes dans le RdP à un instant donné est toujours égal à un (conservation du nombre de marques) :

$$M(P_1) + M(P_2) + M(P_3) + M(P_4) = 1.$$

Cette égalité exprime qu'on ne peut avoir qu'une saison à la fois. D'autre part, elle implique que $\forall i \in \{1, 2, 3, 4\}, M(P_i) \leq 1$. Le RdP est donc sauf.

Dans l'exemple de l'atelier de coupe simplifié présenté Figure 3.1, on a :

$$M(P_2) + M(P_3) + M(P_4) = 4 \qquad M(P_1) + M(P_3) = 1$$

$M(P_2)$ est le nombre de commandes en attente, $M(P_3)$ le nombre de commande traitée par la machine de coupe et $M(P_4)$ celui qui ont été traitées et stockées. Le premier invariant indique donc que le nombre total de commandes dans l'atelier quelque soit sa forme reste constant. $M(P_1)$ vaut 1 si la machine de coupe est disponible et $M(P_3)$ vaut 1 si la machine de coupe traite une commande. Cet invariant indique donc que soit la machine de coupe est disponible, soit elle traite une commande. On peut interpréter cela comme une capacité limitée au niveau de la place P_3 : elle ne peut contenir qu'une marque au maximum.

Definition 3.3.1 Soit R un Réseau de Petri et P l'ensemble de ses places. On a un **invariant de marquage** s'il existe un ensemble de places $P' \subset P$ et un vecteur d'entiers naturels appelé vecteur de pondérations q tels que :

$$\forall M \in {}^*M_0, \quad \sum_{P_i \in P'} q_i M(P_i) = \text{constante}$$

P' est appelé **composante conservative**. Un RdP est **conservatif** si et seulement si $P' = P$.

Le RdP 4 saisons est conservatif.

Remarque La propriété "composante conservative" est indépendante du marquage initial M_0 . Par contre, la valeur de la constante dépend de M_0 .

3.3.2 Composante répétitive

Il s'agit ici d'étudier le comportement cyclique de l'évolution de certains RdPs. Dans l'exemple des 4 saisons, les séquences franchissables à partir du marquage initial

$$M_0 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

sont : $T_1, T_1T_2, T_1T_2T_3, T_1T_2T_3T_4$, etc.. $T_1T_2T_3T_4$ est une séquence qui partant de M_0 ramène à l'état initial :

$$M_0 | T_1T_2T_3T_4 > M_0$$

Elle pourra donc être répétée indéfiniment.

Definition 3.3.2 On appelle **séquence répétitive stationnaire**, une séquence de franchissements S telle que

$$M_0 | S > M_0.$$

La séquence est dite **complète** si elle contient toutes les transitions du RdP. On appelle **séquence répétitive croissante**, une séquence de franchissements S telle que

$$M_0 | S > M'_0 \quad \text{avec} \quad M'_0 > M_0.$$

On appelle **séquence répétitive décroissante**, une séquence de franchissements S telle que

$$M_0 | S > M''_0 \quad \text{avec} \quad M_0 > M''_0.$$

On appelle **composante répétitive** l'ensemble T' des transitions de T apparaissant dans la séquence S . Le RdP est dit **répétitif** si $T = T'$.

Propriété 3.3.1 Si S est une séquence répétitive pour la condition initiale M_0 alors c'est aussi une séquence répétitive pour la marquage initial $M'_0 \geq M_0$.

Après avoir défini les propriétés des RdPs qu'il est intéressant d'analyser, se pose naturellement le problème de les analyser pratiquement. Trois grandes approches ont été développées. La première est basée sur l'utilisation des graphes et arborescences de marquage (chapitre 4), la seconde fait appel à l'algèbre linéaire (chapitre 5) et enfin la troisième consiste à réduire le RdP étudié en un RdP plus simple conservant les propriétés analysées du RdP initial (chapitre 6).

Chapitre 4

Etude des propriétés des Réseaux de Petri par graphes et arborescences de marquages

4.1 Introduction

L'idée la plus naturelle pour étudier les propriétés d'un RdP est de construire le graphe de tous les marquages accessibles. Le graphe des marquages accessibles est un graphe dont chaque sommet correspond à un marquage accessible et dont chaque arc correspond au franchissement d'une transition permettant de passer d'un marquage à l'autre.

Dans l'exemple de l'atelier de coupe simplifié présenté Figure 3.1,

Le graphe est, dans cet exemple, de dimension finie. Par inspection directe, on peut déterminer le RdP marqué : 4-borné, pas vivant, quasi vivant, avec un blocage et un état d'accueil, pas réinitialisable. Il n'y a pas de séquence répétitive stationnaire. On a les invariants de marquage suivant :

Suivant les exemples considérés, le graphe obtenu peut être de dimension infinie. Considérons le RdP représenté Figure 4.1.

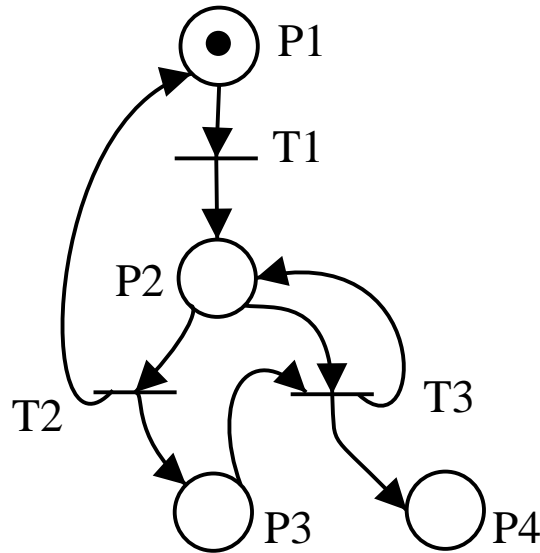


FIG. 4.1 – Exemple 1 Graphes de marquage

La séquence répétitive croissante T_2T_1 mène à un marquage du type :

$$\begin{bmatrix} 0 \\ 1 \\ k \\ 0 \end{bmatrix}$$

où k est un entier qui peut être arbitrairement grand. Le graphe de marquage est donc de dimension infinie. Pour traiter de ce cas de figure, on introduit la notation ω . ω représente un entier arbitrairement grand¹. Il a les propriétés suivantes : avec n un entier naturel

- $\omega + n = \omega + \omega = \omega$
- $\omega - n = \omega$
- $n < \omega$

- On a ainsi obtenu le **graphe de couverture** des marquages accessibles. (Pourquoi le terme couverture ?). Bien qu'associé à un RdP dont le nombre de marquages accessibles n'est pas borné, ce graphe a un nombre fini de sommets.
- A partir du graphe de couverture, on peut conclure sur la bornitude et la quasi vivacité du RdP. Dans cet exemple, P_1 et P_2 sont 1 bornés. P_3 et P_4 ne sont pas bornés. Le RdP est quasi vivant. Il n'est pas possible de conclure sur les autres propriétés car l'introduction de ω correspond à une perte d'information. On ne sait pas quelles valeurs prendra exactement la composante du marquage qui le contient.
- A un graphe de couverture, il est possible d'associer une arborescence de couverture. Une arborescence peut être obtenue à partir d'un graphe en supprimant les circuits. Cela donne ici :

¹En réalité, ω désigne le cardinal de \mathbb{N} ensemble des entiers naturels.

A partir de l'arborescence de couverture, le graphe de couverture est obtenu en fusionnant les sommets qui correspondent au même marquage.

4.2 Construction de l'arborescence de couverture

La question est de savoir s'il existe un algorithme permettant de construire systématiquement l'arborescence de couverture. La réponse est oui. Il peut se décomposer en deux étapes.

Etape 1 A partir du marquage initial M_0 (qui est la racine de l'arborescence), on indique toutes les transitions qui sont validées ainsi que les marquages successeurs. Si l'un de ces marquages est strictement supérieur² à M_0 , on met ω pour chacune des composantes supérieures aux composantes correspondantes de M_0 .

Appliqué à l'exemple précédent, cela donne :

Etape 2 Pour chaque nouveau marquage M_k (sommet) de l'arborescence :

1. S'il existe sur le chemin de M_0 à M_k exclus un marquage $M_l = M_k$ alors M_k n'a pas de successeur.

Appliqué à l'exemple précédant, cela donne :

2. Sinon, on prolonge l'arborescence en ajoutant tous les successeurs de M_k . Pour chaque successeur M_l de M_k :
 - (a) on conserve les composantes ω de M_k
 - (b) s'il existe un marquage M_m sur le chemin de M_0 à M_l exclus tel que $M_l > M_m$ alors on met ω pour chacune des composantes de M_l strictement supérieure aux composantes de M_m .

Appliqué à l'exemple précédant, cela donne :

Exemple de mise en œuvre Appliquer l'algorithme précédent au RdP représenté Figure 4.2. En déduire le graphe de marquage.

4.3 Propriétés et intérêts de l'arborescence (graphe) de couverture

Théorème 4.3.1 *Pour tout RdP marqué, l'arborescence (graphe) de couverture est fini.*

²Il y a alors création d'au moins une nouvelle marque.

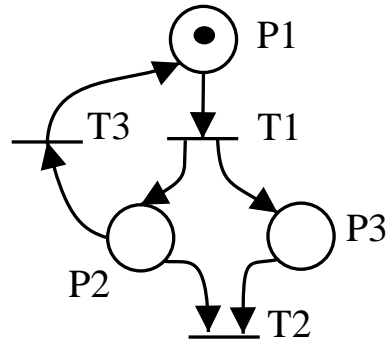


FIG. 4.2 – Exemples Graphes de marquage

Théorème 4.3.2 *Un RdP marqué (une place P_i dans un RdP marqué) est non borné si et seulement s'il existe un sommet de l'arborescence (graphe) de couverture tel qu'une composante d'un marquage (la composante $M(P_i)$) soit ω .*

Une propriété **Prop** concernant un ensemble E est **décidable** quand il existe un algorithme qui avec en entrée un élément e de E fournit en sortie la réponse OUI si e possède la propriété **Prop**, NON sinon. La propriété **être borné** d'un RdP est décidable.

Limite de graphes de couverture dans le cas d'un graphe de marquages de dimension infinie, le graphe de couverture ne permet pas d'analyser la vivacité. L'introduction de ω correspond à une perte d'information.

Chapitre 5

Etude des propriétés des Réseaux de Petri par algèbre linéaire

Dans ce chapitre, les réseaux de Petri généralisés sont introduits. Les RdPs ordinaires précédemment étudiés sont inclus dans les RdPs généralisés. De plus, l'analyse des RdPs généralisés par algèbre linéaire est présentée. Elle permet d'étudier des propriétés structurelles, c'est-à-dire les propriétés du RdP non marqué.

5.1 Notations et définitions

5.1.1 Structure

Les Réseaux de Petri ordinaires ont été définis dans la définition 2.1.1. Une définition alternative est maintenant présentée : son intérêt est de permettre, par une simple extension, l'introduction des Réseaux de Petri généralisés.

Definition 5.1.1 *Un RdP ordinaire non marqué est un quadruplet $Q = \langle P, T, Pre, Post \rangle$ où*

- $P = \{P_1, \dots, P_n\}$ est un ensemble fini non vide de places ;
- $T = \{T_1, \dots, T_m\}$ est un ensemble fini non vide de transitions ;
- $P \cap T = \emptyset$;
- $Pre : P \times T \rightarrow \{0, 1\}$ est l'application d'incidence avant telle que si un arc relie P_i à T_j , $Pre(P_i, T_j) = 1$, sinon $Pre(P_i, T_j) = 0$;
- $Post : P \times T \rightarrow \{0, 1\}$ est l'application d'incidence arrière telle que si un arc relie T_j à P_i , $Post(P_i, T_j) = 1$, sinon $Post(P_i, T_j) = 0$.

Exemple de l'atelier de coupe simplifié D'après son RdP représenté Figure 5.1, on a :

$$\begin{array}{l}
 i \in \{1, 2\} \quad Pre(P_i, T_2) = 1 \\
 i \in \{3, 4\} \quad Pre(P_i, T_2) = 0 \\
 \phantom{i \in \{3, 4\}} \quad Pre(P_3, T_3) = 1 \\
 i \in \{1, 2, 4\} \quad Pre(P_i, T_3) = 0
 \end{array}
 \left| \begin{array}{l}
 \phantom{i \in \{1, 2\}} \\
 i \in \{1, 2, 4\} \quad Post(P_i, T_2) = 0 \\
 i \in \{1, 4\} \quad Post(P_i, T_3) = 1 \\
 i \in \{2, 3\} \quad Post(P_i, T_3) = 0
 \end{array} \right.
 \begin{array}{l}
 Post(P_3, T_2) = 1 \\
 \\
 \\
 \\
 \end{array}$$

Definition 5.1.2 *Un RdP généralisé non marqué est défini comme un RdP ordinaire sauf que :*

- l'application d'incidence avant est définie par $Pre : P \times T \rightarrow \mathbb{N}$;
- l'application d'incidence arrière est définie par $Post : P \times T \rightarrow \mathbb{N}$.

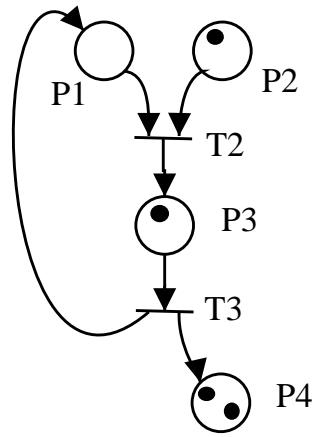


FIG. 5.1 – Atelier de coupe simplifié

A chaque arc reliant P_i à T_j (respectivement T_j à P_i) est ainsi associé un nombre naturel : $Pre(P_i, T_j)$ (resp. $Post(P_i, T_j)$). Ce nombre est appelé “poids” ou valuation et peut être différent de 1.

Sur la représentation graphique du RdP généralisé, ce nombre est indiqué à coté de l’arc correspondant (voir figure 5.2).

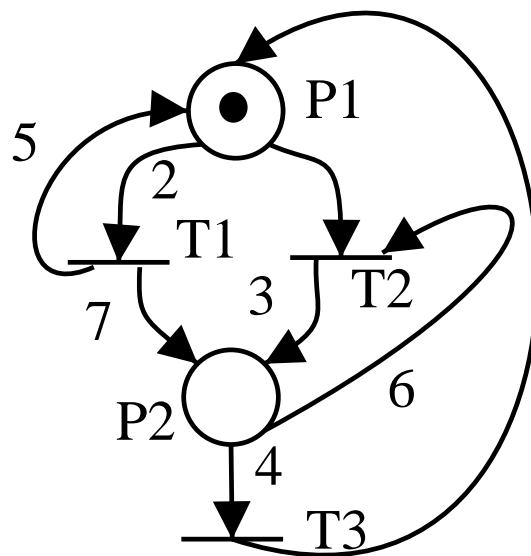


FIG. 5.2 – Exemple 1 Algèbre linéaire

A l’aide des applications Pre et $Post$, on peut définir les ensembles des places (transitions) d’entrée (de sortie) d’une transition (place) :

- Ensemble des places d’entrée de la transition T_j : ${}^oT_j = \{P_i \in P \mid Pre(P_i, T_j) > 0\}$;
- Ensemble des places de sortie de la transition T_j : $T_j^o = \{P_i \in P \mid Post(P_i, T_j) > 0\}$;
- Ensemble des transitions d’entrée de la place P_i : ${}^oP_i = \{T_j \in T \mid Post(P_i, T_j) > 0\}$;
- Ensemble des transitions de sortie de la place P_i : $P_i^o = \{T_j \in T \mid Pre(P_i, T_j) > 0\}$.

Afin de pouvoir les décrire de façon plus compacte, on associe aux applications Pre et $Post$ les matrices suivantes :

Matrice d'incidence avant associée à l'application Pre , elle est définie par :

$$W^- = [w_{ij}^-] \quad \text{avec} \quad w_{ij}^- = Pre(P_i, T_j)$$

Matrice d'incidence arrière associée à l'application $Post$, elle est définie par :

$$W^+ = [w_{ij}^+] \quad \text{avec} \quad w_{ij}^+ = Post(P_i, T_j)$$

Exemple de l'atelier de coupe simplifié D'après son RdP représenté Figure 5.1, on a :

Exemple Figure 5.2

5.1.2 Marquage, évolution

Definition 5.1.3 Un RdP marqué est un doublet $\langle Q, M_0 \rangle$ où Q est un RdP non marqué et où M_0 est le marquage initial.

Condition de franchissement Une transition T_j est franchissable si

$$\forall P_i \in {}^oT_j, \quad M(P_i) \geq Pre(P_i, T_j).$$

Cela signifie que, pour toute place P_i en entrée de la transition T_j , la transition T_j sera franchissable si le nombre de jetons présent dans chaque place P_i est supérieur ou égal à la valuation de l'arc reliant la place P_i à la transition T_j .

Après franchissement, on obtient un nouveau marquage M' donné par :

$$\forall P_i \in P, \quad M'(P_i) = M(P_i) - Pre(P_i, T_j) + Post(P_i, T_j).$$

Cela signifie que, pour toute place P_i en entrée de la transition T_j , le franchissement de la transition T_j consiste à enlever dans la place P_i un nombre de jetons égal à la valuation de l'arc reliant la place P_i à la transition T_j et pour toute place P_i en sortie de la transition T_j , le franchissement de la transition T_j consiste à ajouter dans la place P_i un nombre de jetons égal à la valuation de l'arc reliant la place P_i à la transition T_j .

Exemple On considère le RdP généralisé représenté Figure 5.3, gauche. La transition T_1 est franchissable car le nombre de marques dans P_1 est supérieur ou égal à la valuation de l'arc reliant P_1 à T_1 qui est de 2 et le nombre de marques de P_2 est supérieur ou égal à la valuation de l'arc reliant P_2 à T_1 . Le franchissement de T_1 consiste à enlever deux marques dans P_1 , une marque dans P_2 et à ajouter 5 marques dans P_3 .

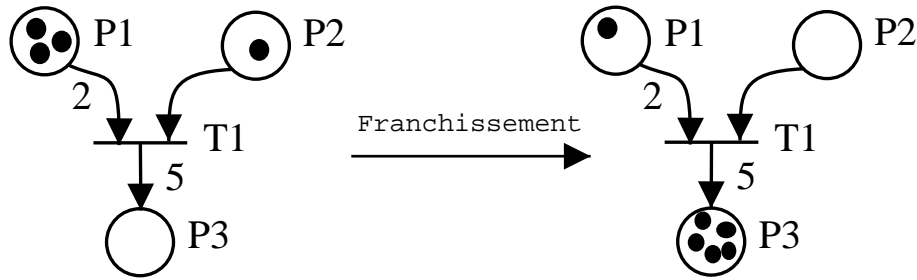
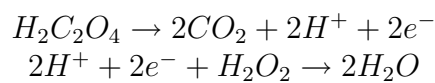


FIG. 5.3 – Exemple 2 Algèbre linéaire

Exemple d'un RdP généralisé : Modèle d'une réaction chimique On considère les réactions chimiques suivantes :



L'état du système est caractérisé par :

- La quantité de $H_2C_2O_4$ (nombre de marques dans la place P_1)
- La quantité de CO_2 (nombre de marques dans la place P_2)
- La quantité de H^+ (nombre de marques dans la place P_3)
- La quantité de e^- (nombre de marques dans la place P_4)
- La quantité de H_2O_2 (nombre de marques dans la place P_5)
- La quantité de H_2O (nombre de marques dans la place P_6)

L'état du système va évoluer quand :

- La première réaction chimique se produit (franchissement de la transition T_1)
- La deuxième réaction chimique se produit (franchissement de la transition T_2)

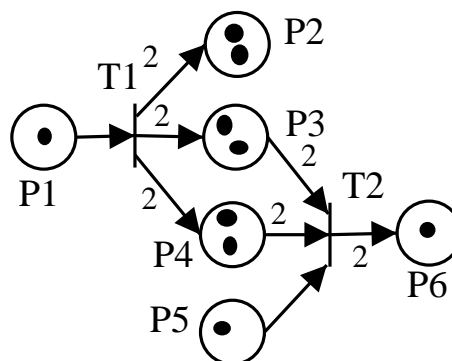


FIG. 5.4 – Réactions chimiques

On obtient alors le RdP représenté Figure 5.4. Le nombre de marques dans chaque place correspond ici à la quantité de chaque produit chimique.

Le franchissement d'une transition T_j modifie le marquage du RdP de la façon suivante :

$$\begin{bmatrix} M'(P_1) \\ \vdots \\ M'(P_n) \end{bmatrix} = \begin{bmatrix} M(P_1) \\ \vdots \\ M(P_n) \end{bmatrix} - \underbrace{\begin{bmatrix} Pre(P_1, T_j) \\ \vdots \\ Pre(P_n, T_j) \end{bmatrix}}_{\substack{w^- \\ \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \\ j}} + \underbrace{\begin{bmatrix} Post(P_1, T_j) \\ \vdots \\ Post(P_n, T_j) \end{bmatrix}}_{\substack{w^+ \\ \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}}}$$

Soit :

$$M' = M + \underbrace{(W^+ - W^-)}_{w \text{ matrice d'incidence}} \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

La j^{ieme} colonne de la matrice d'incidence W donne donc la variation du nombre de jetons dans chaque place lors du franchissement de la transition T_j .

