Méthodes d'analyse hiérarchique des réseaux de Petri

Ghassan Chehaibar

THESE DE DOCTORAT
DE L'ECOLE NATIONALE DES PONTS ET CHAUSSEES

Spécialité Informatique

présentée par

Mr. Ghassan CHEHAIBAR

pour obtenir le titre de

DOCTEUR DE L'ENPC

Sujet de la thèse:

Méthodes d'Analyse Hiérarchique des Réseaux de Petri

Soutenue le 24 Septembre 1991, devant le jury composé de

Messieurs

C. GIRAULT	Président
C. ANDRÉ	Rapporteur
G. MEMMI	Rapporteur
S. HADDAD	Examinateur
R. LALEMENT	Examinateur
R. VALK	Examinateur

Méthodes d'Analyse Hiérarchique des Réseaux de Petri

Ghassan CHEHAIBAR

18 Septembre 1991

à mes parents

Remerciements

C'est à la suite d'un cours de Monsieur le Professeur Claude GIRAULT que je me suis intéressé aux systèmes informatiques, puis à la modélisation de ces systèmes au moyen des réseaux de Petri. Il m'a dirigé tout au long de mon DEA et de la préparation de la thèse, en me prodiguant conseils et encouragements. Toutes les notions ont été développées sur des exemples qu'il m'a fournis et suite à de longues discussions; ses nombreuses suggestions m'ont constamment aidé à améliorer mon travail.

Les travaux de Monsieur le Professeur Charles ANDRÉ sur les équivalences de comportements ont servi de point de départ à mes réflexions sur les équivalences. Je lui suis reconnaissant d'avoir consacré une partie de son temps à une lecture attentive de mon travail.

Monsieur Gérard MEMMI a contribué à ma formation en réseaux de Petri. Je le remercie d'avoir accepté d'être rapporteur de cette thèse malgré ses nombreuses obligations.

Monsieur Serge HADDAD m'a fait l'amitié d'accepter de faire partie de ce jury. Les quelques discussions que j'ai eues avec lui, au cours de la préparation de ma thèse, me font regretter de n'avoir pas eu l'occasion de travailler plus souvent avec lui.

Je remercie Monsieur René LALEMENT d'avoir accepté de faire partie de ce jury et de s'être intéressé à mon travail.

Je remercie Monsieur le Professeur Rüdiger VALK de l'honneur qu'il me fait en acceptant de participer à ce jury.

Grâce à l'appui de Monsieur Nicolas BOULEAU, directeur du Centre d'Enseignement et de Recherche en Mathématiques Appliquées (CERMA), et de Monsieur René LALEMENT, directeur-adjoint du Centre d'Enseignement et de Recherche en Modélisation Informatique et Calcul Scientifique (CERMICS), j'ai pu bénéficier d'une bourse de l'Ecole des Ponts durant les deux premières années de ma thèse.

Ma troisième année s'est déroulée à la Direction Recherche et Programmes Avancés de BULL, grâce à l'appui de Monsieur Gérard MEMMI, au sein de l'équipe Méthodes de Spécifications Formelles dirigée par Monsieur Dominique BOLIGNANO, que je tiens à remercier pour m'avoir permis de mener à terme ma thèse dans de bonnes conditions.

Résumé L'objet de cette thèse est de définir des méthodes d'analyse hiérarchique des réseaux de Petri (un formalisme de modélisation du contrôle des systèmes distribués). Premièrement, nous étudions la transformation de réseaux par remplacement de sous-réseaux ouverts: une notion d'équivalence est définie telle que si on remplace un sous-réseau ouvert par un réseau équivalent on obtient un réseau équivalent au réseau d'origine; puis nous définissons la classe des réseaux réentrants, pour laquelle la vérification de l'équivalence, complexe en général, devient simple. Deuxièmement, l'examen de la composition de réseaux par fusion de places conduit à deux résultats: la conservation d'espace d'accueil dans certaines conditions d'indépendance par rapport aux places partagées, et celle de l'absence d'interblocage lors de la composition des réseaux à ressources ordonnées (dans un sens différent de la solution classique). Enfin, un nouveau concept de norme pour la preuve de la propriété d'espace d'accueil est défini, muni d'opérations de composition permettant une vérification modulaire de cette propriété.

Mots-clés réseaux de Petri, analyse hiérarchique, remplacement, réseaux réentrants, composition, espace d'accueil, normes

Abstract The aim of this thesis is to define hierarchical analysis methods for Petri nets (a formalism for modelling distributed system control). First, we study the net transformation consisting in replacing open subnets: an equivalence notion is defined, such that if an open subnet is replaced with an equivalent net, the net obtained is equivalent to the original one; then, we define the reentrant net class, for which the equivalence verification, complex in general, becomes simple. Secondly, the investigation of net composition by merging places gives two results: the preservation of home space property if some conditions of independance wrt the shared places are fulfilled, and the preservation of deadlock-freeness when composing ordered-resource nets (in a sense different form the classical solution). Finally, a new norm concept for proving home space property is defined, with composition operations allowing a modular verification of this property.

Keywords Petri nets, hierarchical analysis, replacement, reentrant nets, composition, home space, norms

Table des matières

Introduction

Raisonner sur un système distribué dans son ensemble est problématique, car l'explosion combinatoire des configurations globales rend la spécification et la preuve impraticables. La recherche de concepts de modularité mobilise les spécialistes de tous les formalismes de modélisation du parallélisme, d'autant plus que les systèmes distribués sont un domaine où la modularité apparaît naturellement.

La devise de la modularité est diviser pour régner. On distingue deux concepts de modularité: l'horizontale—la composition, et la verticale—le raffinement; et ceci à deux niveaux: la spécification—description qui prend en compte la répartition des objets (ou strucuration), et la preuve—construction d'une preuve globale à partir de preuves locales.

La composition consiste à spécifier un système comme l'assemblage de plusieurs systèmes, cet assemblage ayant une sémantique précise équivalente à celle du système global. La preuve est modulaire si elle se déduit des preuves de chaque constituant (à la composition de spécifications correspond une composition de preuves).

Le raffinement consiste à définir un système avec un système racine et un ensemble de transformations qui sont soit des enrichissements, soit des remplacements d'une partie par un système. Ici la preuve est dite modulaire si la preuve du système, après une transformation, peut être déduite de la preuve de son prédécesseur et de l'étude locale de la partie raffinée et de son raffinement (au raffinement de spécification correspond un raffinement de preuve).

Bien que le formalisme des réseaux de Petri ne contienne pas de constructeurs modulaires dans sa définition, de par sa nature graphique et la représentation locale des états et des transitions, le raffinement et la composition y sont des notions très naturelles: raffinement d'un sommet ou composition de réseaux par fusion de sommets.

La sructuration hiérarchique de la conception et de la spécification améliore la lisibilité de la spécification et permet d'éviter un bon nombre d'erreurs. Cepen-

dant, le problème fondamental est d'analyser un tel modèle hiérarchiquement, car construire une preuve du modèle mis à plat est trop coûteux et ne permet pas de localiser les erreurs.

Dans cette thèse, nous étudions des méthodes d'analyse hiérarchique associées à la composition de réseaux et au remplacement de sous-réseaux, sans nous attacher à définir un réseau de Petri hiérarchique avec sa sémantique. Avant d'exposer les résultats obtenus, nous faisons un panaroma des travaux antérieurs sur la modularité et la hiérarchie.

Les notions de raffinement, de morphisme de réseaux et de sous-réseaux ouverts ou fermés sont apparues assez tôt dans le développement de la théorie des réseaux de Petri[26]. Cependant, il s'agissait essentiellement de concepts graphiques sans aucun lien avec les comportements des réseaux: les propriétés d'un morphisme de réseaux étaient purement topologiques.

Les premières études de décomposition de réseaux sont réalisées par Hack[29] en 73, sur la décomposition en machines à états finis. Symétriquement, les machines à états synchronisées par tampons (fusion de places) sont considérées par Reisig[49] en 79. Ces travaux sont poursuivis et généralisés par Memmi[42] en 83 et Souissi[52,53,54] en 90: ils examinent la conservation de la vivacité par composition (par rendez-vous, à travers un medium de communication).

La composition par fusion de transitions est aussi étudiée par Mazurkievicz[41], en 84, via la théorie des traces (où il montre que la composition des ensembles de traces de deux réseaux est égal à l'ensemble des traces du réseau composé).

Winskel[66] en 85 traite la composition dans le cadre de la théorie des catégories, et montre que la fusion de transitions correspond à un produit et celle de places à une somme; de plus il donne une nouvelle définition de morphisme de réseaux liant aussi les comportements.

Récemment, une étude des réseaux algébriques via les catégories compatible avec la composition/décomposition est menée par Petrucci[44,45].

Une méthode de conception et d'analyse par raffinements successifs est donnée par Valette[57] en 79: remplacement d'une transition par un "bloc bien formé." Les conditions d'applications sont généralisées par Suzuki et Murata[55] en 83.

Vogler[60], en 86, présente des réseaux de remplacement plus généraux appelés "modules," et définit une équivalence: l'objectif est alors de conserver l'équivalence et non une propriété particulière comme la vivacité.

En 82, André[2,3,4] définit une équivalence de comportement (B-éq) basée sur l'étiquetage des transitions, et montre que le remplacement d'un sous-réseau fermé (à frontière de transitions) par un réseau B-équivalent sur la frontière donne un réseau B-équivalent au réseau d'origine; de plus, la composition de réseaux par fu-

sion de transitions est considérée aussi par rapport à la B-éq. En 88, Bourguet[9] généralise ces notions, et le remplacement de sous-réseaux fermés et la composition de réseaux par fusion de transitions sont aussi étudiés par Vogler[61] et Baumgarten[6].