Exemple de l'atelier de coupe

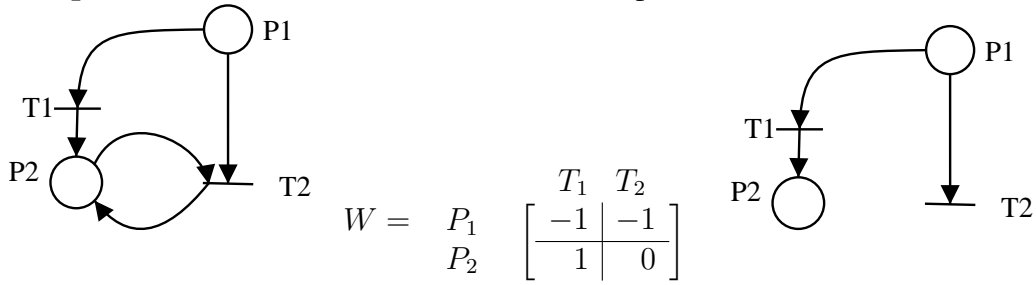
$$W = \begin{array}{c} P_1 \\ P_2 \\ P_3 \\ P_4 \end{array} \begin{array}{c|c} T_2 & T_3 \\ \hline -1 & 1 \\ -1 & 0 \\ \hline 1 & -1 \\ \hline 0 & 1 \end{array}$$

Le franchissement de la transition T_2 enlève une marque à P_1 et une marque à P_2 et ajoute une marque à P_3 . Le franchissement de la transition T_3 enlève une marque à P_3 et ajoute une marque à P_1 et une marque à P_4 .

Remarques

- La matrice d'incidence W est indépendante du marquage.
- La matrice d'incidence W est liée à la structure du RdP. Si le réseau est pur alors il est possible de reconstruire le RdP à partir de sa matrice d'incidence.

Exemple La matrice d'incidence suivante correspond aux deux RdPs ci-dessous



Dans le cas d'un RdP impur, on ne peut donc pas reconstruire le RdP à partir de sa matrice d'incidence. W ne contient donc pas (toujours) toute l'information sur la structure du RdP.

5.2 Equation fondamentale

Exemple A partir du RdP représenté Figure 5.5 :

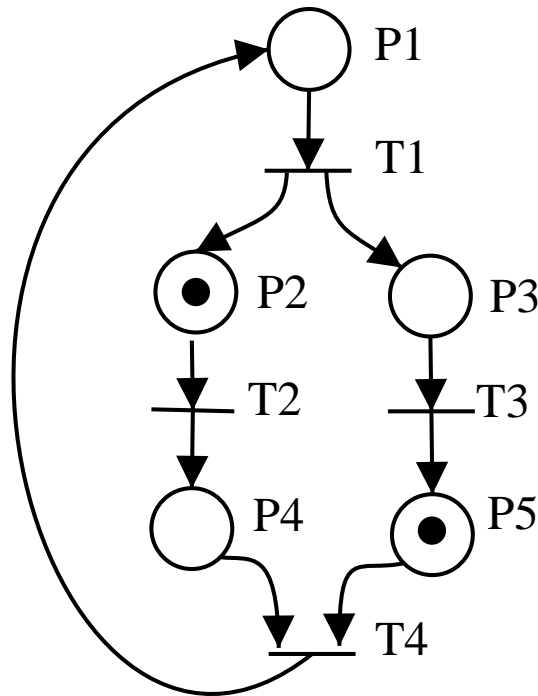


FIG. 5.5 – Exemple 3 Algèbre linéaire

$$\underbrace{\begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}}_{M_0} \xrightarrow{T_2} \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}}_{M_1} \xrightarrow{T_4} \underbrace{\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}}_{M_2} \xrightarrow{T_1} \underbrace{\begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}}_{M_3} \xrightarrow{T_3} \underbrace{\begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}}_{M_4}$$

$$\left\{ \begin{array}{l} M_1 = M_0 + W \\ M_2 = M_1 + W \end{array} \right. \begin{array}{l} \left[\begin{array}{c} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{array} \right] \\ \left[\begin{array}{c} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{array} \right] \end{array} \Rightarrow M_2 = M_0 + W \begin{array}{l} \left[\begin{array}{c} 0 \\ 1 \\ 0 \\ 1 \end{array} \right] \end{array}$$

Le vecteur $\begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \end{bmatrix}$ donne le nombre de franchissement de chaque transition dans la séquence $S = T_4T_2$.

Definition 5.2.1 *Le vecteur caractéristique d'une séquence de franchissement S , notée \underline{s} , est le vecteur de \mathbb{N}^m tel que \underline{s}_j est le nombre de franchissements de la transition T_j dans la séquence S .*

Exemple Pour $S = T_2T_4T_1T_2$, $\underline{s} = \begin{bmatrix} 1 \\ 2 \\ 0 \\ 1 \end{bmatrix}$.

Equation fondamentale Si S est une séquence franchissable à partir du marquage M_i qui mène au marquage M_k :

$$M_i | S > M_k$$

alors

$$M_k = M_i + W\underline{s} \quad (5.1)$$

où \underline{s} est le vecteur caractéristique associé à la séquence de franchissement S .

Exemple (suite) Pour le RdP Figure 5.5, la séquence $S = T_2T_4T_1T_2$ est franchissable à partir de M_0 . A quel marquage mène-t-elle ?

Pour le RdP Figure 5.5,

A la séquence de franchissement $S = T_2T_4T_1T_2$ correspond le vecteur caractéristique :

Par suite :

Remarques

- Deux séquences franchissables à partir d'un même marquage et de même vecteur caractéristique mènent au même marquage. Le marquage obtenu ne dépend donc pas de l'ordre dans lequel les transitions sont franchies.
- $ss' = s + s'$
- \underline{s} est appelé **vecteur caractéristique possible** s'il lui correspond au moins une séquence de franchissements S à partir du marquage M_k .

5.3 Pondération des places et invariants de marquage**5.3.1 Pondérations**

- Soit un vecteur $f = \begin{bmatrix} q_1 \\ \vdots \\ q_n \end{bmatrix} \in \mathbb{N}^n$. La composante q_i est un poids associé à la place P_i .
- Soit $P(f)$ l'ensemble des places P_i telles que $q_i \neq 0$.
- Soit S une séquence de transitions franchissables à partir de $M_0 : M_0 | S > M_k$

D'après l'équation (5.1) :

$$f^T M_k - f^T M_0 = f^T W \underline{s} \quad (5.2)$$

La quantité $f^T W \underline{s}$ correspond à un accroissement pondéré du nombre de jetons.

Exemple de la valeur associée à une production en cours On considère le montage d'un moteur sur une carrosserie pour construire une voiture. L'état du système est caractérisé par :

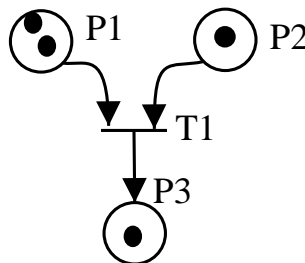


FIG. 5.6 – Assemblage voiture

- La quantité de moteurs en stock (nombre de marques dans la place P_1)
- La quantité de carrosseries en stock (nombre de marques dans la place P_2)
- La quantité de voitures en stock (nombre de marques dans la place P_3)

L'état du système va évoluer quand se produit :

- Le montage d'un moteur sur la caisse (franchissement de la transition T_1)

On obtient le RdP représenté Figure 5.6. Soit le vecteur de pondérations :

$$f = \begin{bmatrix} \text{valeur d'un moteur} \\ \text{valeur d'une carrosserie} \\ \text{valeur d'une voiture} \end{bmatrix}$$

$f^T W \times 1$ correspond à l'accroissement de la valeur induite par le montage d'un moteur sur une carrosserie.

$$W = \begin{matrix} P_1 \\ P_2 \\ P_3 \end{matrix} \begin{matrix} T_1 \\ \left[\begin{matrix} -1 \\ -1 \\ 1 \end{matrix} \right] \end{matrix}$$

D'où

$$f^T W \times 1 = \text{valeur d'une voiture} - (\text{valeur d'un moteur} + \text{valeur d'une carrosserie})$$

Au delà de leur interprétation, les accroissements pondérés permettent d'étudier les propriétés structurelles d'un RdP, c'est-à-dire des propriétés vérifiées pour toute condition initiale.

Propriété 5.3.1 (bornitude) *Le RdP est structurellement borné si et seulement si*

$$\exists f > 0, \quad f^T W \leq 0.$$

Propriété 5.3.2 (vivacité) *Il n'existe pas de marquage M_0 tel que le réseau de Petri $\langle Q, M_0 \rangle$ soit vivant si*

$$\exists f > 0, \quad f^T W \leq 0$$

(tous les éléments de $f^T W$ sont négatifs ou nuls avec au moins un qui est strictement négatif).

5.3.2 Invariants de marquage

Si $f^T W = 0$ alors, d'après l'équation (5.2) :

$$\forall M_k \in {}^*M_0, \quad f^T M_k = f^T M_0$$

On a donc $\sum q_i M(P_i) = \text{constante}$. $P(f)$ est donc une composante conservative. On peut donc calculer les composantes conservative en cherchant $f \neq 0$ tel que $f^T W = 0$.

Définition 5.3.1 *Le vecteur $f \neq 0$ tel que $f^T W = 0$ est appelé un **P-semi flot**.*

La connaissance de P-semi flots permet d'étudier la bornitude de places d'un RdP.

Propriété 5.3.3 *Soit $P(f)$ une composante conservative avec $f = \begin{bmatrix} q_1 \\ \vdots \\ q_n \end{bmatrix}$.*

Toutes les places P_i de $P(f)$ sont bornées et on a :

$$M(P_i) \leq \frac{f^T M_0}{q_i}.$$

Propriété 5.3.4 *Soient f et \bar{f} deux P-semi flots. Soient $\alpha \in \mathbb{N}$ et $\beta \in \mathbb{N}$.*

- $\alpha f + \beta \bar{f}$ est un P-semi flot avec $P(\alpha f + \beta \bar{f}) = P(f) \cup P(\bar{f})$
- Si $f \geq \bar{f}$ alors $f - \bar{f}$ est un P-semi flot.

Exemple Pour le RdP représenté Figure 5.5 : $f = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 0 \end{bmatrix}$ et $\bar{f} = \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 1 \end{bmatrix}$ sont deux P-semi flots. Alors, d'après Propriété 5.3.4

$$f + \bar{f} = \begin{bmatrix} 2 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

est aussi un P-semi flot. Comme $P(f + \bar{f}) = P$, le RdP est conservatif.

D'après Propriété 5.3.4, pour un RdP donné, s'il existe un P-semi flot, celui-ci n'est pas unique : on en a en réalité une infinité. Il est possible d'exprimer l'ensemble des P-semi flots comme une combinaison linéaire d'un nombre minimal de P-semi flots élémentaires f_1, f_2, \dots, f_q :

$$\forall \alpha_1 \in \mathbb{N}, \dots, \forall \alpha_q \in \mathbb{N}, \quad f = \sum_{i=1}^q \alpha_i f_i.$$

Pour décrire l'ensemble des P-semi flots d'un RdP, il suffit donc de déterminer les P-semi flots élémentaires. Leur détermination est illustrée par les exemples suivants.

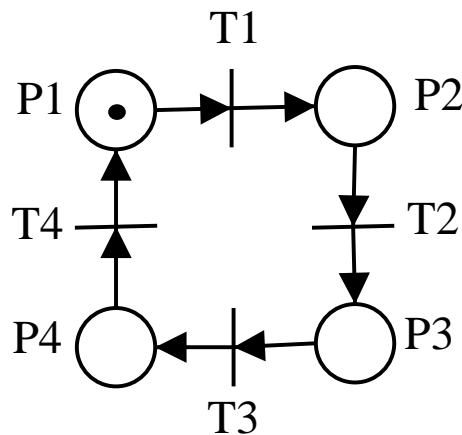


FIG. 5.7 – Quatre saisons

Exemple 4 saisons On reprend l'exemple des 4 saisons (RdP représenté Figure 5.7). La matrice d'incidence est donnée par :

Nous allons maintenant déterminer les solutions $f = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix}$ de l'équation $f^t W = 0$.

Donc, tout P-semi flot de ce RdP s'écrit :

Exemple Atelier de coupe simplifié On reprend l'exemple de l'atelier de coupe simplifié (RdP représenté Figure 3.1). La matrice d'incidence est donnée par :

q_1 et q_2 sont ici deux degrés de liberté. Par suite, tout P-semi flot s'écrit :

5.4 Composantes répétitives

Notation : $T(\underline{s})$ est l'ensemble des transitions qui apparaissent dans la séquence de franchissements S de vecteur caractéristique \underline{s} .

Propriété 5.4.1 *Un ensemble D de transitions est une composante répétitive stationnaire si et seulement s'il existe une séquence de franchissements S telle que $T(\underline{s}) = D$ et $W_{\underline{s}} = 0$. D est une composante répétitive croissante si $W_{\underline{s}} > 0$.*

Definition 5.4.1 *On appelle T-semi flot, tout vecteur \underline{s} tel que $W_{\underline{s}} = 0$.*

Attention, tout T-semi flot ne correspond pas forcément à une composante répétitive stationnaire. En effet, pour l'être, il est nécessaire qu'il corresponde à une séquence franchissable.

Propriété 5.4.2 *Si une séquence franchissable S est telle que \underline{s} est un T-semi flot alors une condition nécessaire pour que le RdP soit réinitialisable est que*

$$M | S > M.$$

5.5 Remarque sur le lien entre RdP ordinaire et RdP généralisé

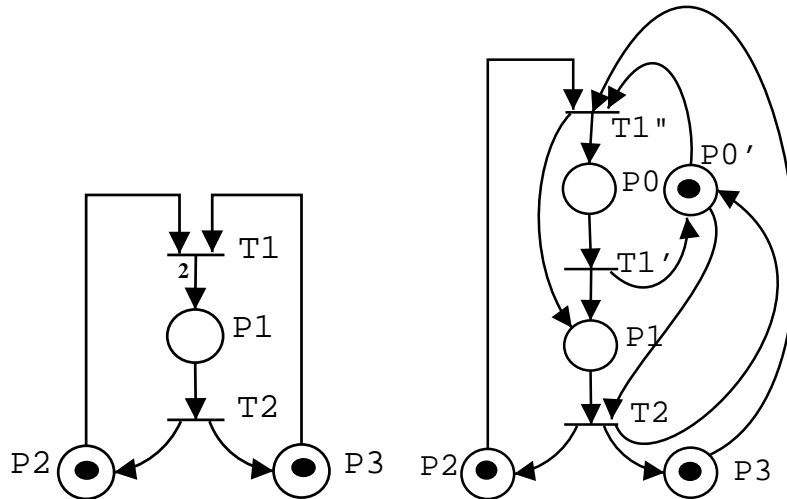


FIG. 5.8 – RdP généralisé et RdP ordinaire équivalent

Propriété 5.5.1 *Tout RdP généralisé peut être transformé en un RdP ordinaire.*

Un exemple de RdP généralisé et de RdP ordinaire équivalent est présenté Figure 5.8. L'intérêt de l'utilisation de RdP généralisés est évident : il permet d'obtenir des modèles RdP plus concis. Les RdPs généralisés font partie des classes de modèles RdPs appelées **abréviations**. Ces classes de modèles RdPs permettent de représenter les mêmes systèmes que les RdPs mais avec une représentation graphique beaucoup plus concise. Une seconde classe d'abréviations sera introduite dans le chapitre 7.

Chapitre 6

Etude des propriétés des Réseaux de Petri par réduction

Pour l'analyse des propriétés d'un RdP, des difficultés peuvent apparaître dans l'utilisation du graphe de marquages voire de l'algèbre linéaire dans le cas d'un RdP de taille significative.

L'objectif de ce chapitre est de présenter des règles permettant d'obtenir à partir d'un RdP marqué, un RdP marqué plus simple, c'est-à-dire avec un nombre réduit de places et un nombre réduit de transitions (règles de **réduction**). Ce dernier doit être

1. pour les propriétés étudiées, équivalent au RdP marqué de départ ;
2. suffisamment simple pour que l'analyse de ses propriétés soit simple.

En général, le RdP simplifié ne peut pas s'interpréter comme le modèle d'un système.

On peut distinguer deux types de règles de réduction :

1. règles de réduction préservant la vivacité et la bornitude (et leurs propriétés associées) ;
2. règles de réduction préservant les invariants de marquage.

Dans ce chapitre, on se concentrera sur les règles les plus simples et les plus utiles. La présentation se fera pour les RdP ordinaires : le cas des RdPs généralisés peut être traité, mutatis mutandis.

6.1 Règles de réduction préservant la vivacité et la bornitude

6.1.1 Règle de réduction R1 : substitution de places

Exemple introductif Sur le RdP Figure 6.1, si la transition T_1 est franchie alors la transition T_2 le sera tôt ou tard car le franchissement de la transition T_1 ajoute une marque dans la place P_2 rendant ainsi la transition T_2 franchissable, transition T_2 n'ayant pas d'autre place d'entrée. Le franchissement de la transition T_2 active alors la place P_3 . On aurait un comportement similaire si la place P_1 était en entrée d'une transition admettant P_3 comme place de sortie. On notera T_{12} la transition qui remplace ainsi l'ensemble transition T_1 , place P_2 et transition T_2 .

On avait déjà vu un tel procédé dans l'exemple du RdP représenté Figure 2.30, RdP modélisant de façon simplifié l'atelier de coupe. Dans ce RdP, la transition T_{23} représente elle-même un RdP représenté Figure 2.31. On est dans le même cas de figure que l'exemple précédent.

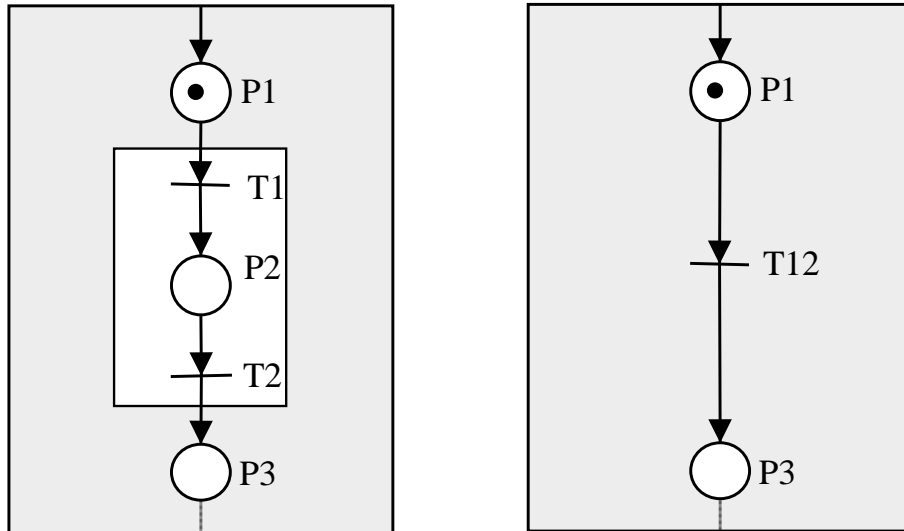


FIG. 6.1 – Réduction R1

Definition 6.1.1 (Place substituable) Une place P_i peut être *substituée*¹ si

1. Les transitions de sortie de P_i n'ont pas d'autre place d'entrée que P_i :

$$\forall T_j \in P_i^o, \quad {}^oT_j = \{P_i\}.$$

Le franchissement d'une transition d'entrée de P_i implique ainsi tôt ou tard le franchissement d'une transition de sortie de P_i .

2. Il n'existe pas de transition T_j qui soit à la fois transition d'entrée de P_i et transition de sortie de P_i :

$$P_i^o \cap {}^oP_i = \emptyset.$$

3. La place P_i admet au moins une transition de sortie qui n'est pas une transition puits :

$$\exists T_j \in P_i^o, \quad T_j^o \neq \emptyset.$$

Réduction R1 Dans le cas où P_i est une place substituable non marquée, la réduction R1 consiste à supprimer la place P_i . Les transitions d'entrée T_j et de sortie T_l sont remplacées par des transitions T_{jl} correspondant à toutes les combinaisons possibles d'une transition d'entrée avec une transition de sortie. Les places d'entrée de la transition T_{jl} sont les places d'entrée de la transition T_j ; les places de sortie de la transition T_{jl} sont les places de sortie des transitions T_j et T_l . Voir exemple Figure 6.2.

Dans le cas où P_i est une place substituable contenant une ou plusieurs marques, cette ou ces marques vont se retrouver dans les places de sortie des nouvelles transitions T_{jl} . Si plusieurs cas sont possibles, il est nécessaire de les considérer tous. Dans ce cas-là le RdP sera équivalent à plusieurs RdPs réduits, voir exemple Figure 6.3.

Propriétés conservées par la réduction R1 le RdP initial et le(s) RdP(s) réduit(s) sont équivalent pour les propriétés suivantes : *borné, sauf, vivant, quasi vivant, sans blocage, avec état d'accueil, conservatif*. Il y a néanmoins une perte d'information puisque s'ils sont bornés, la valeur de la borne n'est pas conservée (à l'exception de *sauf*). Les états d'accueil s'ils existent ne sont pas les mêmes.

¹Le terme supprimée serait plus adéquat : il ne sera pas utilisé car non consacré par l'usage.

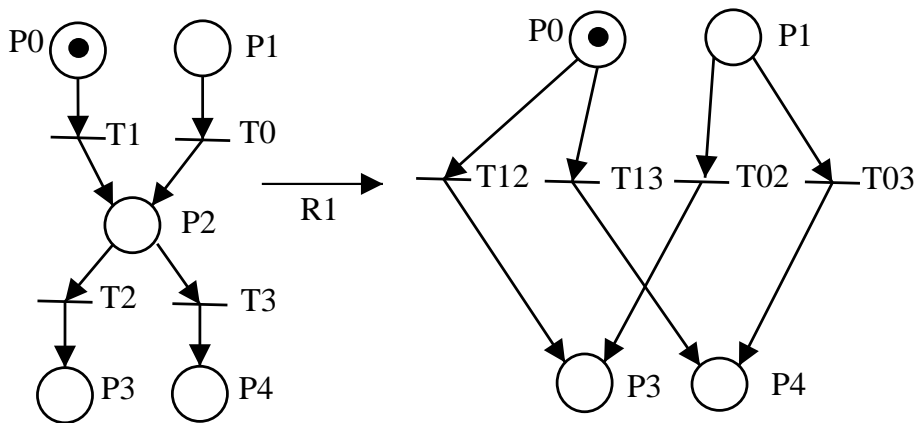
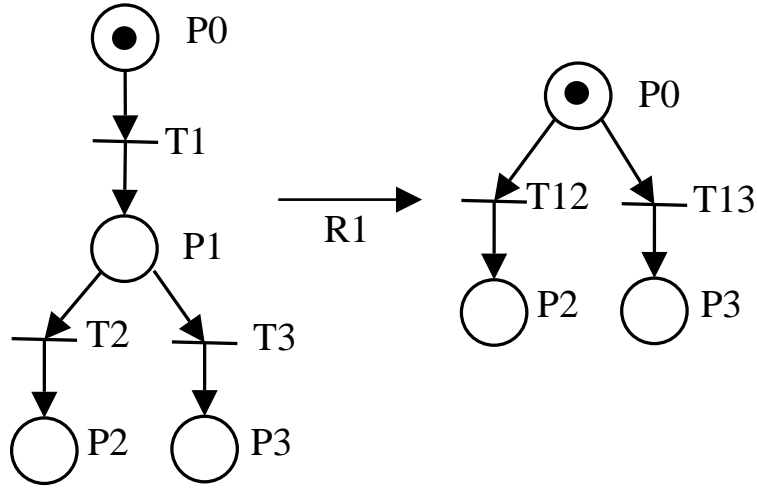
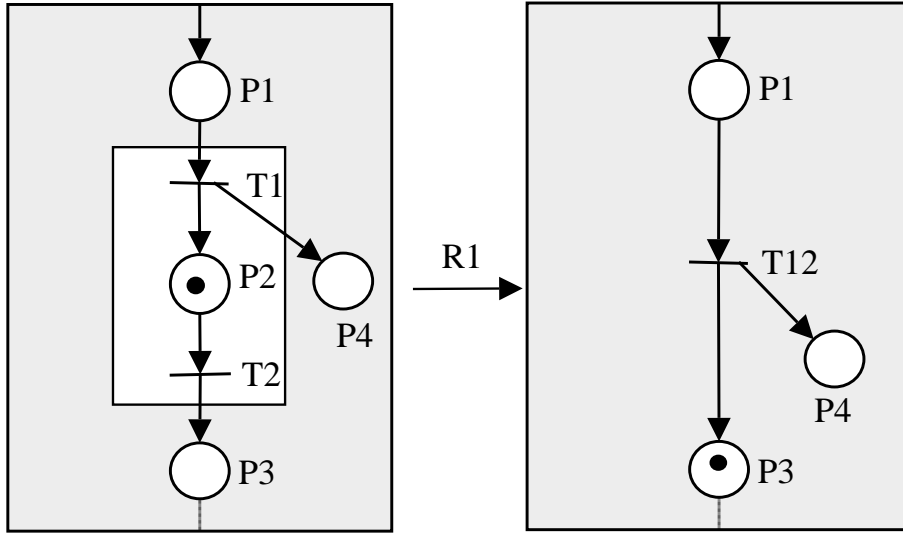


FIG. 6.2 – Réduction R1

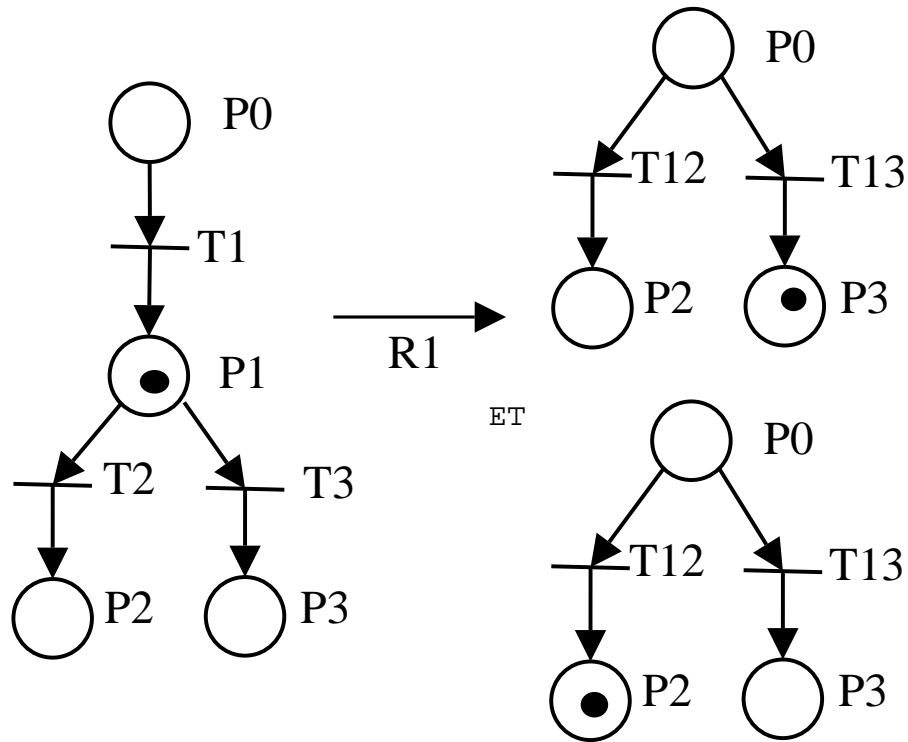


FIG. 6.3 – Réduction R1 place substituable marquée

6.1.2 Règle de réduction R2 : place implicite

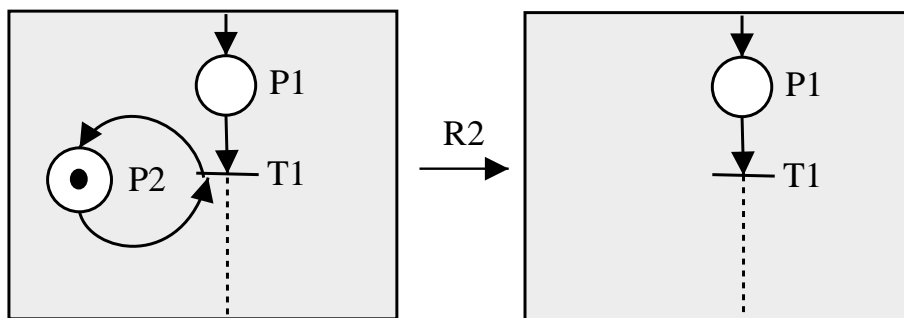


FIG. 6.4 – Réduction R2

Exemple introductif Sur le RdP représenté Figure 6.4, du fait du marquage initial, la place P_2 est ici toujours marquée. Le fait que la transition T_1 soit franchissable est donc indépendant de l'évolution du marquage de P_2 . Le RdP évoluerait donc de la même façon si la place P_2 n'existait pas : P_2 n'est pas un obstacle au franchissement de T_1 . Sa suppression n'aura donc pas d'influence sur l'évolution du RdP.

Definition 6.1.2 (Place implicite) Une place P_i est *implicite* si

1. Le marquage de P_i n'est jamais un obstacle au franchissement de ses transitions de sortie :

$$\forall T_j \in P_i^o, \quad \forall P_k \in {}^oT_j \setminus \{P_i\}, \quad M(P_k) \geq \text{Pre}(P_k, T_j) \implies M(P_i) \geq \text{Pre}(P_i, T_j)$$

2. Son marquage se déduit du marquage des autres places par la relation :

$$M(P_i) = \sum_{k \neq i} \alpha_k M(P_k) + \beta$$

où α_k et β sont des nombres rationnels positifs. On assure ainsi que si les places P_k avec $k \neq i$ sont bornées alors la place P_i est forcément bornée.

Dans l'exemple introductif :

1. Comme $M(P_2) = 1$ et $Pre(P_2, T_1) = 1$, on a toujours $M(P_2) \geq Pre(P_2, T_1)$.
2. Cette relation est vérifiée avec $\alpha_k = 0$ et $\beta = 1$.

Réduction R2 La réduction R2 consiste à supprimer la place implicite avec ses arcs en entrée et en sortie.

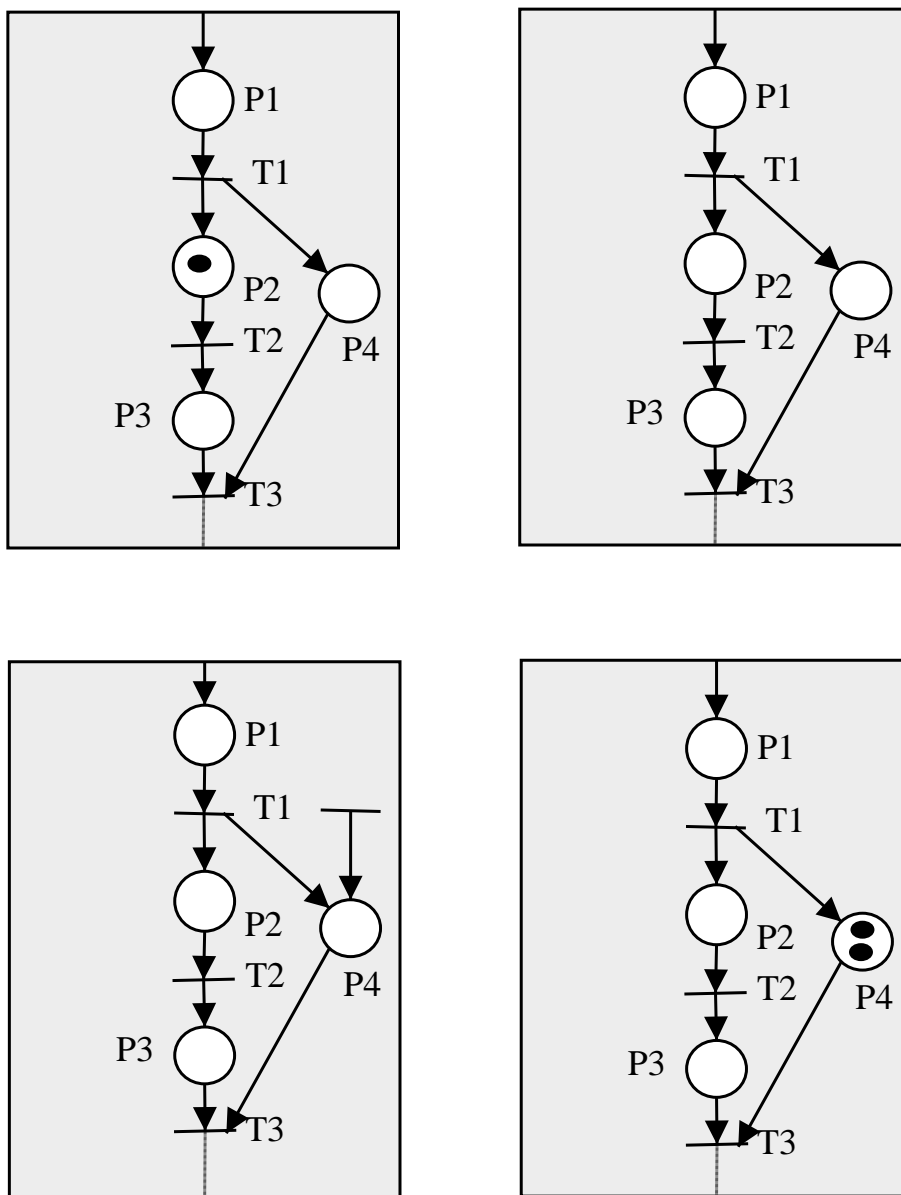


FIG. 6.5 – Réduction R2

Exemple Pour les 4 RdPs (partiels) représentés Figure 6.5, y-a-t-il une place implicite ?

Propriétés conservées par la réduction R2 le RdP initial et le(s) RdP(s) réduit(s) sont équivalents pour les propriétés suivantes : *borné, vivant, quasi vivant, sans blocage, avec état d'accueil, conservatif*. En général, la propriété *sauf* n'est pas conservée.

6.1.3 Règle de réduction R3 : transition neutre

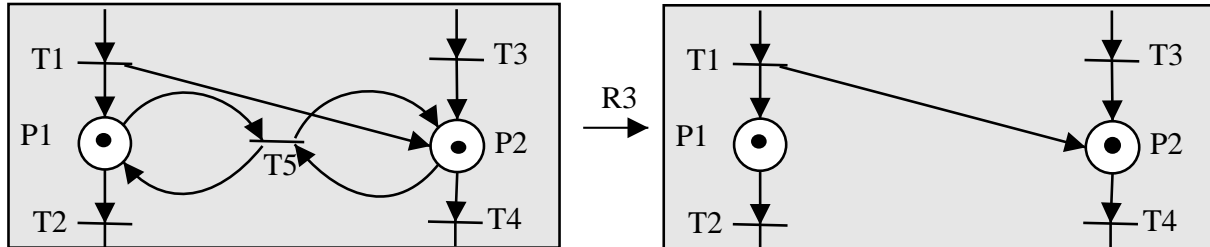


FIG. 6.6 – Réduction R3

Exemple introductif Sur le RdP Figure 6.6, le franchissement de la transition T_5 n'entraîne pas la modification du marquage. Elle ne participe donc pas à l'évolution du RdP. On peut donc simplifier le RdP en la supprimant.

Definition 6.1.3 (Transition neutre) Une transition est *neutre* si l'ensemble de ces places d'entrée est égal à l'ensemble de ses places de sortie :

$${}^oT_j = T_j^o.$$

Réduction R3 La réduction R3 consiste à supprimer la transition neutre T_j avec l'ensemble de ses arcs en entrée et en sortie. Elle ne sera effectuée que s'il existe une transition $T_l \neq T_j$ telle que son franchissement mette suffisamment de marques dans les places d'entrée de T_j pour la rendre franchissable :

$$\exists T_l \neq T_j, \forall P_i \in {}^oT_j, \quad Post(P_i, T_l) \geq Pre(P_i, T_j).$$

Propriétés conservées par la réduction R3 le RdP initial et le(s) RdP(s) réduit(s) sont équivalent pour les propriétés suivantes : *borné, sauf, vivant, quasi vivant, sans blocage, avec état d'accueil, conservatif*.

6.1.4 Règle de réduction R4 : transitions identiques

Exemple introductif Dans l'exemple de RdP Figure 6.7, les transitions T_1 et T_2 sont identiques : mêmes places d'entrée, mêmes places de sortie.

Definition 6.1.4 (Transitions identiques) Deux transitions T_j et T_l sont *identiques* si elles ont mêmes ensembles de places d'entrée et mêmes ensembles de places de sortie :

$${}^oT_j = {}^oT_l \quad \text{et} \quad T_j^o = T_l^o.$$

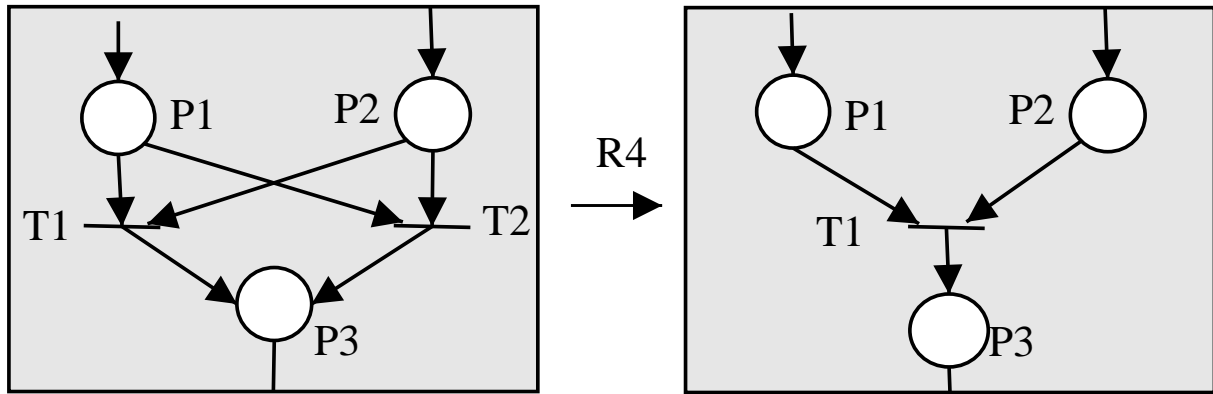


FIG. 6.7 – Réduction R4

Réduction R4 La réduction R4 consiste à supprimer l'une des transitions identiques avec l'ensemble de ses arcs en entrée et en sortie.

Propriétés conservées par la réduction R4 le RdP initial et le RdP réduit sont équivalents pour les propriétés suivantes : *borné, vivant, sauf, quasi vivant, sans blocage, avec état d'accueil, conservatif.*

6.1.5 Exemple de mise en œuvre des différentes règles R1 à R4

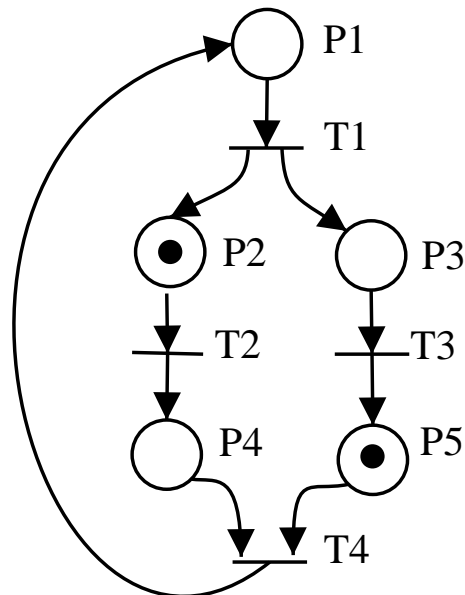


FIG. 6.8 – Exemple mise en œuvre réduction R1 à R4

Mettre en œuvre les règles R1 à R4 pour la réduction du RdP représenté Figure 6.8.

6.2 Règles de réduction préservant les invariants de marquage

6.2.1 Règle de réduction Ra : transition impure

Definition 6.2.1 (Transition impure) une transition T_j est *impure* si elle admet au moins une place d'entrée qui est aussi place de sortie :

$${}^{\circ}T_j \cap T_j^{\circ} \neq \emptyset.$$

Soit :

$$\exists P_i \in P, P_i \in {}^{\circ}T_j \text{ et } P_i \in T_j^{\circ}.$$

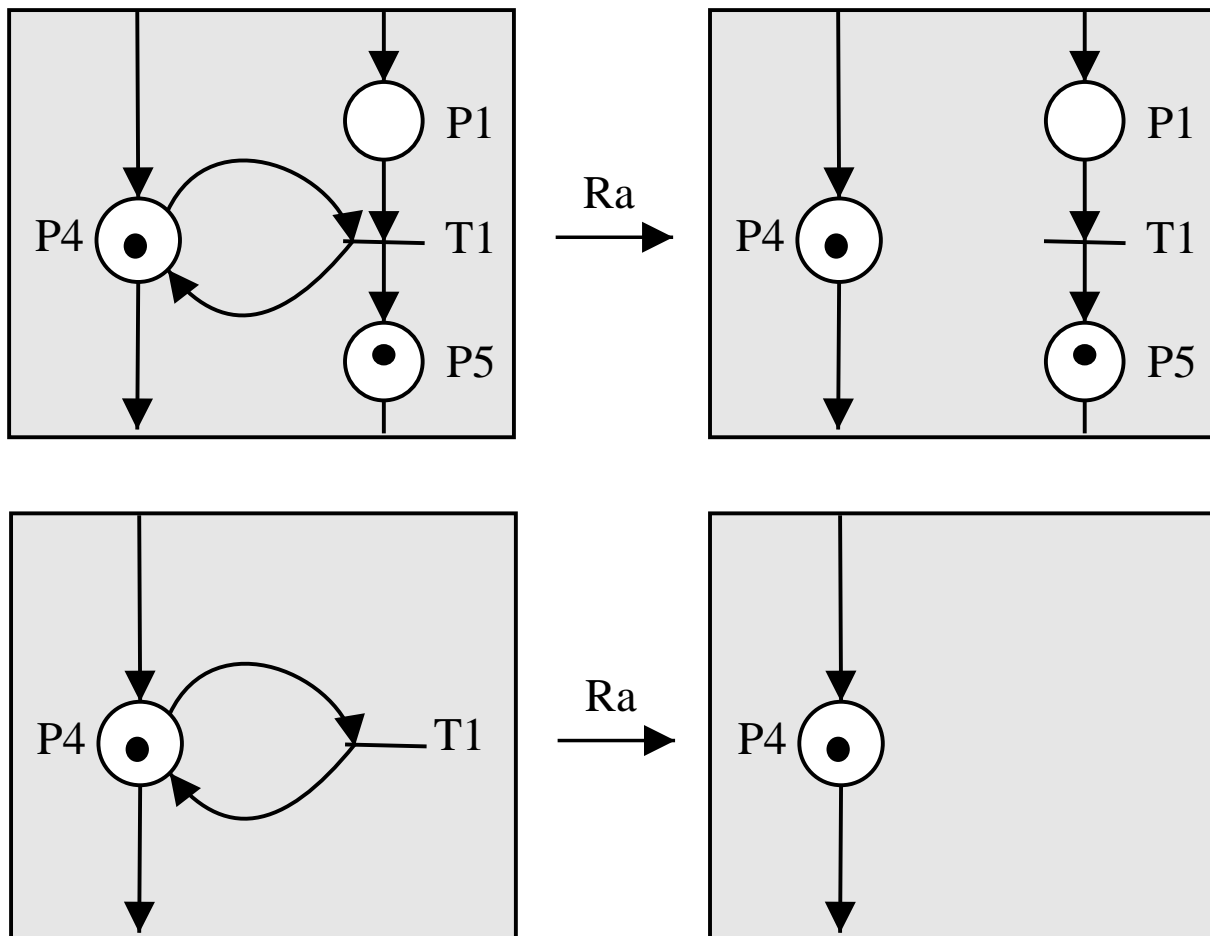


FIG. 6.9 – Réduction Ra

Réduction Ra Pour une transition impure T_j , la réduction Ra consiste à :

1. supprimer les arcs reliant T_j à la place P_i qui est à la fois place d'entrée de T_j et place de sortie de T_j ;
2. supprimer la transition T_j si elle n'a plus de place en entrée et si elle n'a plus de place en sortie.