Les travaux de l'équipe italienne de l'Université de Milan (De Cindio, De Michelis, Pomello, Simone) débouchent, en 86, sur la définition de deux notions d'équivalence[20,19,46]: EB-éq (Exhibited-Behavior) pour le remplacement de sous-réseaux fermés et EF-éq (Exhibited Functionality) pour le remplacement de sous-réseaux ouverts; ces remplacements sont effectués uniquement sur les SA-nets[18], qui sont des machines à états composées par fusion de transitions, les sous-réseaux remplacés et les réseaux de remplacement devant être eux-mêmes des SA-nets. Dernièrement, les résultats de composition des SA-nets ont été étendus aux OBJSA-nets[5] qui sont des SA-nets augmentés de spécifications algébriques en OBJ.

Il existe une autre approche des raffinements qui ne vise pas à une équivalence entre un réseau et son transformé mais cherche des équivalences qui sont des congruences par rapport aux raffinements: si deux réseaux sont équivalents, ils le restent si on les transforme par le "même" raffinement[62,27].

Les réductions de réseaux développées par Berthelot[8] et Haddad[31] sont une autre voie de réduction de la complexité de l'analyse des réseaux. Ces transformations sont toujours présentées dans le sens réductions d'un grand réseau qui conservent certaines propriétés; cependant, dans une analyse hiérarchique, ce sont les transformations inverses qui sont appropriées: si on remplace un sous-réseau dans N_1 par un réseau plus détaillé pour obtenir N_2, la question est de savoir si cette transformation correspond à une série de réductions inverses de N_1 qui donne N_2.

Nous distinguons trois directions dans notre contribution aux méthodes d'analyse hiérarchique des réseaux de Petri:

- Dans une démarche classique de l'étude des remplacements de sous-réseaux au moyen de relations d'équivalence, et de façon complémentaire aux travaux sur le remplacement de sous-réseaux fermés (à frontière de transitions), nous construisons une théorie de remplacement de sous-réseaux ouverts (à frontière de places). A notre connaissance c'est la seule tentative dans ce domaine, mis à part celle de l'école italienne qui ne se place pas à un tel niveau de généralité.

- Etude de conservation de propriétés par composition de réseaux par fusion de places: conservation d'espace d'accueil, absence de blocage lors de partage

de ressources.

- Dans une démarche plus originale, nous explorons une méthode de réutilisation de preuve, appliquée à la vérification d'espace d'accueil. Au lieu de chercher des transformations de réseaux qui conservent la propriété d'espace d'accueil, nous définissons un nouveau concept de norme muni d'opérations de composition qui permettent de faire de la vérification modulaire.

Le plan de la thèse est le suivant:

Dans le premier chapitre, nous étudions la preuve modulaire d'espaces d'accueil. La notion d'espace d'accueil est fondamentale dans l'analyse des réseaux de Petri, car elle est l'outil le plus utilisé pour prouver des propriétés temporelles de vivacité. Un nouveau concept de norme est défini, muni d'opérations de composition qui rendent possible, dans certaines conditions, la construction d'une norme globale prouvant l'existence d'un espace d'accueil à partir de normes locales prouvant des relations d'accessibilité.

Le deuxième chapitre est consacré à l'étude du raffinement de réseaux par remplacement de sous-réseaux ouverts (à frontière de places). Si on note $N\,[N_1 \leftarrow N_2]$ le remplacement du sous-réseau N_1 par le réseau N_2 dans N, il s'agit de trouver deux notions d'équivalence telles que si $N_1 \equiv N_2$ alors $N\,[N_1 \leftarrow N_2] \equiv' N$.

Nous ne considérons pas de simples réseaux de Petri mais des réseaux où on distingue des places internes, des places d'interface et un ensemble de marquages des places internes (les états stables). Le théorème de remplacement est obtenu pour deux relations d'équivalence appelées SF-eq et SST-eq. Ce sont des bisimulations fondées sur l'étiquetage des places et la comparaison de transformations d'états.

Le troisième chapitre est consacré à la composition de réseaux par fusion de places, que nous appelons composition asynchrone et notons $\otimes$. Sans hypothèses particulières, le lien est faible entre un système composé et ses composants: seuls les flots de $N_1 \otimes N_2$ peuvent être déduits de ceux de N_1 et N_2. Avec une hypothèse d'indépendance de l'accessibilité d'un espace d'accueil par rapport aux places partagées (l'espace est accessible sans franchir des transitions en sortie des places partagées), nous obtenons la conservation de cette propriété.

Mais quand les places partagées sont bornées par leur marquage initial (dans les constituants), les projections des séquences de $(N_1; M_{01}) \otimes (N_2; M_{02})$ appartiennent aux langages de $(N_i; M_{0i})$. Nous considérons alors une classe de réseaux où l'on distingue des places (ressources) ordonnées bornées par leur marquage initial, et nous montrons la conservation de l'absence de blocage lors de la composition

par partage de ressources, dans cette classe; l'ordonnancement des ressources est différent de celui de la solution classique de prévention de l'interblocage.