Exemple Voir exemple Figure 6.9.

6.2.2 Règle de réduction Rb : transition pure

Definition 6.2.2 (Transition pure) une transition T_j est *pure* si elle n'est pas impure.

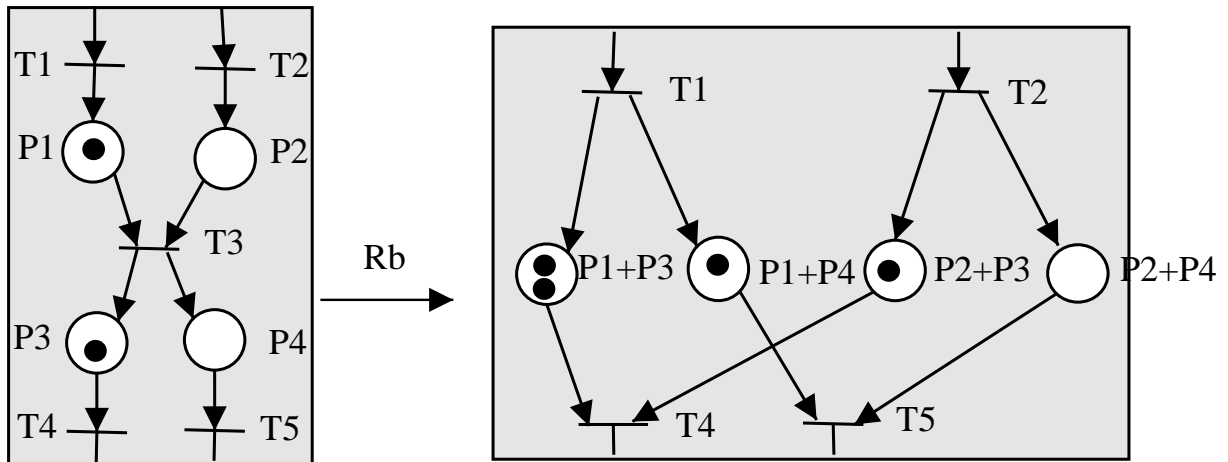


FIG. 6.10 – Réduction Rb

Réduction Rb Pour une transition pure T_j , la réduction Rb s'applique sous la condition que T_j ait au moins une place en entrée et une place en sortie :

$${}^oT_j \neq \emptyset \quad \text{et} \quad T_j^o \neq \emptyset.$$

La réduction Rb consiste à :

1. Supprimer la transition T_j ;
2. A tout couple de places (P_i, P_k) avec $P_i \in {}^oT_j$ et $P_k \in T_j^o$, on associe une place notée $P_i + P_k$ dont le marquage est donné par : $M(P_i + P_k) = M(P_i) + M(P_k)$.
3. Les transitions d'entrée de $P_i + P_k$ sont les transitions d'entrée de P_i et de P_k , la transition T_j exceptée, les transitions de sortie de $P_i + P_k$ sont les transitions de sortie de P_i et de P_k , la transition T_j exceptée :

$${}^o(P_i + P_k) = {}^oP_i \cup {}^oP_k \setminus \{T_j\} \quad \text{et} \quad (P_i + P_k)^o = P_i^o \cup P_k^o \setminus \{T_j\}$$

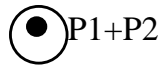
Exemple Voir exemple Figure 6.10.

6.2.3 Exemple de mise en œuvre des règles Ra et Rb

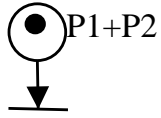
Mettre en œuvre les règles Ra et Rb pour la réduction du RdP représenté Figure 6.8.

6.2.4 Remarques

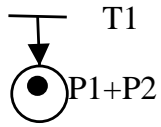
Quand les réductions Ra et Rb ne s'appliquent plus, on est dans un cas similaires aux 4 cas suivants :



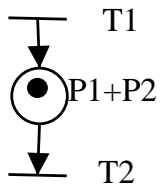
$\{P_1, P_2\}$ forme une composante conservative associée à l'invariant de marquage : $M(P_1) + M(P_2) = 1$



La transition T_1 (si vivante) étant une transition puits, le marquage de P_1 peut être nul



P_1 ou P_2 est non bornée (T_1 vivante)



Le RdP réduit obtenu dépend de l'ordre dans lesquelles les opérations ont été effectuées. Par exemple, réduire le RdP représenté Figure 6.11 en commençant par effectuer la réduction Rb sur T_1 puis recommencer en commençant par effectuer la réduction Rb sur T_2 . Comparer les deux RdPs réduits obtenus.

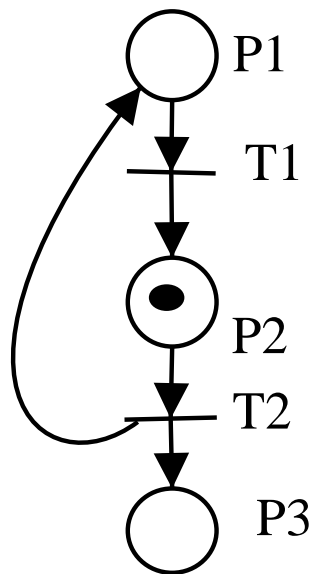


FIG. 6.11 – Réduction Ra et Rb

Chapitre 7

Réseaux de Petri colorés

La modélisation d'un système réel peut mener à des réseaux de Petri de taille trop importante rendant leur manipulation et/ou leur analyse difficile. La question est alors de modifier (étendre) la modélisation par RdP de façon à obtenir des modèles RdP de plus petite taille. Cette question a motivé l'introduction des RdPs généralisés, chapitre 5. Elle a été discutée dans la section 5.5. Une telle classe de modèles y a été désignée par le terme d'abréviation. Dans le chapitre présent, nous introduisons une autre classe d'abréviation, les RdPs colorés. Ils sont importants pour la modélisation de systèmes de production (au sens large). Ils ont été motivés par le fait suivant.

Une taille trop importante peut découler du fait que l'on ne peut pas distinguer entre elles les différentes marques d'une place. Plusieurs marques dans une place peuvent modéliser un certain nombre de pièces identiques dans un stock. Si le stock contient plusieurs types de pièces, des places supplémentaires doivent être introduites pour sa modélisation.

Voir par exemple Figure 7.1. Le nombre de marques dans la place P_3 correspond au nombre de pièces de type 1 en stock. Le nombre de marques dans la place P_4 correspond au nombre de pièces de type 2 en stock. Le nombre de marques dans la place P_7 indique le nombre d'emplacements libres dans le stock. Si un stock est susceptible de contenir deux (respectivement n) types de pièces, il doit donc être modélisé par au moins 3 (resp. $n + 1$) places. Peut-on imaginer une modification du modèle RdP qui permette de la modéliser par un nombre de places

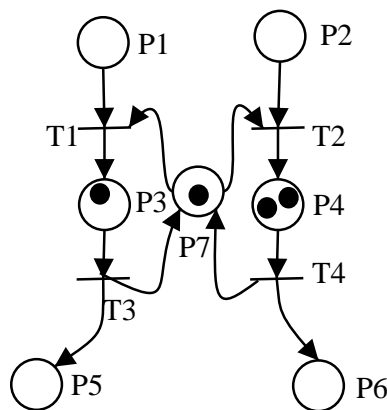


FIG. 7.1 – Stock contenant deux types de pièces

indépendant du nombre de types de pièces stockées ? Intuitivement, si on est capable de distinguer les différentes marques les unes des autres, on pourrait associer à chaque type de marques un type de pièces. Deux places seraient alors nécessaires pour modéliser le stock : une place dont le marquage indique le nombre (et le type) de pièces stockées et une place dont le marquage

indique le nombre d'emplacements libres. Pour distinguer les marques les unes des autres, on introduit la notion de couleur.

7.1 Introduction aux RdPs colorés à travers des exemples

Exemple 1 : introduction de marques colorées Un système de production est constitué d'une première machine avec son stock en sortie de capacité limitée à 3 produisant la pièce a et d'une seconde machine avec son stock en sortie de capacité limitée à 3 produisant la pièce b. L'état du système est caractérisé par :

- La première machine produit la pièce a (une marque dans la place Pa_1)
- La première machine est en attente (une marque dans la place Pa_3)
- Le nombre de pièces dans le stock de sortie de la première machine (nombre de marques dans la place Pa_2)
- Le nombre de places libres dans le stock de sortie de la première machine (nombre de marques dans la place Pa_4)
- La seconde machine produit la pièce b (une marque dans place Pb_1)
- La seconde machine est en attente (une marque dans place Pb_3)
- Le nombre de pièces dans le stock de sortie de la seconde machine (nombre de marques dans la place Pb_2)
- Le nombre de places libres dans le stock de sortie de la seconde machine (nombre de marques dans la place Pb_4)

L'état du système va évoluer quand se produit :

- le début de la production d'une pièce a par la première machine (franchissement de Ta_1)
- la fin de la production d'une pièce a par la première machine (franchissement de la transition Ta_2)
- le retrait d'une pièce a du stock de la première machine (franchissement de la transition Ta_3)
- le début de la production d'une pièce b par la seconde machine (franchissement de Tb_1)
- la fin de la production d'une pièce b par la seconde machine (franchissement de la transition Tb_2)
- le retrait d'une pièce b du stock de la seconde machine (franchissement de la transition Tb_3)

Au départ, la première machine attend tandis que la deuxième machine produit une pièce b. Le stock de sortie de la première machine contient une pièce a et celui de la seconde deux pièces b. On obtient le RdP représenté Figure 7.2 gauche. Ce système est constitué de deux sous systèmes identiques (première machine avec son stock et seconde machine avec son stock) chacun produisant un type de pièces différent. Cela suggère l'idée que l'on pourrait superposer les RdPs représentant chacun des sous systèmes. Par contre, dans le RdP représentant le premier sous système, une marque dans une place fait référence à tout ce qui a trait à la pièce a : première machine occupée ou non, état du stock de la machine a. Dans le second RdP, une marque fait référence à la pièce b. Si on superpose les deux RdPs, il est donc nécessaire de différencier deux types de marque : une première que l'on baptise $\langle a \rangle$ qui correspond au marquage du premier RdP et une seconde baptisée $\langle b \rangle$ correspondant au marquage du second RdP. On parle alors de **couleur**¹. L'ensemble des couleurs est noté $C = \{ \langle a \rangle, \langle b \rangle \}$. On obtient alors le **RdP coloré** représenté Figure 7.2, droite. Chaque transition est alors validée pour une couleur.

- La transition T_1 est validée pour la couleur $\langle a \rangle$ mais pas pour la couleur $\langle b \rangle$: le

¹Le mot couleur fait évidemment référence à des marques qui seraient de couleur différente. Le document étant reprographié en noir et blanc, on remplace le point de couleur par une lettre entre \langle et \rangle .

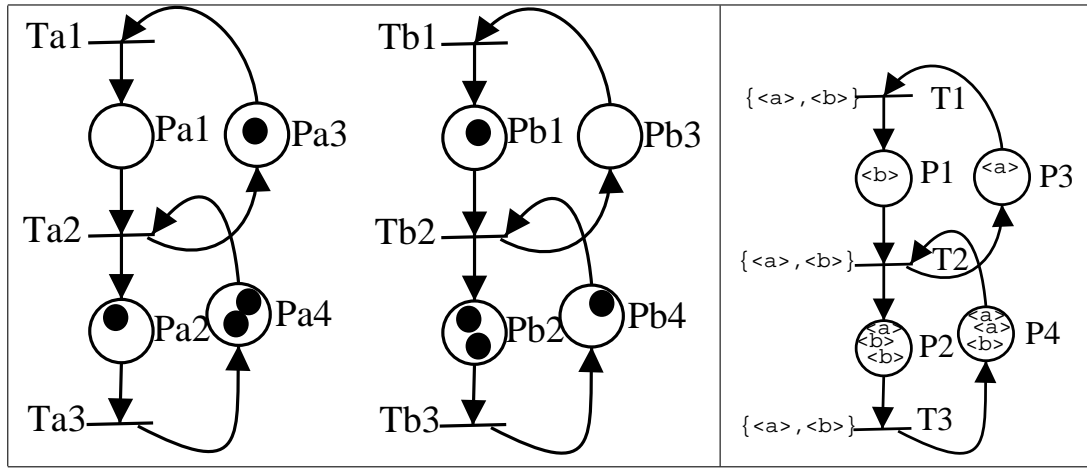


FIG. 7.2 – Machines a et b avec leurs stocks de sortie, RdP (gauche) et RdP coloré (droite)

franchissement de T_1 pour la couleur $\langle a \rangle$ consiste à enlever une marque $\langle a \rangle$ dans la place P_3 et à ajouter une marque $\langle a \rangle$ dans la place P_1 ;

- La transition T_2 est validée pour la couleur $\langle b \rangle$ mais pas pour la couleur $\langle a \rangle$: le franchissement de T_2 pour la couleur $\langle b \rangle$ consiste à enlever une marque $\langle b \rangle$ dans les places P_1 et P_4 et à ajouter une marque $\langle b \rangle$ dans les places P_2 et P_3 ;
- La transition T_3 est validée pour les couleurs $\langle a \rangle$ et $\langle b \rangle$: le franchissement de T_3 pour la couleur $\langle a \rangle$ (respectivement $\langle b \rangle$) consiste à enlever une marque $\langle a \rangle$ (resp. $\langle b \rangle$) dans la place P_2 et à ajouter une marque $\langle a \rangle$ (resp. $\langle b \rangle$) dans la place P_4 .

On associe à chaque transition l'ensemble des couleurs pour lesquelles le franchissement est envisagé : par exemple pour T_1 , l'ensemble $\{\langle a \rangle, \langle b \rangle\}$.

Le passage du RdP (Figure 7.2 gauche) au RdP coloré (Figure 7.2 droite) est appelé **pliage**, le passage du RdP coloré au RdP **dépliage**.

Le marquage d'un RdP coloré est maintenant défini pour chaque couleur. Par exemple, sur le RdP coloré représenté Figure 7.2 droite, le marquage

- pour la couleur $\langle a \rangle$ est :
$$\begin{bmatrix} 0 \\ 1 \\ 1 \\ 2 \end{bmatrix} ;$$

- pour la couleur $\langle b \rangle$ est :
$$\begin{bmatrix} 1 \\ 2 \\ 0 \\ 1 \end{bmatrix} .$$

Le marquage peut être ainsi défini par une matrice où chaque colonne correspond à une couleur :

$$M_0 = \begin{matrix} & \langle a \rangle & \langle b \rangle \\ \begin{matrix} P_1 \\ P_2 \\ P_3 \\ P_4 \end{matrix} & \begin{bmatrix} 0 & 1 \\ 1 & 2 \\ 1 & 0 \\ 2 & 1 \end{bmatrix} \end{matrix}$$

Il peut être aussi défini comme une application qui à une couleur donnée associe un marquage :

$$\langle a \rangle \mapsto \begin{bmatrix} 0 \\ 1 \\ 1 \\ 2 \end{bmatrix} \quad \text{et} \quad \langle b \rangle \mapsto \begin{bmatrix} 1 \\ 2 \\ 0 \\ 1 \end{bmatrix}$$

Le marquage est donc une application de $C \rightarrow \mathbb{N}^4$. Pour ne pas alourdir l'écriture, cette application est définie en utilisant des notations formelles :

$$\begin{bmatrix} 0 \\ 1 \\ 1 \\ 2 \end{bmatrix} \langle a \rangle + \begin{bmatrix} 1 \\ 2 \\ 0 \\ 1 \end{bmatrix} \langle b \rangle = \begin{bmatrix} \langle b \rangle \\ \langle a \rangle + 2\langle b \rangle \\ \langle a \rangle \\ 2\langle a \rangle + \langle b \rangle \end{bmatrix}.$$

Pour une place donnée, le marquage est une application de $C \rightarrow \mathbb{N}$.

Quand il n'y a pas de couleur associée à une marque, on parle d'une marque de **couleur neutre**. Elle est notée $\langle \bullet \rangle$.

Exemple 2 : introduction de fonctions associées aux arcs Un système de production est constitué d'une seule machine avec son stock en sortie de capacité limitée à 3. Cette machine produit soit la pièce a soit la pièce b. Le RdP de ce système de production peut être obtenu à partir des RdPs modélisant les deux sous systèmes dans l'exemple précédent par² :

- la mise en commun de la machine pour la fabrication des pièces a et b (fusion des places Pa_3 et Pb_3 en la place P_3);
- la mise en commun du stock de sortie de la machine (fusion des places Pa_4 et Pb_4 en la place P_4).

Au départ, une pièce a a déjà été produite et stockée. La machine est disponible. On obtient alors le RdP représenté Figure 7.3, à gauche. Le RdP coloré obtenu par pliage est représenté

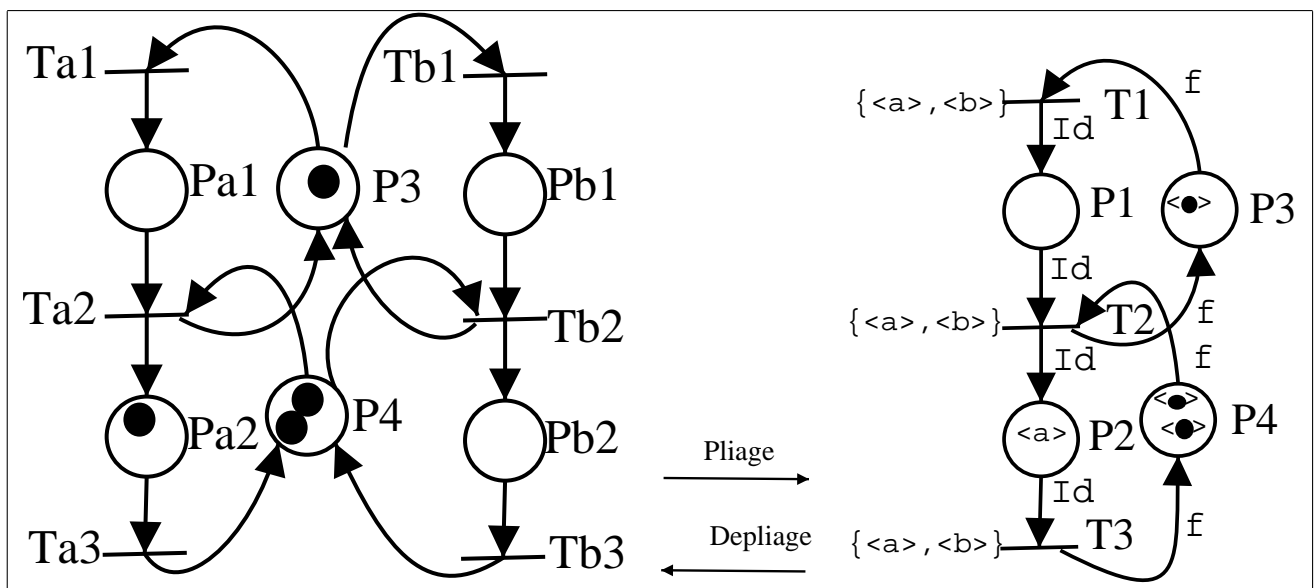


FIG. 7.3 – Machine avec son stock de sortie produisant a et b, RdP (gauche) et RdP coloré (droite)

Figure 7.3, à droite :

- les places Pa_1 et Pb_1 se plient en la place P_1 : une marque dans la place Pa_1 correspond à une marque $\langle a \rangle$ dans P_1 , une marque dans la place Pb_1 correspond à une marque $\langle b \rangle$ dans P_1 ;
- les places Pa_2 et Pb_2 se plient en la place P_2 : une marque dans la place Pa_2 correspond à une marque $\langle a \rangle$ dans P_2 , une marque dans la place Pb_2 correspond à une marque $\langle b \rangle$ dans P_2 ;

²voir sous section 2.2.2.2, page 31.