Dans le quatrième chapitre, nous appliquons les résultats des chapitres précédents à une classe particulière de réseaux, les réseaux réentrants. Pour cette classe, la vérification de l'SST-équivalence définie dans le deuxième chapitre, complexe dans le cas général, devient simple; aussi, certains résultats de la composition asynchrone lui sont appliqués. Cette classe semble utile en pratique car, intuitivement, un réseau réentrant est un serveur qui interdit toute synchronisation entre deux exécutions (que ce soit du même service ou de deux services différents).

Chapitre 0

Préliminaires

0-1 Notations préliminaires

Dans cette section, nous donnons quelques définitions générales sur les relations binaires et les multiensembles.

Notation Si A est un ensemble, $\mathcal{B}(A)$ est l'ensemble des parties de A.

Définition 0-1 (Relations binaires) *Une relation binaire $\mathcal{R}$ d'un ensemble A vers un ensemble B est un sous-ensemble de $A \times B$. $(x, y) \in \mathcal{R}$ est noté $x\mathcal{R}y$. Le domaine de $\mathcal{R}$ est*

$$dom(\mathcal{R}) = \{x \in A; \exists y \in B, x\mathcal{R}y\}$$

et le codomaine

$$cod(\mathcal{R}) = \{y \in B; \exists x \in A, x\mathcal{R}y\}$$

Si $\mathcal{R}_1 \subseteq A \times B$ et $\mathcal{R}_2 \subseteq B \times C$, la composition de $\mathcal{R}_1$ et $\mathcal{R}_2$ est notée $\mathcal{R} = \mathcal{R}_1 \circ \mathcal{R}_2$ et définie par $\mathcal{R} \subseteq A \times C$ tel que

$$x\mathcal{R}z \Leftrightarrow \exists y \in B, x\mathcal{R}_1 y \bigwedge y\mathcal{R}_2 z$$

On rencontre souvent des multiensembles dans les réseaux de Petri puisque le marquage, la fonction de valuation des arcs et les flots sont des multiensembles.

Définition 0-2 (Multiensemble) *Un multiensemble sur X est une fonction $f : X \to \mathbf{N}$. Le support de f est défini par*

$$SPR(f) = \{x \in X; f(x) \neq 0\}$$

L'ensemble des multiensembles sur X est noté $\mathbf{N}^X$.

Dans ce qui suit on définit la somme, la comparaison et la soustraction de multiensembles. Il s'agit d'une généralisation de l'union, de l'inclusion et de la soustraction d'ensembles et pas d'un report de ces notions de $\mathbf{N}$ vers les fonctions.

Définition 0-3 (Somme) *Si $f_i \in \mathbf{N}^{X_i}$ pour $i = 1, 2$, alors $f = f_1 + f_2$ est définie par*

$$f \in \mathbf{N}^{X_1 \cup X_2} \bigwedge f(x) = \begin{cases} f_i(x) & si\ x \in X_i \setminus X_j \\ f(x) = f_1(x) + f_2(x) & si\ x \in X_1 \cap X_2 \end{cases}$$

On note selon les cas $f_1 + f_2$ ou $f_1 \cup f_2$.

Définition 0-4 (Comparaison) *Si $f \in \mathbf{N}^X$ et $g \in \mathbf{N}^Y$ tels que $X \subseteq Y$, alors*

$$f \leq g \Leftrightarrow \forall x \in X, f(x) \leq g(x)$$

Définition 0-5 (Soustraction) *Si $f \in \mathbf{N}^X$ et $g \in \mathbf{N}^Y$, alors $f - g$ est défini par*

$$(f - g) \in \mathbf{N}^X \bigwedge (f - g)(x) = \begin{cases} f(x) & si\ x \in X \setminus Y \\ \max(0, f(x) - g(x)) & si\ x \in X \cap Y \end{cases}$$

Quand on compose et décompose des réseaux, on rencontre souvent l'opérateur de restriction (ou de projection): on prend la restriction d'un marquage du réseau total à un sous-réseau ou bien on considère la restriction d'une séquence de franchissements. Par exemple, si un réseau N est la composition de N_1 et N_2, et M un marqage de N, on note $M{\downarrow_{P_1}}$ le marquage correspondant de N_1. Inversement, on a souvent besoin de considérer l'ensemble des marquages de N dont la restriction à N_1 appartient à un ensemble de marquages Q_1: on notera cet ensemble $Q_1{\uparrow^P}$.

Définition 0-6 (Restriction et extension de multiensembles) *Si $f \in \mathbf{N}^X$ et $X' \subseteq X$, alors $f{\downarrow_{X'}}$ est la restriction de f à X'.*

A étant un ensemble, et $Q \subseteq \mathbf{N}^A$, la restriction de Q à $B \subseteq A$ est

$$Q{\downarrow_B} = \{f \in \mathbf{N}^B; \exists f' \in Q, f = f'{\downarrow_B}\}$$

et l'extension de Q à $B \supseteq A$ est

$$Q{\uparrow^B} = \{f \in \mathbf{N}^B; f{\downarrow_A} \in Q\}$$

Définition 0-7 (Restriction de séquence) *Si $\sigma \in A^*$ et $A' \subseteq A$, $\sigma{\downarrow_{A'}}$ est la séquence obtenue par suppression dans σ de toutes les transitions $t \notin A'$.*

0-2 Réseaux de Petri

Les résultats de cette thèse concernent les réseaux places/transitions à capacités infines; quelques extensions aux réseaux colorés sont données rapidement. Dans cette section, nous donnons les principales définitions de réseaux[11,50].