- les marquages des places P_3 et P_4 ne sont pas modifiés ;
- les transitions Ta_1 et Tb_1 se plient en la transition T_1 ; de même, Ta_2 et Tb_2 se plient en T_2 et Ta_3 et Tb_3 se plient en T_3 .

Comment définir la validation d'une transition ? Dans le cas de la transition T_1 , une marque de couleur neutre dans la place P_3 signifie que la machine est disponible pour commencer à produire la pièce a (franchissement de la transition T_1 pour la couleur $\langle a \rangle$) ou pour commencer à produire la pièce b (franchissement de la transition T_1 pour la couleur $\langle b \rangle$). Cela est décrit dans le RdP coloré en associant une fonction f à l'arc reliant la place P_3 à la transition T_1 . L'argument de cette fonction est la couleur pour laquelle la validation de la transition T_1 est envisagée. La valeur prise par cette fonction correspond au nombre de marques (quelles que soient leurs couleurs) nécessaire pour valider la transition pour la couleur considérée. Il s'agit donc d'une application de C dans l'ensemble des applications de $C \rightarrow \mathbb{N}$, ce qui est noté $C \rightarrow (C \rightarrow \mathbb{N})$. Dans l'exemple, pour valider la transition T_1

- pour la couleur $\langle a \rangle$, P_3 doit avoir au moins une marque de couleur neutre :

$$f(\langle a \rangle) = \langle \bullet \rangle$$

- pour la couleur $\langle b \rangle$, P_3 doit avoir au moins une marque de couleur neutre :

$$f(\langle b \rangle) = \langle \bullet \rangle .$$

$Pre(P_3, T_1)$ n'est plus ici un entier naturel mais une application de C dans l'ensemble des applications de $C \rightarrow \mathbb{N}$. Par simplicité, on préfère définir $Pre(P_3, T_1)$ à l'aide de fonctions annexes comme la fonction $f : Pre(P_3, T_1) = f$.

A partir du moment où la place P_3 contient au moins une marque de couleur neutre, le franchissement de T_1 peut donc se faire pour la couleur $\langle a \rangle$ ou la couleur $\langle b \rangle$. Dans l'exemple considéré, il consiste à retirer une marque de la place P_2 et à ajouter dans la place P_1 une marque de la couleur pour laquelle le franchissement de T_1 a été décidé. On dit que la **fonction identité** notée Id est associée à l'arc reliant la transition T_1 à la place P_1 :

$$Id(\langle a \rangle) = \langle a \rangle \quad \text{et} \quad Id(\langle b \rangle) = \langle b \rangle .$$

La fonction identité peut être définie comme une application de C dans $C \rightarrow \{0, 1\}$ telle que

$$\begin{cases} Id(\langle c_k \rangle)(\langle c_k \rangle) = 1 \\ \forall l \neq k, Id(\langle c_k \rangle)(\langle c_l \rangle) = 0 \end{cases}$$

Dans le cas de la transition T_2 , le franchissement peut se faire par rapport à la couleur $\langle a \rangle$ (resp. $\langle b \rangle$) : si une marque $\langle a \rangle$ (resp. $\langle b \rangle$) est disponible dans la place P_1 et une marque $\langle \bullet \rangle$ est disponible dans P_3 , le franchissement consiste alors à enlever ces marques pour mettre une marque de couleur $\langle a \rangle$ (resp. $\langle b \rangle$) dans P_2 et une marque de couleur neutre dans P_3 . La fonction Id est donc associée aux arcs reliant P_1 à T_2 et T_2 à P_2 . La fonction f est associée à l'arc reliant T_2 à P_3 . Ce raisonnement s'applique aussi au reste du RdP.

Suite de l'exemple 2 : introduction de couleurs complexes Lorsque l'on examine le RdP coloré représenté Figure 7.3, on constate qu'il est constitué par la double répétition d'un même motif. On peut donc envisager une nouvelle opération de pliage.

- Les places P_1 et P_2 se plient en une place P_u : il est alors nécessaire de distinguer entre les pièces a (resp. b) dans la machine en cours de fabrication et celles qui sont stockées. Pour cela, la couleur est maintenant définie par un doublet $\langle ., . \rangle$ dont le premier terme fait référence au type de pièce (a ou b) et le second au lieu où se trouve la pièce (m pour la machine ou s pour le stock). On parle alors de **couleur complexe**.

- Les places P_3 et P_4 se plient en une place P_r . Dans cette place, il faut donc distinguer entre une marque traduisant la disponibilité de la machine (couleur $\langle m \rangle$) et une marque traduisant la disponibilité d'un emplacement dans le stock (couleur $\langle s \rangle$).

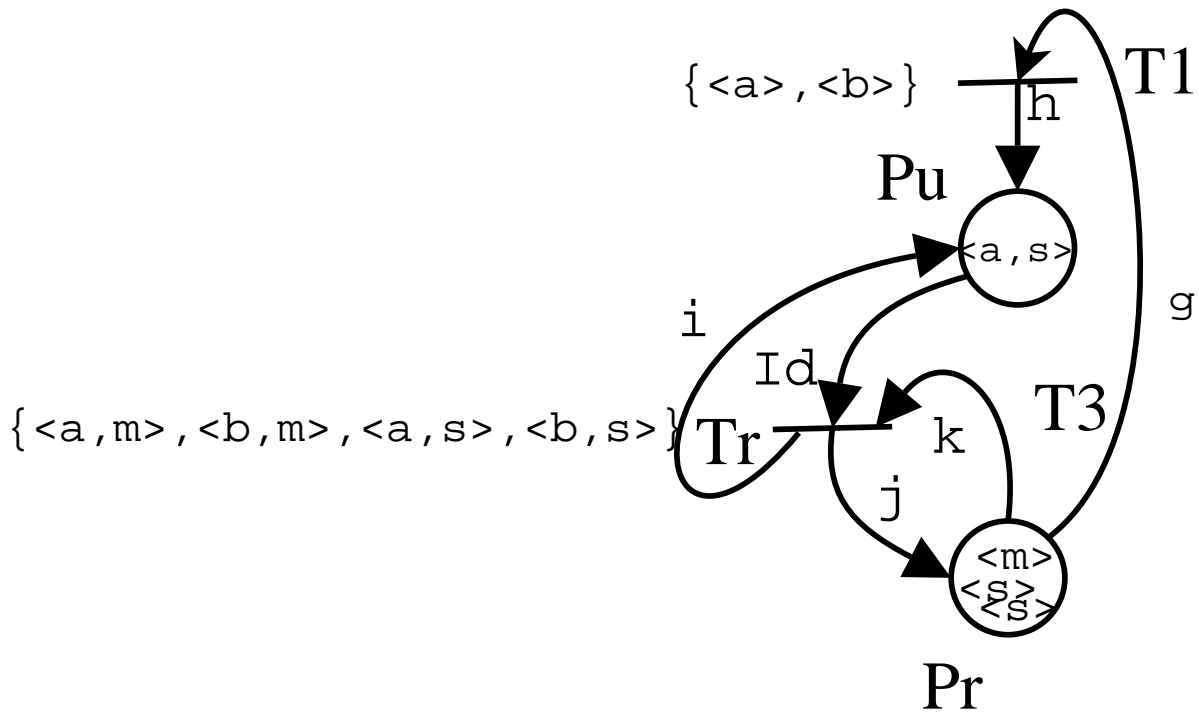


FIG. 7.4 – Machine avec son stock de sortie produisant a et b, RdP coloré suite

- Les deux transitions T_2 et T_3 se plient en la transition T_r . Le franchissement de T_r correspond au départ d'une pièce a ou b de la machine ou du stock.
- Comme précédemment, le franchissement de T_1 peut se faire pour la couleur $\langle a \rangle$ ou $\langle b \rangle$. Dans les deux cas, une marque $\langle m \rangle$ est enlevée de la place P_r . On associe à l'arc reliant P_r à T_1 la fonction g définie par :

Après franchissement de T_1 pour la couleur $\langle a \rangle$ (resp. $\langle b \rangle$), on ajoute dans la place P_u une marque $\langle a, m \rangle$ (resp. $\langle b, m \rangle$). On associe à l'arc reliant T_1 à P_u la fonction h définie par :

- Le franchissement de T_r peut se faire pour les couleurs complexes $\langle a, m \rangle$, $\langle b, m \rangle$ (départ d'une pièce a, b de la machine), $\langle a, s \rangle$, $\langle b, s \rangle$ (départ d'une pièce a, b du stock). On associe à l'arc reliant P_u à T_r la fonction Id. Dans le cas du départ d'une pièce a ou b de la machine vers le stock, un emplacement du stock doit être disponible. La place P_r est donc en entrée de la transition T_r avec la fonction associée k définie par :

Le franchissement de T_r pour les couleurs complexes $\langle a, m \rangle$ et $\langle b, m \rangle$ introduit dans la place P_u les marques $\langle a, s \rangle$ et $\langle b, s \rangle$ et dans P_r une marque $\langle m \rangle$. Le franchissement de T_r pour les couleurs complexes $\langle a, s \rangle$ et $\langle b, s \rangle$ introduit dans la place P_r

une marque $\langle s \rangle$. On associe donc à l'arc reliant T_r à P_u la fonction i définie par :

et à l'arc reliant T_r à P_r la fonction j définie par :

On obtient alors le RdP coloré représenté Figure 7.4.

Remarque A partir d'un même RdP, on a obtenu deux RdPs colorés, le second comportant moins de places et de transitions que le premier. Pour un RdP, le RdP coloré obtenu n'est donc pas unique. De plus, on peut se poser la question de la plus petite taille d'un RdP coloré équivalent à un RdP. Il est possible de ramener un RdP à un RdP coloré comprenant une seule place et une seule transition, ce qui est certes très compacte mais peut être difficilement lisible et interprétable. Ces deux critères permettent de choisir le RdP coloré le plus adapté pour un RdP donné.

7.2 Réseaux de Petri colorés (simplifiés)

7.2.1 Définition

Definition 7.2.1 Une *couleur* C_k est un n -uplet $C_k = \langle c_{k_1}, \dots, c_{k_q} \rangle$.

Definition 7.2.2 Un **RdP coloré** Q_c est un sextuplet $Q_c = \langle P, T, C, C_{sec}, Pre, Post \rangle$ où

- $P = \{P_1, \dots, P_n\}$ est un ensemble fini non vide de places ;
- $T = \{T_1, \dots, T_m\}$ est un ensemble fini non vide de transitions ;
- $P \cap T = \emptyset$;
- $C = \{c_1, \dots, c_r\}$ est un ensemble fini non vide de couleurs ;
- $C_{sec} : P \cup T \rightarrow \mathcal{P}(C)$ est la fonction sous-ensemble de couleurs qui à chaque place et à chaque transition associe un sous ensemble de C ;
- Pre est l'application définie sur $P \times T$ qui à tout couple (P_i, T_j) associent une application définie de $C_{sec}(T_j)$ dans l'ensemble des applications de $C_{sec}(P_i) \rightarrow \mathbb{N}$.
- $Post$ est l'application définie sur $P \times T$ qui à tout couple (P_i, T_j) associent une application définie de $C_{sec}(T_j)$ dans l'ensemble des applications de $C_{sec}(P_i) \rightarrow \mathbb{N}$.

Un **RdP coloré marqué** R_c est un RdP coloré Q_c associé à un marquage initial $M_0 : R_c = \langle Q_c, M_0 \rangle$. Pour une place $P_i \in P$, $M_0(P_i)$ est défini comme une application de $C_{sec}(P_i) \rightarrow \mathbb{N}$.

Exemple Reprenons le RdP coloré marqué Figure 7.4.

- $P = \{P_u, P_r\}$;
- $T = \{T_1, T_r\}$;
- $C = \{\langle a, m \rangle, \langle b, m \rangle, \langle a, s \rangle, \langle b, s \rangle, \langle s \rangle, \langle m \rangle\}$;
- C_{sec} est définie par :

$$\begin{aligned} C_{sec}(P_r) &= \{\langle s \rangle, \langle m \rangle\} \\ C_{sec}(P_u) &= \{\langle a, m \rangle, \langle b, m \rangle, \langle a, s \rangle, \langle b, s \rangle\} \\ C_{sec}(T_r) &= \{\langle a, m \rangle, \langle b, m \rangle, \langle a, s \rangle, \langle b, s \rangle\} \\ C_{sec}(T_1) &= \{\langle a \rangle, \langle b \rangle\} \end{aligned}$$

- Pre est définie par

– $Pre(P_r, T_1) = g$, soit :

$$\begin{aligned} Pre(P_r, T_1)(\langle a \rangle)(\langle s \rangle) &= 0 & Pre(P_r, T_1)(\langle a \rangle)(\langle m \rangle) &= 1 \\ Pre(P_r, T_1)(\langle b \rangle)(\langle s \rangle) &= 0 & Pre(P_r, T_1)(\langle b \rangle)(\langle m \rangle) &= 1 \end{aligned}$$

– $Pre(P_u, T_1) = 0$;

– $Pre(P_r, T_r) = k$ soit :

$$\begin{aligned} Pre(P_r, T_r)(\langle a, m \rangle)(\langle s \rangle) &= 1 & Pre(P_r, T_r)(\langle a, m \rangle)(\langle m \rangle) &= 0 \\ Pre(P_r, T_r)(\langle b, m \rangle)(\langle s \rangle) &= 1 & Pre(P_r, T_r)(\langle b, m \rangle)(\langle m \rangle) &= 0 \\ Pre(P_r, T_r)(\langle a, s \rangle)(\langle s \rangle) &= 0 & Pre(P_r, T_r)(\langle a, s \rangle)(\langle m \rangle) &= 0 \\ Pre(P_r, T_r)(\langle b, s \rangle)(\langle s \rangle) &= 0 & Pre(P_r, T_r)(\langle b, s \rangle)(\langle m \rangle) &= 0 \end{aligned}$$

– $Pre(P_u, T_r) = Id$;

– $Post$ est définie par :

– $Post(P_r, T_1) = 0$;

– $Post(P_u, T_1) = h$ soit :

$$\begin{aligned} Post(P_u, T_1)(\langle a \rangle)(\langle a, m \rangle) &= 1 & Post(P_u, T_1)(\langle a \rangle)(\langle b, m \rangle) &= 0 \\ Post(P_u, T_1)(\langle a \rangle)(\langle a, s \rangle) &= 0 & Post(P_u, T_1)(\langle a \rangle)(\langle b, s \rangle) &= 0 \\ Post(P_u, T_1)(\langle b \rangle)(\langle a, m \rangle) &= 0 & Post(P_u, T_1)(\langle b \rangle)(\langle b, m \rangle) &= 1 \\ Post(P_u, T_1)(\langle b \rangle)(\langle a, s \rangle) &= 0 & Post(P_u, T_1)(\langle b \rangle)(\langle b, s \rangle) &= 0 \end{aligned}$$

– $Post(P_r, T_r) = j$ soit :

$$\begin{aligned} Post(P_r, T_r)(\langle a, m \rangle)(\langle s \rangle) &= 0 & Post(P_r, T_r)(\langle a, m \rangle)(\langle m \rangle) &= 1 \\ Post(P_r, T_r)(\langle b, m \rangle)(\langle s \rangle) &= 0 & Post(P_r, T_r)(\langle b, m \rangle)(\langle m \rangle) &= 1 \\ Post(P_r, T_r)(\langle a, s \rangle)(\langle s \rangle) &= 1 & Post(P_r, T_r)(\langle a, s \rangle)(\langle m \rangle) &= 0 \\ Post(P_r, T_r)(\langle b, s \rangle)(\langle s \rangle) &= 1 & Post(P_r, T_r)(\langle b, s \rangle)(\langle m \rangle) &= 0 \end{aligned}$$

– $Post(P_u, T_r) = i$ soit :

$$\begin{aligned} Post(P_u, T_1)(\langle a, m \rangle)(\langle a, m \rangle) &= 0 & Post(P_u, T_1)(\langle a, m \rangle)(\langle a, s \rangle) &= 1 \\ Post(P_u, T_1)(\langle a, m \rangle)(\langle b, m \rangle) &= 0 & Post(P_u, T_1)(\langle a, m \rangle)(\langle b, s \rangle) &= 0 \\ Post(P_u, T_1)(\langle b, m \rangle)(\langle a, m \rangle) &= 0 & Post(P_u, T_1)(\langle b, m \rangle)(\langle a, s \rangle) &= 0 \\ Post(P_u, T_1)(\langle b, m \rangle)(\langle b, m \rangle) &= 0 & Post(P_u, T_1)(\langle b, m \rangle)(\langle b, s \rangle) &= 1 \\ Post(P_u, T_1)(\langle a, s \rangle)(\langle a, m \rangle) &= 0 & Post(P_u, T_1)(\langle a, s \rangle)(\langle a, s \rangle) &= 0 \\ Post(P_u, T_1)(\langle a, s \rangle)(\langle b, m \rangle) &= 0 & Post(P_u, T_1)(\langle a, s \rangle)(\langle b, s \rangle) &= 0 \\ Post(P_u, T_1)(\langle b, s \rangle)(\langle a, m \rangle) &= 0 & Post(P_u, T_1)(\langle b, s \rangle)(\langle a, s \rangle) &= 0 \\ Post(P_u, T_1)(\langle b, s \rangle)(\langle b, m \rangle) &= 0 & Post(P_u, T_1)(\langle b, s \rangle)(\langle b, s \rangle) &= 0 \end{aligned}$$

– Le marque initial M_0 est défini par :

$$\begin{aligned} M_0(P_r)(\langle m \rangle) &= 1 & M_0(P_r)(\langle s \rangle) &= 2 \\ M_0(P_u)(\langle a, m \rangle) &= 0 & M_0(P_u)(\langle b, m \rangle) &= 0 \\ M_0(P_u)(\langle a, s \rangle) &= 1 & M_0(P_u)(\langle b, s \rangle) &= 0 \end{aligned}$$

7.2.2 Evolution du marquage d'un RdP coloré

Une transition T_j est dite **validée pour la couleur** c_{k_l} si

$$\forall P_i \in {}^oT_j, \forall c_k \in C_{sec}(P_i), M(P_i)(c_k) \geq Pre(P_i, T_j)(c_{k_l})(c_k).$$

Exemple Sur le RdP coloré marqué Figure 7.4, la transition T_1 est validée pour les couleurs $\langle a \rangle$ et $\langle b \rangle$. La transition T_r est validée pour la couleur $\langle a, s \rangle$ mais n'est pas validée pour les couleurs $\langle a, m \rangle$, $\langle b, m \rangle$, $\langle b, s \rangle$.

Si la transition est validée pour la couleur c_{k_l} , le **franchissement de T_j pour la couleur c_{k_l}** consiste à ôter $Pre(P_i, T_j)(c_{k_l})(c_k)$ marques dans les places P_i en entrée de la transition T_j et à mettre $Post(P_i, T_j)(c_{k_l})(c_k)$ marques dans les places P_i en sortie de la transition T_j : le nouveau marquage obtenu M' est tel que :

$$\forall P_i \in P, \forall c_k \in C_{sec}(P_i), M'(P_i)(c_k) = M(P_i)(c_k) + Post(P_i, T_j)(c_{k_l})(c_k) - Pre(P_i, T_j)(c_{k_l})(c_k).$$

7.2.3 Analyse d'un RdP coloré marqué

L'analyse des RdPs colorés est basée sur la propriété suivante.

Propriété 7.2.1 *Tout RdP coloré marqué peut être déplié (c'est-à-dire transformé de façon équivalente) en un RdP marqué.*

L'analyse peut donc être entreprise en dépliant le RdP coloré en un RdP équivalent. Une autre approche est de généraliser les méthodes d'analyse étudiées pour les RdPs dans les chapitres 4, 5 et 6. L'inconvénient de la première approche est qu'elle peut déboucher sur l'analyse de RdP de grande taille ; l'inconvénient de la seconde est que la généralisation des méthodes d'analyse aux RdPs colorés est assez lourde. Vu les objectifs du cours, l'analyse des RdPs colorés se fera par la première approche.

7.3 Un exemple d'application à la modélisation de systèmes de production

L'utilisation des RdPs colorés est particulièrement adaptée à la modélisation de systèmes de production. Cela va être illustré par deux exemples.