0-2.1 Réseaux P/T

Définition 0-8 (Réseau P/T) *Un réseau place/transition (P/T) est un triplet* $N = (P, T; W)$ *où*

- P *et* T *sont des ensembles finis (ensemble de places et ensemble de transitons)*
- $W : (P \times T) \cup (T \times P) \to \mathbf{N}$ *est la fonction de valuation.*

La matrice d'incidence est $C : P \times T \to \mathbf{N}$ définie par

$$C(p,t) = W(t,p) - W(p,t)$$

W et C sont étendues à $(P \times T^*) \cup (T^* \times P)$ et $P \times T^*$ comme d'habitude[11]: soit $\sigma \in T^*$, $t \in T$ et λ le mot vide

$$C(p,\lambda) = 0 \bigwedge C(p,\sigma t) = C(p,\sigma) + C(p,t)$$

$$W(p,\lambda) = 0 \bigwedge W(p,\sigma t) = \max(W(p,\sigma), W(p,t) - C(p,\sigma))$$

$$W(\lambda,p) = 0 \bigwedge W(\sigma,p) = C(p,\sigma) + W(p,\sigma)$$

Le pré-ensemble (resp. post-ensemble) d'un ensemble de sommets X est noté ${}^\bullet X$ (resp. $X^\bullet$), et défini par

$${}^\bullet X = \{y; \exists x \in X, W(y,x) > 0\}$$

$$X^\bullet = \{y; \exists x \in X, W(x,y) > 0\}$$

${}^\bullet X^\bullet$ désigne ${}^\bullet X \cup X^\bullet$.

Si $\sigma \in T^*$ et si $t \in T$, alors $\bar{\sigma}(t)$ représente le nombre d'occurrences de t dans σ.

Les réseaux peuvent être non connexes, impurs et avoir des sommets isolés; les éventuelles restrictions seront signalées. Nous avons préféré d'utiliser la fonction W plutôt que Pre et $Post$ car elle est plus "visuelle": $W(x,y)$ est la valuation de

l'arc (x, y). Mais quand on voudra manipuler les matrices on utilisera ces fonctions définies par $Pre(p,t) = W(p,t)$ et $Post(p,t) = W(t,p)$.

On suppose qu'il existe deux ensembles $\mathcal{P}$ et $\mathcal{T}$ tels que $\mathcal{P} \cap \mathcal{T} = \emptyset$, et tous les réseaux P/T considérés vérifient $P \subseteq \mathcal{P}$ et $T \subseteq \mathcal{T}$: donc on peut parler de l'ensemble des réseaux P/T vérifiant une certaine propriété.

Définition 0-9 (Système P/T) *Un réseau P/T marqué ou un système P/T est un couple $\Sigma = (N; M_0)$ où N est un réseau P/T et $M_0 \in \mathbf{N}^P$ (le marquage initial).*

On adopte la règle de franchissement séquentielle faible:

$$M \xrightarrow{\sigma} M' \text{ ssi } \forall p \in P, M(p) \geq W(p,\sigma) \text{ et } M'(p) = M(p) + C(p,\sigma)$$

L'ensemble des marquages accessibles est noté

$$R(N; M_0) = \{M \in \mathbf{N}^P; \exists \sigma \in T^*, M_0 \xrightarrow{\sigma} M\}$$

0-2.2 Réseaux colorés

Nous reprenons la première définition des réseaux colorés de Jensen[34], avec quelques changements de notations.

Définition 0-10 (Réseau coloré) *Un réseau coloré est un quadruplet $N = (P, T; D; W)$ où*

- *P et T sont des ensembles finis (ensemble de places et ensemble de transitions)*
- *D une famille d'ensembles finis indexée par $P \cup T$: si $x \in P \cup T$, $D(x)$ est le domaine de couleur de x.*
- *W est la fonction de valuation définie sur $(P \times T) \cup (T \times P)$ telle que si $(x,y) \in \{(p,t),(t,p)\}$ où $(p,t) \in P \times T$, alors $W(x,y) : D(t) \times D(p) \to \mathbf{N}$.*

La matrice d'incidence C est définie par

$$C(p,t) = W(t,p) - W(p,t)$$

Un marquage d'un réseau coloré est une fonction M définie sur P telle que $M(p) : D(p) \to \mathbf{N}$. L'ensemble des marquages potentiels d'un réseau coloré est noté

$$POT(N) = \{M; M(p) : D(p) \to \mathbf{N}\}$$

Chapitre I

Espaces d'accueil

Introduction

Un état d'accueil d'un système est un état qu'il est toujours possible d'atteindre, quelle que soit l'évolution de ce système. Ce concept a été introduit par Keller[36] pour les systèmes de transitions, et il a proposé le concept de norme (inspiré de Floyd) comme outil de preuve. Une norme est une fonction à valeurs entières définie sur les états, qui vaut zéro sur l'état d'accueil et qu'il est toujours possible de faire décroître.