7.3.1 Modélisation d'un convoyeur de type FIFO

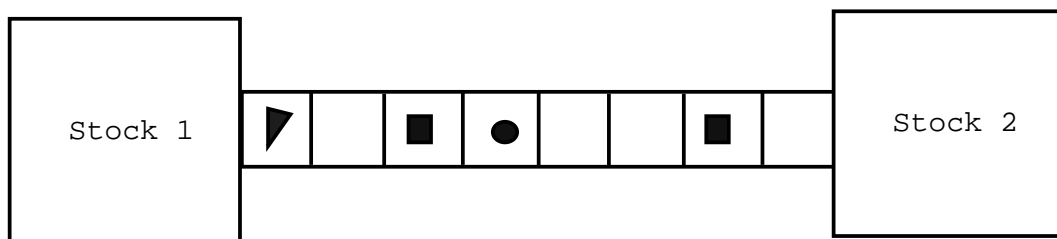


FIG. 7.5 – Convoyeur de type FIFO

Un convoyeur amène des objets d'un stock 1 à un stock 2. Les objets, en nombre quelconque, sont de m types différents. Le convoyeur est de type FIFO, c'est-à-dire First In, First Out (premier entré, premier sorti). Il peut être décomposé en n secteurs. Un secteur ne peut contenir qu'un seul objet à la fois (voir Figure 7.5). L'opération de convoyage est modélisée par le passage d'un secteur au secteur suivant.

L'état du système est caractérisé par les listes des :

- secteurs libres (place P_l) ;
- objets dans le stock 1 avec l'indication de leur type (place P_1) ;
- objets dans le stock 2 avec l'indication de leur type (place P_2) ;
- secteurs occupés par un objet avec l'indication du type de l'objet (place P_o).

Une couleur est associée à chaque secteur et à chaque type d'objet :

- pour le secteur numéro i , couleur $\langle s_i \rangle$;
- pour le j^{ieme} type d'objet, couleur $\langle o_j \rangle$.

Par suite,

- la place P_l va contenir des marques de couleurs simples : celles correspondant aux secteurs libres, soient les couleurs $\langle s_i \rangle$;
- les places P_1 et P_2 vont contenir des marques de couleurs simples : celles correspondant aux types d'objets qui sont stockés, soient les couleurs $\langle o_j \rangle$;
- la place P_o va contenir des marques de couleurs complexes car elles doivent contenir deux informations : secteur occupé et type de l'objet occupant ce secteur, soit $\langle s_i, o_j \rangle$.

L'état du système va évoluer quand se produit :

- la sortie d'un objet de type i du stock 1 pour occuper le secteur $\langle s_1 \rangle$ (transition T_1) ;
- la sortie d'un objet de type i du convoyeur via le dernier secteur (secteur n) vers le stock 2 ; le secteur n est ainsi libéré (transition T_2) ;
- le passage d'un objet de type j du secteur i vers le secteur $i + 1$; ce passage entraîne la libération du secteur i et l'occupation du secteur $i + 1$ (transition T_a).

Par suite :

- Le franchissement de la transition T_1 va dépendre des types d'objets stockés dans le stock 1 ainsi que de l'occupation du secteur 1 : il est donc naturel d'associer à la transition T_1 l'ensemble des couleurs complexes suivant :

$$A_1 = \{ \langle s_1, o_1 \rangle, \langle s_1, o_2 \rangle, \dots, \langle s_1, o_m \rangle \}.$$

T_1 aura en entrée les places P_1 et P_l .

- Pour qu'un objet de couleur $\langle o_j \rangle$ sorte du stock 1, celui-ci doit en contenir au moins un (au moins une marque de couleur $\langle o_j \rangle$ dans P_1) : pour l'arc reliant P_1 à T_1 , la fonction associée est donc $proj_1$ définie par

$$proj_1(\langle s_1, o_j \rangle) = \langle o_j \rangle .$$

- Pour qu'un objet occupe le secteur 1, celui-ci doit être libre (une marque de couleur $\langle s_1 \rangle$ dans P_l) : pour l'arc reliant P_l à T_1 , la fonction associée est donc $proj_2$ définie par

$$proj_2(\langle s_1, o_j \rangle) = \langle s_1 \rangle .$$

Le franchissement de la transition T_1 pour la couleur $\langle s_1, o_j \rangle$ correspond à mettre un objet de couleur $\langle o_j \rangle$ dans le secteur 1. Son franchissement pour la couleur $\langle s_1, o_j \rangle$ ajoute une marque de même couleur dans la place P_o . T_1 aura donc en sortie la place P_o , avec la fonction Id.

- Le franchissement de la transition T_2 va dépendre de l'occupation du secteur n ainsi que, s'il est occupé, du type de l'objet présent dans ce secteur : il est donc naturel d'associer à la transition T_2 l'ensemble des couleurs complexes suivant :

$$A_2 = \{ \langle s_n, o_1 \rangle, \langle s_n, o_2 \rangle, \dots, \langle s_n, o_m \rangle \}$$

T_2 aura en entrée la place P_o , avec la fonction Id et en sortie les places P_l et P_2 :

- le franchissement de T_2 pour la couleur $\langle s_n, o_j \rangle$ libère le secteur n , ce qui revient à rajouter la marque de couleur $\langle s_n \rangle$ dans la place P_l : pour l'arc reliant T_2 à P_l , la fonction associée est donc $proj_2$;

- le franchissement de T_2 pour la couleur $\langle \mathbf{s}_n, \mathbf{o}_j \rangle$ correspond à entreposer un objet de couleur $\langle \mathbf{o}_j \rangle$ dans le stock 2, soit l'ajout d'une marque de couleur $\langle \mathbf{o}_j \rangle$ dans le stock 2 : pour l'arc reliant T_2 à P_2 , la fonction associée est donc $proj_1$.
- Le franchissement de la transition T_a va dépendre de l'occupation des secteur i et $i + 1$ ainsi que, si le secteur i est occupé, du type j de l'objet présent dans ce secteur : il est donc naturel d'associer à la transition T_a l'ensemble des couleurs complexes définis par :

$$A_a = \{ \langle \mathbf{s}_i, \mathbf{o}_j \rangle, \quad i \in \{1, \dots, n-1\}, \quad j \in \{1, \dots, m\}, \}$$

T_a a pour places d'entrée les places P_l et P_o .

- Pour qu'il y ait passage du secteur i au secteur $i + 1$, le secteur $i + 1$ doit être libre, c'est-à-dire, P_l contient une marque $\langle \mathbf{s}_{i+1} \rangle$. Pour l'arc reliant P_l à T_2 , la fonction associée est donc $proj_2 \circ succ_1$ avec $succ_1$ définie par :

$$succ_1(\langle \mathbf{s}_i, \mathbf{o}_j \rangle) = \langle \mathbf{s}_{i+1}, \mathbf{o}_j \rangle .$$

- Pour qu'il y ait passage du secteur i au secteur $i + 1$ d'un objet de type $\langle \mathbf{o}_j \rangle$, le secteur i doit en contenir un. Pour l'arc reliant P_o à T_a , la fonction associée est donc Id.

T_a a aussi pour places de sortie les places P_l et P_o .

- Le passage du secteur i au secteur $i + 1$ rend occupé le secteur $i + 1$, soit l'ajout de la marque $\langle \mathbf{s}_{i+1}, \mathbf{o}_j \rangle$ dans la place P_o . Pour l'arc reliant T_a à P_o , la fonction associée est $succ_1$ définie par :

$$succ_1(\langle \mathbf{s}_i, \mathbf{o}_j \rangle) = \langle \mathbf{s}_{i+1}, \mathbf{o}_j \rangle .$$

- Le passage du secteur i au secteur $i + 1$ libère le secteur i , soit l'ajout de la marque $\langle \mathbf{s}_i \rangle$ dans la place P_l . Pour l'arc reliant T_a à P_l , la fonction associée est donc $proj_2$.

Le RdP coloré finalement obtenu est représenté Figure 7.6.

7.3.2 Modélisation d'une machine sur un convoyeur FIFO

On considère que le convoyeur présenté précédemment dessert une Machine située au niveau des sections 5 et 6 (voir Figure 7.7). Cette Machine va effectuer une opération seulement sur les objets de couleur $\langle \mathbf{o}_1 \rangle$ et $\langle \mathbf{o}_2 \rangle$. La Machine ne peut effectuer qu'une seule opération à la fois. Quand un objet de ce type arrive dans la section 5, cette pièce est chargée sur la Machine, si celle-ci est libre. Dans le cas contraire, l'objet reste dans le secteur 5 jusqu'à ce que l'objet soit chargé dans la Machine. Une fois l'opération effectuée, l'objet est remis sur le convoyeur mais dans la section 6.

Pour obtenir le modèle de ce système, on va compléter le modèle du convoyeur introduit dans la section précédente. Par contre, il est nécessaire d'introduire deux places supplémentaires pour :

- indiquer si la Machine est disponible (une marque dans la place P_d) ;
- si la Machine est occupée, sur quel type d'objets la Machine effectue une opération : comme il y a deux types possibles, dans cette place P_{op} , les marques seront de couleurs $\langle \mathbf{o}_1 \rangle$ ou $\langle \mathbf{o}_2 \rangle$

et deux transitions supplémentaires pour indiquer quand

- une opération commence sur la Machine : comme on ne peut faire d'opération que sur les pièces de couleurs $\langle \mathbf{o}_1 \rangle$ et $\langle \mathbf{o}_2 \rangle$ qui se trouve dans le secteur 5, seules deux couleurs seront associées à cette transition T_d :

$$A_d = \{ \langle \mathbf{s}_5, \mathbf{o}_1 \rangle, \langle \mathbf{s}_5, \mathbf{o}_2 \rangle \}.$$

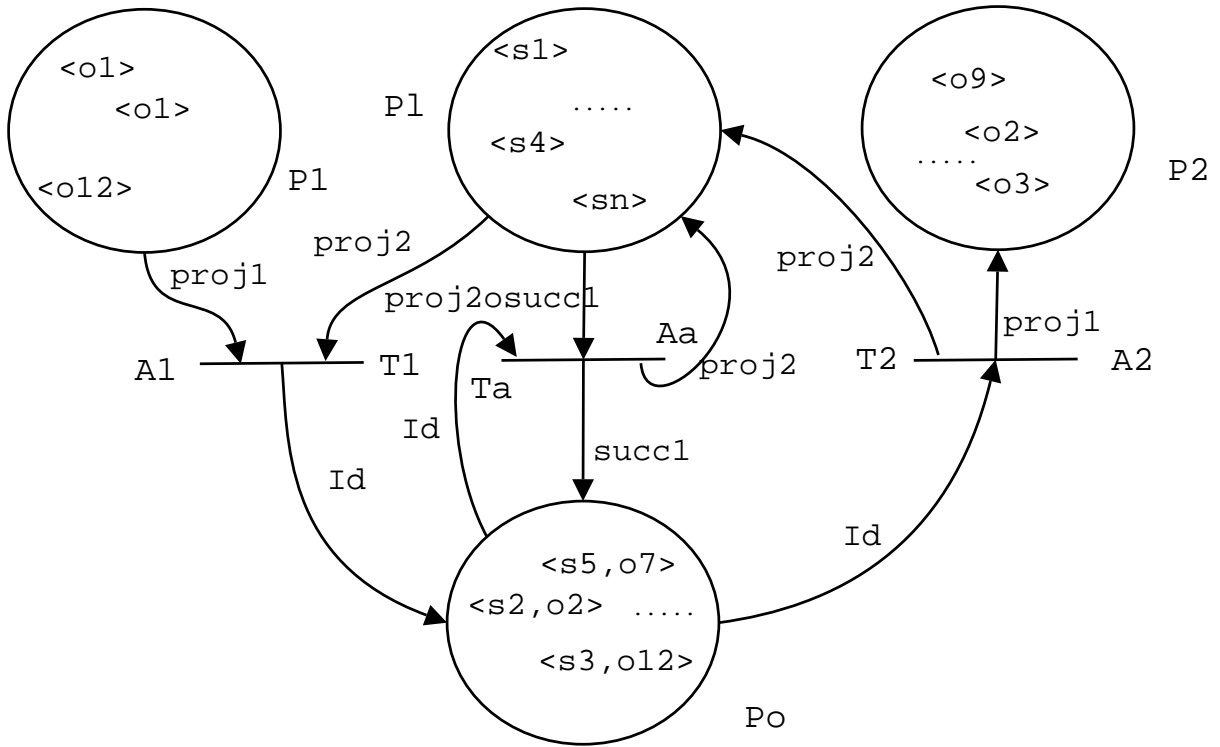


FIG. 7.6 – Rdp coloré du convoyeur de type FIFO

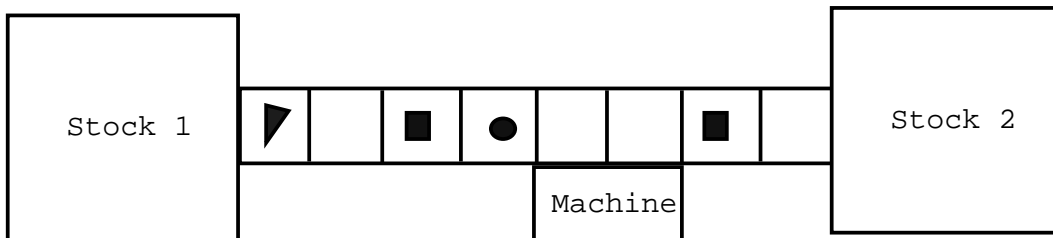


FIG. 7.7 – Convoyeur de type FIFO

L'autre condition pour franchir T_d est que la Machine soit disponible. Un arc relie donc P_o à T_d avec pour fonction Id et un arc relie P_d à T_d avec pour fonction Dec la fonction de décoloration définie par :

$$Dec(\langle s_5, o_1 \rangle) = \langle \bullet \rangle .$$

Le franchissement correspond à charger la pièce sur la Machine libérant ainsi le secteur 5. Par suite, une marque de couleur qui indique le type de la pièce chargée est ajoutée à la place P_{op} et une marque de couleurs $\langle s_5 \rangle$ est ajoutée à la place P_l . La transition T_d a donc pour places de sortie P_{op} avec pour fonction associée à l'arc la fonction $proj_1$ et P_l avec pour fonction associée à l'arc la fonction $proj_2$.

- la pièce quitte la Machine : comme cela ne peut se faire que s'il y a une pièce de couleurs $\langle o_1 \rangle$ ou $\langle o_2 \rangle$ sur la Machine et que si le secteur 6 est libre, on associe à cette transition T_f l'ensemble de couleurs :

$$A_f = \{ \langle s_6, o_1 \rangle, \langle s_6, o_2 \rangle \}.$$

La transition T_f a donc en entrée les places P_l avec la fonction associée $proj_2$ et P_{op} avec la fonction associée $proj_1$. Son franchissement mène à l'occupation du secteur 6 et à la disponibilité de la Machine, soit une marque de la couleur pour laquelle la transition a été franchie dans la place P_o (fonction associée Id) et une marque non colorée dans la place P_d (fonction associée Dec).

Le RdP coloré sera obtenu à partir du RdP coloré modélisant le système de convoyeur en notant que cette fois-ci, si le secteur 5 contient une pièce de type $\langle o_1 \rangle$ ou de type $\langle o_2 \rangle$, elle ne peut pas passer sur le secteur 6. La transition T_a ne pourra donc être franchie que pour l'ensemble de couleur A_a privé des couleurs $\langle s_5, o_1 \rangle$ et $\langle s_5, o_2 \rangle$:

$$A'_a = A_a \setminus \{ \langle s_5, o_1 \rangle, \langle s_5, o_2 \rangle \}.$$

Le RdP coloré finalement obtenu est représenté Figure 7.8.

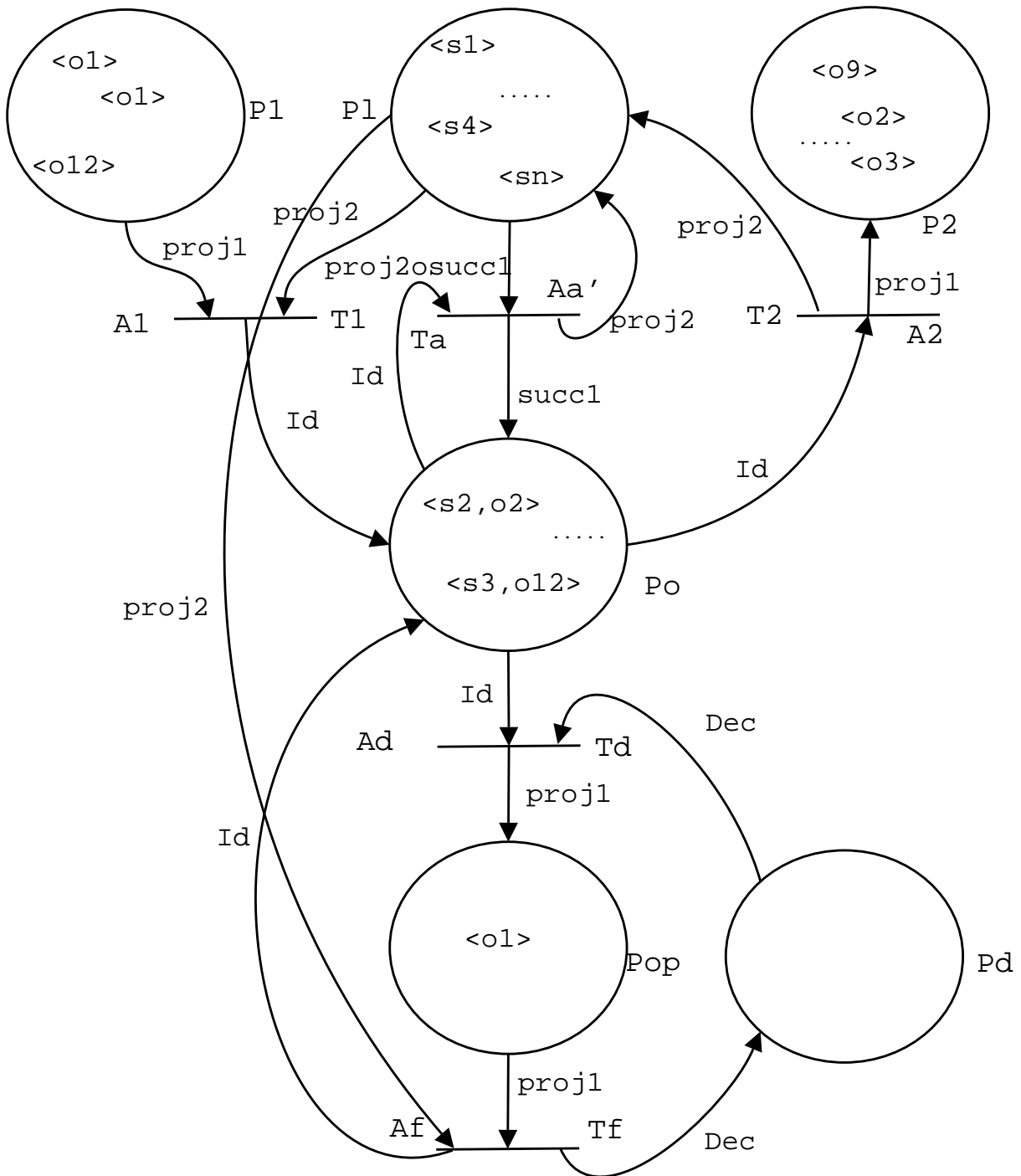


FIG. 7.8 – Rdp coloré de la machine avec convoyeur de type FIFO

Chapitre 8

Réseaux de Petri non autonomes

L'évolution du RdP va maintenant dépendre de l'occurrence d'événements externes ou du temps. Par simplicité, on se concentrera sur l'extension des RdPs ordinaires, même si cela peut s'appliquer aux RdPs généralisés.

8.1 Réseaux de Petri synchronisés (RdPS)

Dans les modélisations RdPs que nous avons vues dans les chapitres précédents, le fait qu'une transition soit franchissable indique que toutes les conditions sont réunies pour qu'elle soit effectivement franchie. Le moment où se produira le franchissement n'est pas connu. Un RdP synchronisé est un RdP où à chaque transition est associée un événement. La transition sera alors franchie si elle est validée mais quand l'événement associé se produit.

8.1.1 Événements associés à une variable logique

Un **événement** est un front montant ou un front descendant d'une variable logique. Un événement n'a pas de durée : il est caractérisé par l'instant où il se produit (voir figure 8.1).

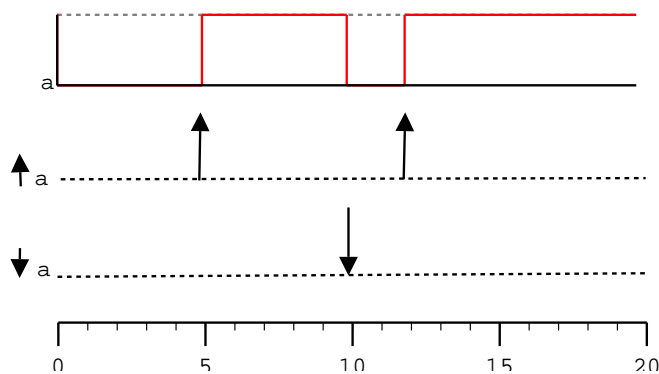


FIG. 8.1 – Chronogramme de la variable logique a

Notation : $\uparrow a$ (front montant) $\downarrow a$ (front descendant)

– $\uparrow a.b = (\uparrow a).b$ est un événement qui se produit en même temps que $\uparrow a$ si $b = 1$; ici le $.$ désigne “et” ;

– $\uparrow a + b = (\uparrow a) + b$: ici le $+$ désigne “ou” ;

– $\uparrow a. \uparrow b$ est un événement qui se produit quand $\uparrow a$ et $\uparrow b$ se produisent simultanément.

Cela n'est possible que si a et b ne sont pas indépendants, du fait de l'hypothèse suivante.

Hypothèse Deux événements indépendants ne se produisent jamais simultanément : $\uparrow a. \uparrow b = 0$ si a et b sont indépendants.

Propriétés Avec e l'événement toujours occurrent :

$\uparrow a = \downarrow \bar{a}$	$\uparrow a.e = \uparrow a$	$\uparrow a. \uparrow a = \uparrow a$	$\uparrow a. \uparrow \bar{a} = 0$
$\uparrow a.a = \uparrow a$	$\uparrow a \uparrow \bar{a} = 0$	$\downarrow a.\bar{a} = \downarrow a$	$\downarrow a.a = 0$

où \bar{a} désigne "non a ". Avec a, b et c indépendants :

$\uparrow (a.b) = \uparrow a.b + \uparrow b.a$	$\uparrow (a + b) = \uparrow a.\bar{b} + \uparrow b.\bar{a}$	$\uparrow (a.b) \uparrow (a.c) = \uparrow a.b.c$
--	--	--

Pour un système donné, un événement est dit **externe** s'il correspond à un changement du monde extérieur au système.

8.1.2 Définition d'un Réseau de Petri Synchronisé

Exemple : atelier de coupe Sur le modèle RdP de l'atelier de coupe, il est naturel d'associer le franchissement de la transition T_1 à l'arrivée d'une nouvelle commande (voir Figure 8.2). Celle-ci correspond à un événement externe noté E_1

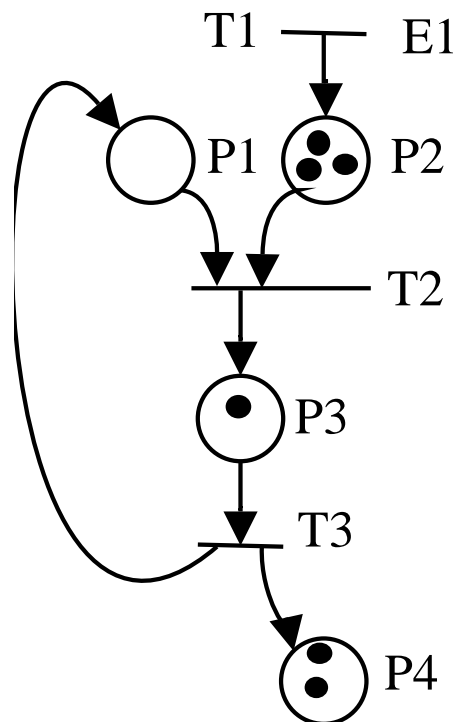


FIG. 8.2 – Atelier de coupe

Définition 8.1.1 On appelle **Réseau de Petri Synchronisé** le triplet $\langle R, E, Sync \rangle$ où

- R est un Réseau de Petri marqué ;
- E est un ensemble d'événements externes ;
- $Sync : T \rightarrow E \cup \{e\}$ où
 - T est l'ensemble des transitions de R ;

– e est l'événement toujours *occurent*¹.

Le RdP est **totalelement Synchronisé** si aucune transition n'est associée à e .

Dans un RdP synchronisé, une transition validée n'est pas forcément franchissable. La transition est validée quand la condition sur les marquages est satisfaite. Elle deviendra franchissable quand l'événement externe associé à la transition se produit : elle est alors immédiatement franchie. Si en fonction du marquage de ses places d'entrée, plusieurs franchissements sont possibles, un seul se produira effectivement, celui dont l'événement associé se produit en premier.

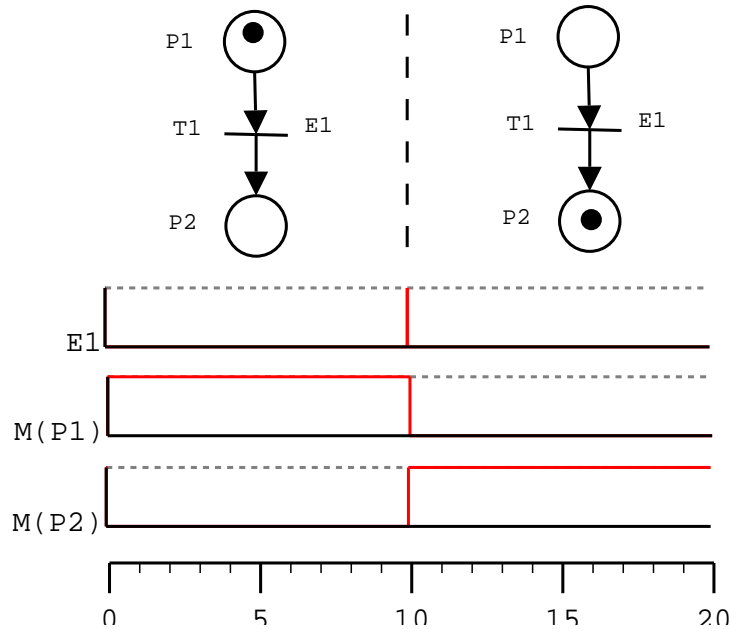


FIG. 8.3 – Franchissement d'une transition synchronisée

Exemples de franchissements de transition Sur le RdP synchronisé représenté figure 8.3, au temps $t = 0$, la transition T_1 est validée. Le franchissement de la transition T_1 s'effectue au temps $t = 10$ quand l'événement E_1 se produit.

Sur le RdP synchronisé représenté figure 8.4, au temps $t = 0$, la transition T_1 est validée. Son marquage est suffisant pour pouvoir franchir T_1 deux fois. Le franchissement de la transition T_1 s'effectue au temps $t = 10$ quand l'événement E_1 se produit. On impose qu'un seul franchissement de cette transition à la fois.

Sur le RdP synchronisé représenté figure 8.5, au temps $t = 0$, la transition T_1 est validée. Le franchissement de la transition T_1 s'effectue au temps $t = 10$ quand l'événement E_1 se produit. L'événement E_1 est aussi² associé à la transition T_2 . Comme la transition T_2 n'était pas validée au moment où l'événement E_1 se produit, T_2 n'est pas franchie.

Sur le RdP synchronisé représenté figure 8.6, au temps $t = 0$, la transition T_1 est validée. Le franchissement de la transition T_1 s'effectue au temps $t = 5$ quand l'événement E_1 se produit. La transition T_2 est franchie au moment de la seconde occurrence de l'événement E_1 soit au temps $t = 15$. Le marquage du RdP synchronisé évolue donc dans le temps de la façon suivante :

– Pour t allant de 0 à 5, $M_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$;

¹L'événement qui se produit en permanence.

²Il est possible d'associer un même événement à plusieurs transitions.

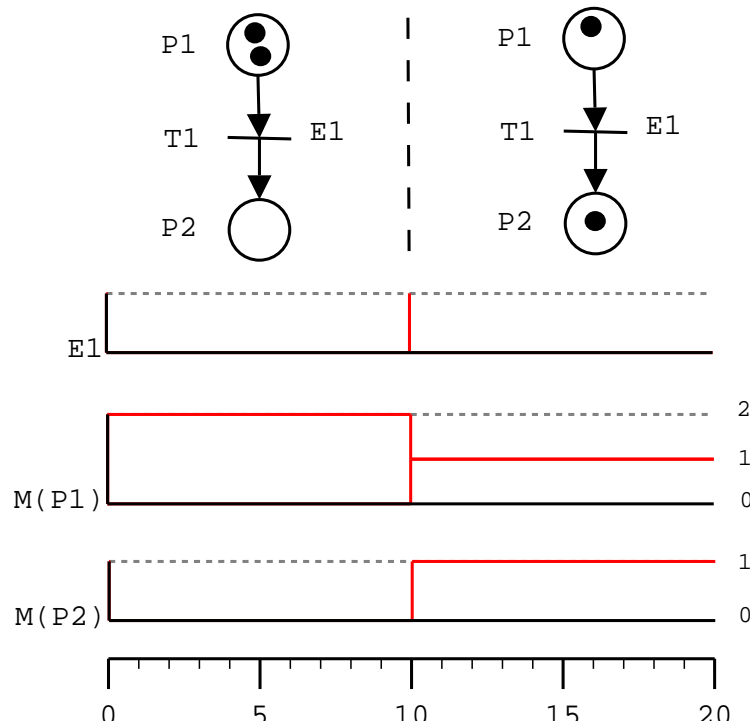


FIG. 8.4 – Franchissement d’une transition synchronisée

- Pour t allant de 5 à 15, $M_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$;
- Pour t allant de 15 à 20, $M_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$;

Le RdP synchronisé représenté figure 8.7 est similaire à celui figure 8.6 sauf que l’événement associé à la transition T_2 est e l’événement toujours occurrent. Après le franchissement de la transition T_1 , la transition T_2 est validée. Comme l’événement toujours occurrent y est associé, la transition T_2 est alors immédiatement franchie. Le marquage M_2 a donc une durée nulle. On parle de *marquage instable* (à opposer à *marquage stable*). On peut définir un graphe des marquages stables : dans celui-ci, on passe directement du marquage M_1 au marquage M_3 par franchissement de la séquence T_1T_2 : on parle d’un *franchissement itéré sur occurrence* de l’événement E_1 .

Il est possible de construire un modèle RdP synchronisé dans lequel un marquage stable peut ne pas être accessible par le franchissement d’un nombre fini de transitions, voir Figure 8.8. Ici, après occurrence de l’événement E_1 , il n’est pas possible d’obtenir un marquage stable. Dans la suite du chapitre, nous supposons que pour les RdPs synchronisés considérés, un marquage stable est forcément atteint après franchissement d’un nombre fini de transitions.

8.1.3 Propriétés des RdPs synchronisés

Les notions de bornitude et de vivacité peuvent se généraliser aux RdPs synchronisés. Un RdP synchronisé est borné si, pour tout marquage stable et instable, il est borné. Une transition T_j est vivante si pour tout marquage accessible, il existe une séquence d’événements externes telle que T_j soit franchie. Si on considère le RdP autonome associé au RdP synchronisé, peut-on

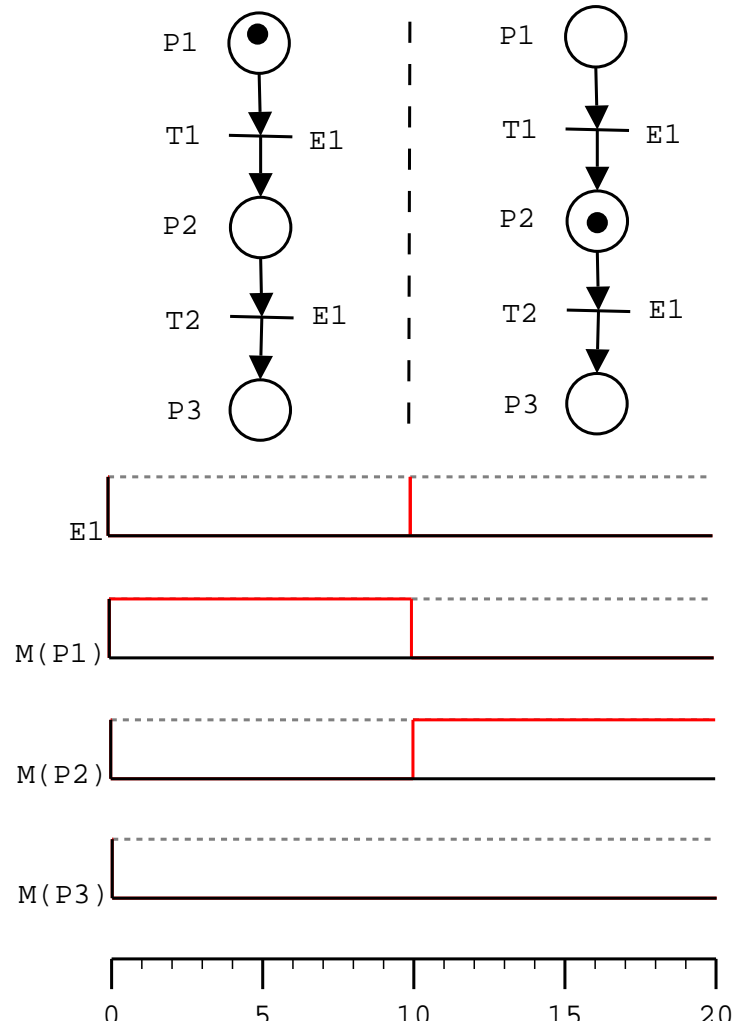


FIG. 8.5 – Franchissement d'une transition synchronisée

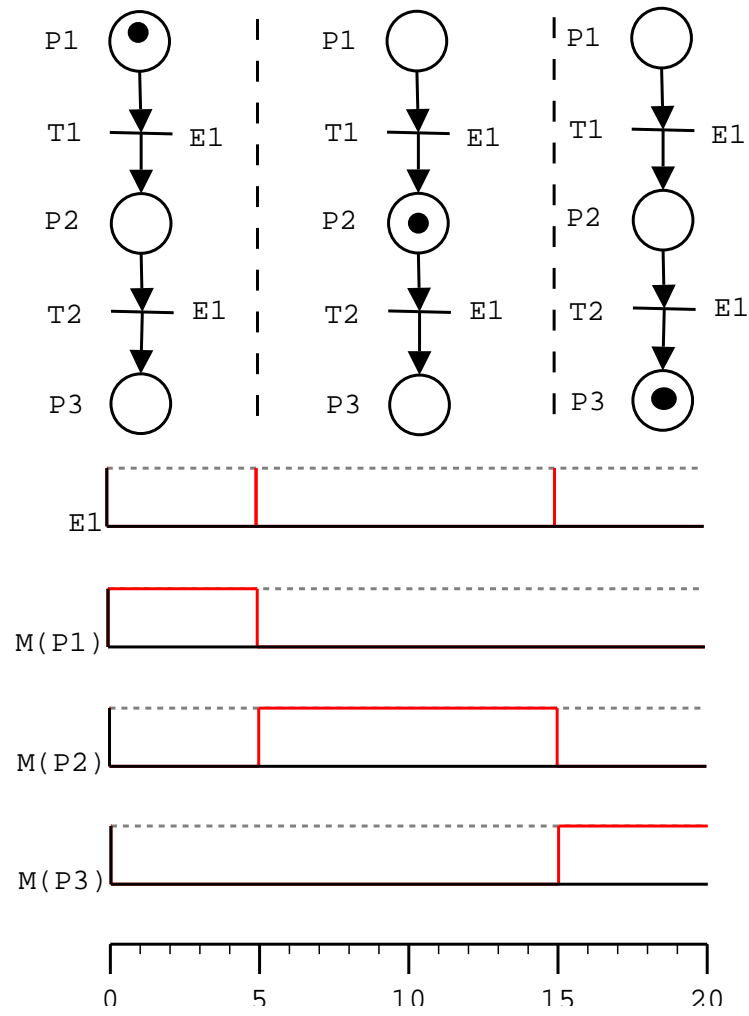


FIG. 8.6 – Franchissement d'une transition synchronisée

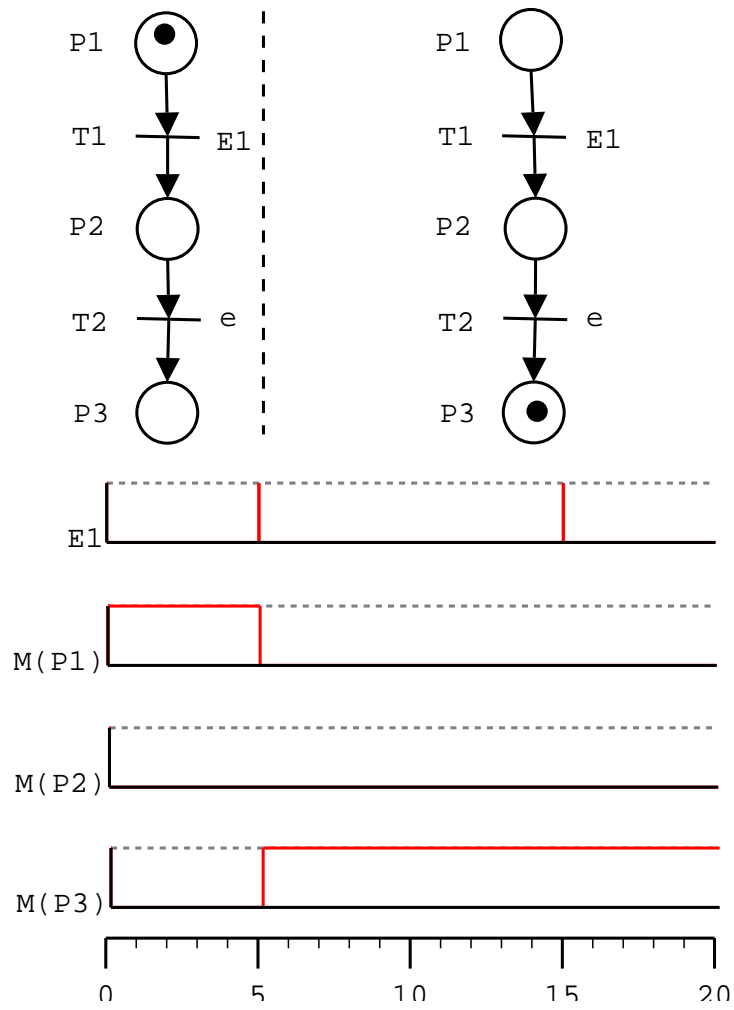


FIG. 8.7 – Franchissement d’une transition synchronisée

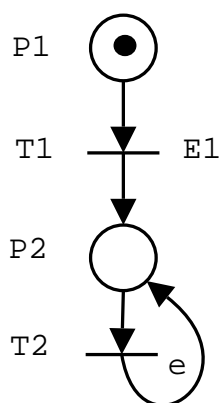


FIG. 8.8 – RdP synchronisé avec marquages instables

déduire les propriétés de l'un à partir de l'autre ?

Propriété 8.1.1

1. Un Rdp synchronisé est borné pour un marquage initial M_0 si le Rdp autonome associé est borné pour le marquage initial M_0 .
2. La réciproque n'est pas forcément vraie.

Démonstration Le 1. découle du fait que l'ensemble des marquages accessibles du Rdp synchronisé est inclus dans l'ensemble des marquages accessibles du Rdp autonome associé. Le 2.

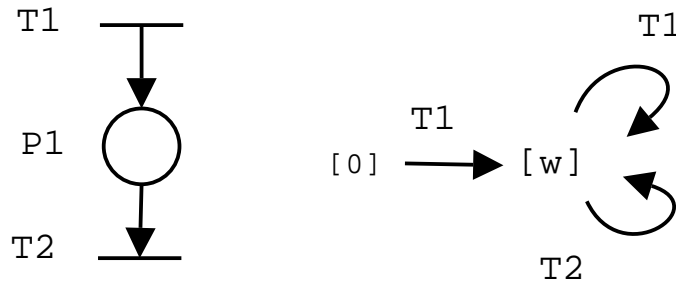


FIG. 8.9 – Rdp modélisant un stock avec son graphe de couverture

découle de l'exemple présenté Figure 8.9 et Figure 8.10. Le nombre de marques dans la place P_1 représente le nombre d'objets entreposés dans un stock. Le franchissement de la transition T_1 correspond à introduire un nouvel objet dans le stock. Le franchissement de la transition T_2 correspond à retirer un objet du stock. D'après le graphe de marquage Figure 8.9, ce Rdp autonome n'est pas borné. D'autre part, il est associé au Rdp synchronisé représenté Figure 8.10. Celui-ci est clairement borné puisque l'introduction d'un nouvel objet se produit sur occurrence de l'événement E_1 . Si la place P_1 a un nombre de marques non nul, la transition T_2 est validée : l'événement E_1 étant associée à T_2 , sur son occurrence, une marque est alors ôtée de la place P_1 . Le marquage de celle-ci est forcément borné. On a utilisé le fait que dans un Rdp synchronisé, deux transitions validées associées à un même événement sont simultanément franchies sur occurrence de cet événement. \square

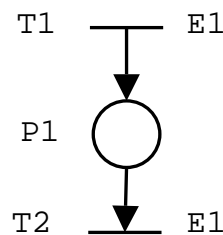


FIG. 8.10 – Rdp synchronisé modélisant un stock

Propriété 8.1.2 Pour qu'un Rdp synchronisé soit vivant pour un marquage initial M_0 , il est ni nécessaire, ni suffisant que le Rdp autonome associé soit vivant pour le marquage initial M_0 .