Dans BRAMS[11], on trouve la transposition de cette notion aux réseaux de Petri, et des exemples de normes qui sont des combinaisons linéaires des marquages des places.

La notion d'espace d'accueil est une généralisation de celle d'état d'accueil due à Memmi[42,43]: au lieu de pouvoir toujours atteindre un état fixé, on peut toujours atteindre un état appartenant à un ensemble fixé (c-à-d, un état vérifiant une certaine propriété). La vérification d'espace d'accueil joue un rôle essentiel dans l'analyse des réseaux de Petri car il s'agit de montrer qu'on peut toujours atteindre une certaine propriété: cette notion est utile dans la preuve de propriétés temporelles des systèmes.

Memmi avait donné une autre méthode de preuve que la norme: preuve par raffinements successifs qui consiste en des réductions successives d'un espace contenant tous les marquages accessibles pour arriver à l'espace donné. Dans sa thèse, Johnen[35] a étudié la décidabilité et la vérification automatique des espaces d'accueil (utilisant des techniques de réécriture), et a proposé un algorithme qui est une formalisation et une automatisation de cette méthode. Cet algorithme prend en entrée l'espace global G (contenant tous les marquages accessibles) défini par des équations linéaires des marquages (égalités et inégalités) et l'espace E

supposé être espace d'accueil: puis il vérifie par raffinements successifs que E est accessible à partir de G.

Nous visons à munir les espaces d'accueil de méthodes de preuve compositionnelles. Nous proposons un nouveau concept de norme qui réunit ces deux méthodes: nous conservons la notion de norme qui est une fonction sur les marquages, mais en plus, ces normes peuvent représenter le passage d'un ensemble d'états à un autre et donc un pas de la preuve par raffinements successifs: une norme qui prouve l'existence de l'espace d'accueil est la composition de ces normes.

Ainsi ces normes sont munies d'opérateurs de composition qui permettent de les construire par raffinements successifs, et en plus, de construire la norme d'une composition de réseaux en composant leurs normes (dans certaines conditions).

Dans la première section, la définition d'un espace d'accueil est rappelée, et nous donnons une caractérisation de cette propriété en termes d'ordre bien fondé.

Dans la deuxième section, nous définissons un nouveau concept de norme, avec un cas particulier: les normes linéaires. Ces normes sont des applications de l'ensemble des marquages d'un réseau vers N^k, associées à des relations d'accessibilité.

La troisième section présente les opérations de composition de normes. La première est l'ordonnancement de normes, qui correspond à la preuve d'espace d'accueil par réductions successives d'un espace global. La deuxième opération est la somme de norme, qui permet de montrer que l'intersection de deux espaces d'accueil en est un, si une certaine condition "d'indépendance" est vérifiée.

La quatrième section est consacrée à la vérification hiérarchique des normes. Nous définissons la preuve d'une norme et la composition de ces preuves. Une méthode de réutilisation de preuve dans la vérification de normes est illustrée par des exemples au moyen de ces notions.

Nous définissons aussi une notion d'espace d'accueil plus précise: un A-espace d'accueil est un espace d'accueil accessible sans franchir des transitions dans A. Ceci peut s'exprimer par une propriété des preuves d'une norme associée à cet espace.

I-1 Ordre bien fondé sur les états

Les espaces d'accueil sont un des concepts de base pour l'expression et la preuve des propriétés temporelles des réseaux de Petri. Souvent la preuve de l'absence de blocage consiste à montrer qu'il existe un espace d'accueil E tel que pour tout marquage de E, il existe une séquence franchissable.

D'autre part, les espaces d'accueil sont utilisables dans la preuve de certaines relations synchroniques. Si on veut prouver que les occurrences de deux transitions a et b forment un langage de parenthèses dont le niveau d'imbrication est borné par n, il suffit d'ajouter une place p entre a et b, de montrer que p est implicite et bornée par n, puis que $E = \{M; M(p) = 0\}$ est un espace d'accueil.

Nous rappelons d'abord la définition d'un espace d'accueil, puis nous donnons une caractérisation de cette propriété en fonction de l'existence d'un ordre bien fondé sur les états accessibles.

Définition I-1 (Espace d'accueil) *Soit $\Sigma = (P, T; W; M_0)$ un système P/T. Un ensemble de marquages $H \subseteq \mathbf{N}^P$ est un* espace d'accueil *de Σ ssi*

$$\forall M \in R(\Sigma), \exists \sigma \in T^*, \exists M' \in H, M \xrightarrow{\sigma} M' \in H$$

Si $\{M\}$ est un espace d'accueil alors M est appelé état d'accueil.