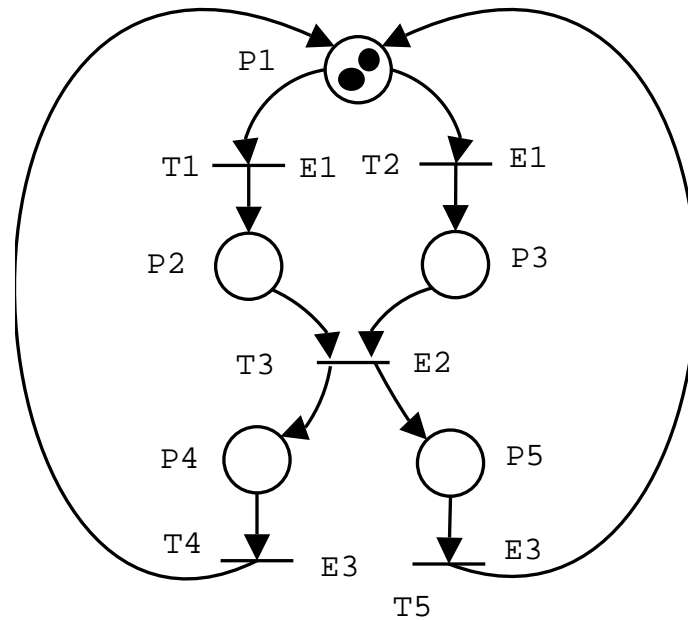


FIG. 8.11 – Vivacité d'un RdP (synchronisé)

Démonstration Si on construit le graphe des marquages du RdP autonome associé au RdP synchronisé représenté Figure 8.11, on constate que pour le marquage initial :

$$M_0 = \begin{bmatrix} 2 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

on peut aboutir aux blocages :

$$M_{b_1} = \begin{bmatrix} 0 \\ 2 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad \text{et} \quad M_{b_2} = \begin{bmatrix} 0 \\ 0 \\ 2 \\ 0 \\ 0 \end{bmatrix}.$$

Le RdP autonome n'est donc pas vivant. Pour le RdP synchronisé, partant du même marquage initial, par franchissement de $T_1 T_2$ sur occurrence de E_1 , on obtient :

$$M_1 = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

puis par franchissement de la transition T_3 sur occurrence de E_2 :

$$M_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}$$

et par franchissement de T_4T_5 sur occurrence de E_3 , on revient au marquage initial. Le RdP synchronisé est vivant car pour une transition donné, il existe toujours une séquencement d'événements externes qui mène au franchissement de cette transition.

Par contre, le RdP synchronisé représenté Figure 8.12 n'est pas vivant pour le marquage initial représenté sur cette figure alors que le RdP autonome associé l'est. \square

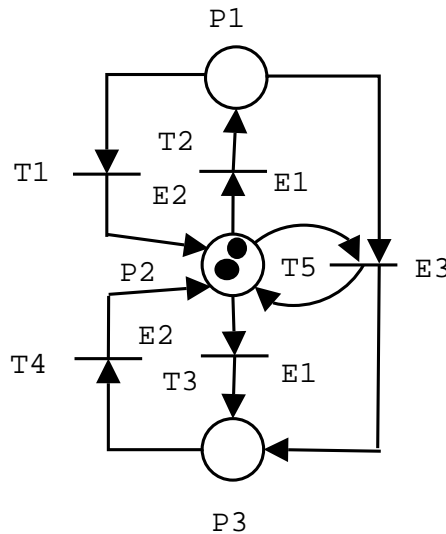


FIG. 8.12 – Vivacité d'un RdP (synchronisé)

8.2 Réseaux de Petri temporisés

Le comportement de ces modèles dépend du temps. Ils permettent d'évaluer les performances temporelles d'un système. Par exemple, il peut s'agir d'évaluer le temps d'exécution d'une série d'opérations (par exemple dans un procédé de fabrication industriel) en fonction de la capacité des machines et/ou des stocks, etc..

Exemple : atelier de coupe Sur le modèle RdP de l'atelier de coupe, la présence d'un jeton dans la place P_3 correspond au traitement d'une commande par la machine de coupe. La découpe elle-même durera un certain temps d_3 . Quand une marque est introduite dans la place P_3 (début de découpe), pour modéliser ce phénomène, il est nécessaire de "bloquer" la marque dans la place P_3 pendant au moins un temps d_3 (voir Figure 8.13). Pendant ce laps de temps, la marque n'est pas disponible pour la validation de la transition T_3 . On parle alors de *marque indisponible*. La place est dotée d'une *temporisation* de durée d_3 , c'est-à-dire, le laps de temps qu'il est nécessaire pour que la *marque* devienne *disponible* pour la validation de sa transition de sortie.

Definition 8.2.1 On appelle **Réseau de Petri P-temporisé** le doublet $\langle R, Tempo \rangle$ où

- R est un Réseau de Petri marqué ;
- $Tempo : P \rightarrow \mathbb{Q}^+$ où
 - \mathbb{Q}^+ est l'ensemble des nombres rationnels positifs ou nuls ;
 - $Tempo(P_i)$ est la temporisation associée à la place P_i .

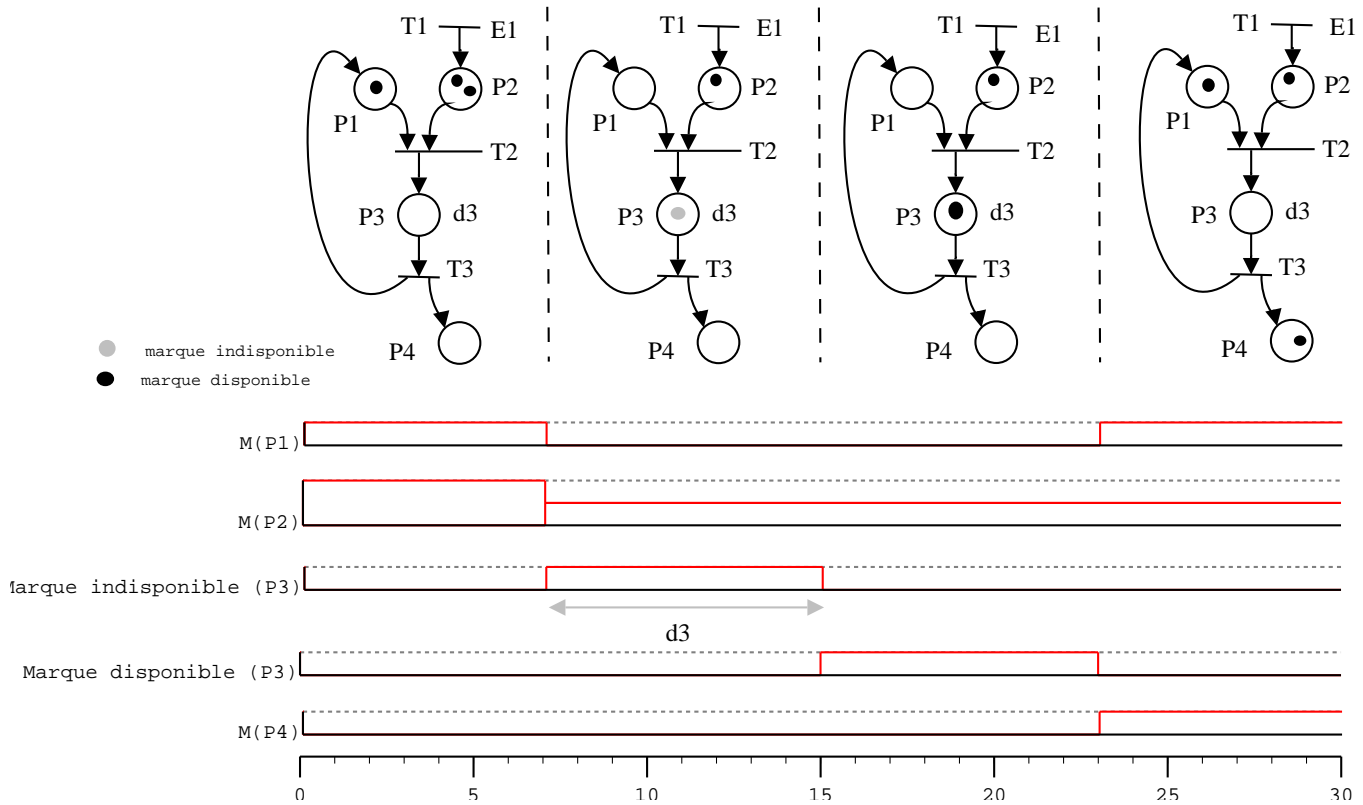


FIG. 8.13 – Atelier de coupe

Dans un RdP P-temporisé, quand une marque est déposée dans la place P_i , elle reste indisponible pendant un temps de $Tempo(P_i)$. Une fois ce temps écoulé, elle devient disponible. Pour déterminer si une transition est validée, ne sont pris en compte que les marques disponibles. Le marquage initial est supposé disponible au temps 0. Le franchissement d’une transition validée est supposé de durée nulle.

Dans l’exemple représenté Figure 8.13, quand la marque de la place P_1 est devenue disponible, la transition T_3 a été validée. On a néanmoins attendu 8 unités de temps avant de la franchir. Dans la cas où dès qu’une transition est validée, elle est immédiatement franchie, on parle de *fonctionnement à vitesse maximale*. Pour l’exemple précédent, si on associe la temporisation d_1 à la place P_1 , cela donnerait l’évolution³ représentée Figure 8.14.

Propriété 8.2.1 *Pour un RdP P-temporisé, le fonctionnement à vitesse maximale conduit à un fonctionnement périodique au bout d’un temps fini pour tout marquage initial tel que le RdP P-temporisé soit borné.*

Il est intéressant de caractériser ce fonctionnement périodique.

Definition 8.2.2 *La fréquence de franchissement $F_j > 0$ d’une transition T_j est définie comme le nombre moyen de franchissements de T_j par unité de temps lorsque le fonctionnement périodique est établi.*

Soit W (W^+) la matrice d’incidence (arrière) associée(s) au RdP. Soient f_i les q P-semi flots élémentaires associés à W . Soit D la matrice diagonale telle que $D_{ii} = d_i$ où d_i est la temporisation associée à la place P_i . Alors le vecteur $F = [F_j]$ de fréquences de franchissement est

³Pour les places pour lesquelles une temporisation n’est pas indiquée, elle vaut implicitement 0.

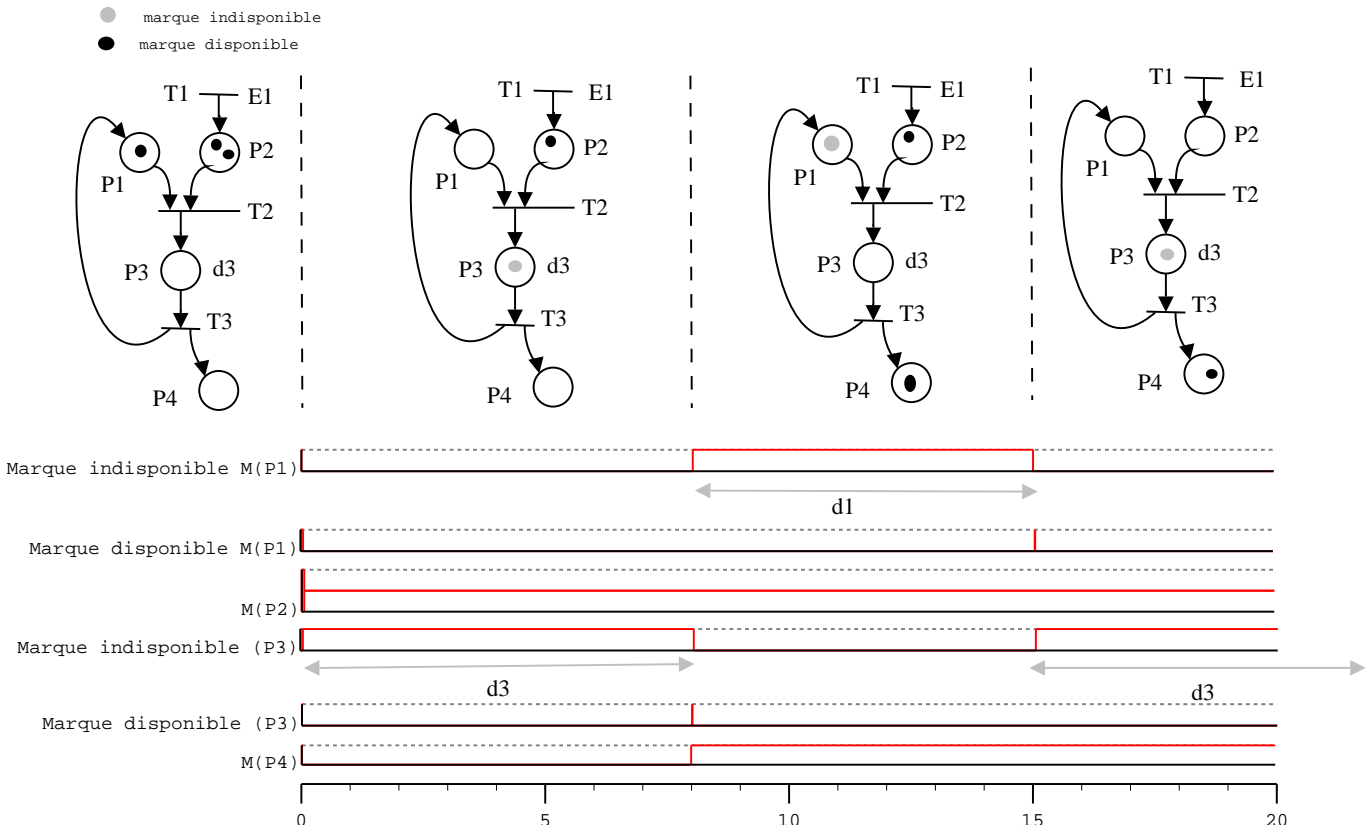


FIG. 8.14 – Atelier de coupe

solution de

$$WF = 0 \quad \text{et} \quad \forall i \in \{1, \dots, q\}, \quad f_i^T M_0 \geq f_i^T DW + F$$

Démonstration Soit

- M_t le marquage du RdP P temporisé à l’instant t ;
- W (W^+) sa matrice d’incidence (arrière) ;
- un temps t_2 suffisamment plus grand qu’un temps t_1 , t_1 étant lui même plus grand pour être dans un fonctionnement périodique ;
- \underline{s} le vecteur caractéristique de la séquence de franchissement permettant de passer du marquage M_{t_1} au marquage M_{t_2} .

D’après l’équation fondamentale, on a :

$$M_{t_2} = M_{t_1} + W\underline{s}.$$

Or, $F = \frac{1}{t_2 - t_1} \underline{s}$. Par suite :

$$M_{t_2} = M_{t_1} + (t_2 - t_1)WF.$$

Après un régime transitoire, le fonctionnement devient périodique : il existe donc $t_2 > t_1$ tels que $M_{t_2} = M_{t_1}$. Donc

$$WF = 0.$$

Pour la place P_i , on peut définir le nombre moyen de marques lors d’un fonctionnement en régime périodique. Soient $M_{t_{k_1}}, \dots, M_{t_{k_p}}$ les marquages successivement atteints lors d’une

période avec pour durée $\delta_1, \dots, \delta_p$. Alors la valeur moyenne peut être définie par

$$\bar{M} = \frac{\sum_{i=1}^p \delta_i M_{t_{k_i}}}{\sum_{i=1}^p \delta_i}$$

De plus, le fait que chaque marque reste au moins d_i unités de temps dans la place $P(i)$ se traduit par :

$$\bar{M}(P_i) \geq d_i W_i^+ F$$

où W_i^+ est la ligne i de la matrice W^+ . En effet :

- $W_i^+ F$ est le nombre de marques mis dans la place P_i par unité de temps ;
- d_i étant le temps minimum où une marque reste dans la place P_i , le nombre moyen de marques dans P_i sera donc au minimum de $d_i W_i^+ F$;
- Chaque marque peut rester dans la place P_i une durée supérieure ou égale à d_i : le nombre moyen de marques dans P_i , $\bar{M}(P_i)$, sera donc supérieur ou égal à $d_i W_i^+ F$.

L'ensemble de ces inégalités peut s'écrire :

$$\bar{M} \geq DW^+ F \tag{8.1}$$

Si f est un P semi-flot, on a $f^T \bar{M} = f^T M_0$. En effet :

$$f^T \bar{M} = \frac{\sum_{i=1}^p \delta_i f^T M_{t_{k_i}}}{\sum_{i=1}^p \delta_i}$$

Or, il existe une séquence de franchissement de vecteur caractéristique \underline{s}^{k_i} telle que :

$$M_{t_{k_i}} = M_0 + W \underline{s}^{k_i}$$

Donc $f^T M_{t_{k_i}} = f^T M_0$ et $f^T \bar{M} = f^T M_0$. A partir de l'inéquation (8.1), on obtient donc :

$$f^T \bar{M} \geq f^T DW^+ F$$

et comme $f^T \bar{M} = f^T M_0$,

$$f^T M_0 \geq f^T DW^+ F.$$

□

Exemple On considère le fonctionnement d'une entreprise de location de voitures. Cette entreprise a deux types de clients :

- chacun des N_1 clients de type 1 utilise une voiture pendant une durée d_1 ; le temps moyen entre deux demandes est de d_{d_1} ;
- chacun des N_2 clients de type 2 utilisent une voiture pendant une durée d_2 ; le temps moyen entre deux demandes est de d_{d_2} .

N_0 représente le nombre de voitures de l'entreprise. Après chaque retour de location, la voiture passe au garage pour entretien pour une durée d_e . On désire déterminer la durée d_0 minimale d'attente d'une voiture après entretien et avant location.

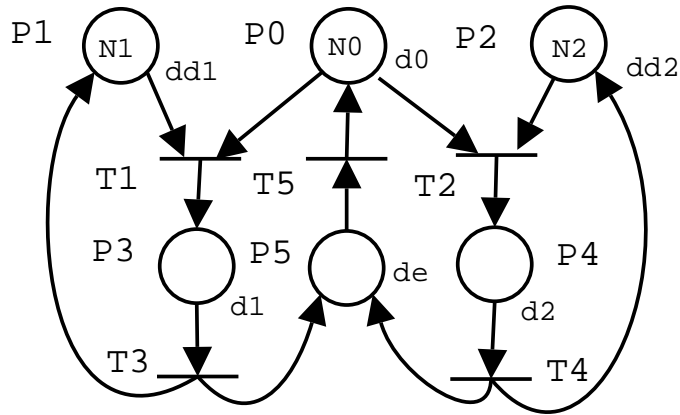


FIG. 8.15 – Entreprise de location de voitures

Le fonctionnement de cette entreprise est représenté par le Rdp P-temporisé Figure 8.15. Les matrices d'incidence (arrière) sont données par :

$$W = \begin{matrix} & T_1 & T_2 & T_3 & T_4 & T_5 \\ \begin{matrix} P_0 \\ P_1 \\ P_2 \\ P_3 \\ P_4 \\ P_5 \end{matrix} & \begin{bmatrix} -1 & -1 & 0 & 0 & 1 \\ -1 & 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 \\ 1 & 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 1 & 1 & -1 \end{bmatrix} \end{matrix} \quad \text{et} \quad W^+ = \begin{matrix} & T_1 & T_2 & T_3 & T_4 & T_5 \\ \begin{matrix} P_0 \\ P_1 \\ P_2 \\ P_3 \\ P_4 \\ P_5 \end{matrix} & \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \end{bmatrix} \end{matrix}$$

Par suite,

$$WF = 0 \Leftrightarrow \begin{cases} F_1 = F_3 \\ F_2 = F_4 \\ F_5 = F_3 + F_4 \end{cases}$$

Les P-semi flots sont déterminés à partir de l'équation :

$$f^T W = 0 \Leftrightarrow \begin{cases} q_3 = q_0 + q_1 \\ q_4 = q_0 + q_2 \\ q_5 = q_0 \end{cases}$$

Par suite, tout P-semi flot s'écrit : Par suite,

$$\forall i, f_i^T M_0 \geq f_i D W^+ F \Leftrightarrow \begin{cases} (d_0 + d_1 + d_e)F_3 + (d_0 + d_2 + d_e)F_4 \leq N_0 \\ (d_1 + d_{d_1})F_3 \leq N_1 \\ (d_2 + d_{d_2})F_4 \leq N_2 \end{cases}$$

Le fonctionnement le plus rapide est obtenu avec $F_3 = \frac{N_1}{(d_1 + d_{d_1})}$ et $F_4 = \frac{N_2}{(d_2 + d_{d_2})}$. D'où :

$$(d_0 + d_1 + d_e) \frac{N_1}{(d_1 + d_{d_1})} + (d_0 + d_2 + d_e) \frac{N_2}{(d_2 + d_{d_2})} = N_0$$

⇕

$$d_0 = \frac{1}{\frac{N_1}{(d_1 + d_{d_1})} + \frac{N_2}{(d_2 + d_{d_2})}} \left(N_0 - \frac{d_e + d_1}{d_1 + d_{d_1}} N_1 - \frac{d_e + d_2}{d_2 + d_{d_2}} N_2 \right).$$

Remarque Dans l'exemple précédent, on est dans un mode de fonctionnement où

$$\forall i, \quad f_i^T M_0 = f_i DW^+ F.$$

Ce mode de fonctionnement est appelé **fonctionnement en vitesse propre**. Dans celui-ci, une marque reste dans une place qu'un temps égal à la temporisation associée à cette place. Il n'est pas possible pour tout RdP P-temporisé. C'est un cas particulier du fonctionnement à vitesse maximale.

Remarque De manière similaire aux RdPs P-temporisés, il est possible de définir des RdPs T-temporisés : les temporisations sont associées aux transitions au lieu d'être associées aux places. On peut toujours construire pour un RdP T-temporisé un RdP P-temporisé équivalent.

Bibliographie

- [1] G.W. Brams. *Réseaux de Petri : Théorie et pratique*. Masson, 1983.
- [2] R. David and H. Alla. *Du Grafset aux réseaux de Petri*. Hermes, 1992.
- [3] K. Jensen. *Coloured Petri Nets*, volume 1. Springer-Verlag, 1992.
- [4] G. Vidal-Naquet and A. Choquet-Geniet. *Réseaux de Petri et Systèmes Parallèles*. Armand Colin, 1992.