La preuve de la propriété d'espace d'accueil, que ce soit au moyen d'une norme ou par raffinements successifs d'espaces, repose sur la notion d'ordre bien fondé. Un ordre sur un ensemble E est bien fondé s'il n'existe pas dans E de suite infinie strictement décroissante.

Définition I-2 (Ordre bien fondé) *Un ordre $\leq$ sur un ensemble E est dit bien fondé, s'il n'existe pas $(a_i)_{i \in \mathbf{N}} \in E^{\mathbf{N}}$ tel que $a_0 > a_1 > \cdots a_i > a_{i+1} > \cdots$*

Théorème I-1 *H est un espace d'accueil de Σ si, et seulement si, il existe un ordre (partiel) bien fondé $\leq$ sur $R(\Sigma)$, tel que $H \cap R(\Sigma)$ contienne l'ensemble des éléments minimaux de $R(\Sigma)$, et si $M \in R(\Sigma)$ n'est pas minimal, alors il existe $\sigma \in T^*$, $M \xrightarrow{\sigma} M'$ et $M' < M$.*

Preuve

-) Si H est un espace d'accueil, on définit $<$ par $M < Q$ ssi $M \in H$, $Q \notin H$ et $\exists \sigma \in T^*, Q \xrightarrow{\sigma} M$.

-) La réciproque est prouvée au moyen du principe de l'induction nœthérienne, qui s'énonce ainsi: si E est un ensemble muni d'un ordre bien fondé $\leq$, et si $A \subseteq E$ alors

$$\forall x \, [(\forall y < x, y \in A) \Rightarrow x \in A] \Rightarrow A = E$$

On suppose donc qu'il existe un ordre bien fondé $\leq$ sur $R(\Sigma)$ vérifiant les hypothèses énoncées dans le théorème. On considère

$$A = \{M \in R(\Sigma); \exists \sigma, M \xrightarrow{\sigma} M' \in H\}$$

Pour $M \in R(\Sigma)$, on note

$$B(M) = \{m \in R(\Sigma); m < M\}$$

Il faut montrer que $B(M) \subseteq A \Rightarrow M \in A$ pour en déduire $A = R(\Sigma)$ et par conséquent H espace d'accueil.

Si $M \in H$, alors $B(M) = \emptyset \subseteq A$ et $M \in A$ (on prend $\sigma = \lambda$).

Si $M \notin H$ et $B(M) \subseteq A$. Par hypothèse (du théorème), il existe σ, $M \xrightarrow{\sigma} m$ et $m < M$, c-à-d $m \in B(M)$. Par hypothèse d'induction, il existe σ' telle que $m \xrightarrow{\sigma'} m' \in H$, et donc $M \xrightarrow{\sigma''} m'$ où $\sigma'' = \sigma\sigma'$. D'où, $M \in A$. □

Quand on prouve un espace d'accueil par raffinements successifs, on part d'un espace global E_0 (contenant l'ensemble d'accessibilité), et on montre qu'il existe une suite d'espaces $(E_i)_{i=0,n}$ tels qu'on puisse atteindre E_{i+1} à partir de E_i, $E_{i+1} \subseteq E_i$ et $E_n = H$. L'ordre sous-jacent ici, est $M' < M$ ssi il existe $i < j$ tels que $M' \in E_j$ et $M \in E_i$.

Une norme classique est une application de l'ensemble des états accessibles vers $\mathbf{N}$, et donc transporte l'ordre de $\mathbf{N}$ sur ces états. Sa preuve part aussi d'un espace global et se fait par disjonction de cas (qui correspond à une partition de cet espace), et on montre que dans chaque cas il est possible d'atteindre un état de norme inférieure.

I-2 Normes

Dans tous les cas, un pas de preuve d'espace d'accueil consiste à montrer qu'il existe une séquence menant d'un ensemble d'états à un autre ensemble d'états d'ordre strictement inférieur. Le concept de norme que nous définissons représente ce pas de preuve, et répond au souci d'avoir un outil maniable de preuve qui se prête à la composition. Notons que la norme est définie pour un réseau non marqué, et que c'est uniquement une relation d'accessibilité.

Pour ordonner les états, nous considérons des normes qui sont des applications à valeurs dans $\mathbf{N}^k$ muni de l'ordre lexicographique. Le choix de cet ordre est justifié par la définition des normes linéaires et les compositions de normes définies dans la suite.

Définition I-3 (Norme) *Soit $N = (P, T; W)$ un réseau de Petri, J et K deux sous-ensembles de $\mathbf{N}^P$ tels que $J \supseteq K$. Une (J, K)-norme sur N est une fonction $\nu : \mathbf{N}^P \to \mathbf{N}^k$ $(k > 0)$ telle que*

- $\forall m \in J, \nu(m) = 0 \Leftrightarrow m \in K$

- *si* $m \in J \wedge \nu(m) > 0$ *(ordre lexicographique), alors*

$$\exists m' \in J, \exists \sigma \in T^*, m \xrightarrow{\sigma} m' \wedge \nu(m') < \nu(m)$$

J est appelé le domaine de ν et K son noyau. L'ensemble des (J,K)-normes sur N est noté $\mathcal{N}(N,J,K)$; N est omis quand le contexte le permet.

Une norme est censée représenter la preuve en entier ou seulement un pas de preuve (qui est la concaténation de tous les pas). Pour que ce soit une preuve d'espace d'accueil, il faut que son domaine contienne l'ensemble d'accessibilité du réseau marqué considéré.

Corollaire I-1 *Si ν est une (J,K)-norme telle que $J \supseteq R(N;M_0)$, alors K est un espace d'accueil de $(N;M_0)$.*

Preuve On considère l'ordre défini sur $R(\Sigma)$ par $M < M'$ si $\nu(M) < \nu(M')$ et on applique le théorème I-1. □

La généralisation des normes qui sont des combinaisons linéaires des marquages de places, donne des vecteurs de marquages de places. L'intérêt est de ne plus avoir de coefficients à gérer mais un ordre, ce qui facilite les choses lors de la composition: insérer une place dans un ordre est plus simple que translater les coefficients ou trouver un coefficient intermédiaire.

Définition I-4 (Norme linéaire) *Une norme ν est dite linéaire s'il existe $\{p_1,\ldots,p_k\} \subseteq P$, tel que $\nu(m) = (m(p_1),\ldots,m(p_k))$. On note $\nu = (p_1,\ldots,p_k)$.*

Remarquons que si $\nu = (p_1,\ldots,p_k)$ est une (J,K)-norme linéaire telle que J contienne l'ensemble d'accessibilité, l'ensemble des marquages où $M(p_1) = \cdots = M(p_k) = 0$ est un espace d'accueil.

La notion de norme linéaire est une des raisons du choix de l'ordre lexicographique sur N^k, comme le montre l'exemple suivant.

Le réseau de la figure I-1 modélise le problème des lecteurs-écrivains avec priorité alternée. La place LA représente les processus en attente de lecture, EA ceux en attente d'écriture. Les places LX et EX représentent les processus respectivement en cours de lecture et en cours d'écriture. Quand PE est marquée, la priorité est aux écrivains, et quand PL est marquée, elle est aux lecteurs.

La transition *rl* est la demande de lecture, *dl* le début de lecture et *fl* la fin de lecture; les événements analogues pour l'écriture sont *re*, *de* et *fe*. *fp* est l'événement qui met fin à la priorité aux lecteurs. *tl* fait passer des lecteurs de l'état d'attente à l'état d'exécution, quand la priorité est aux lecteurs.

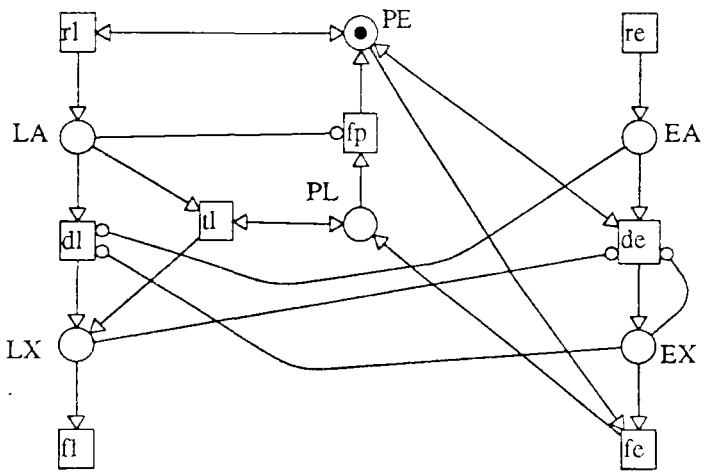


Figure I-1: Lecteurs-écrivains avec priorité alternée

Initialement, seule PE est marquée. Un écrivain en attente ne peut passer en exécution (de) que si la priorité est aux écrivains et qu'il n'y a pas de lecteurs ou d'écrivains en exécution. Quand un écrivain termine, la priorité passe aux lecteurs. S'il n'y a pas de lecteurs en attente, la priorité repasse aux écrivains. Un lecteur en attente ne peut commencer l'exécution que si la priorité est aux lecteurs, ou que la priorité est aux écrivains et qu'il n'y a pas d'écrivains en attente ou en exécution.

Nous allons montrer au moyen de la norme linéaire $\nu = (LA, EA, EX, PL, LX)$ que l'état initial est un état d'accueil. Les invariants du réseau sont les suivants:

$$M(PE) + M(PL) = 1 \quad \text{(I.1)}$$

$$M(EX) \leq 1 \quad \text{(I.2)}$$

$$M(EX) = 1 \Rightarrow M(PE) = 1 \quad \text{(I.3)}$$

L'invariant I.1 signifie que la priorité est soit aux écrivains, soit aux lecteurs; l'invariant I.2 exprime qu'il y a au plus un écrivain en exécution; l'invariant I.3 dit que, quand il y a un écrivain en exécution, alors la priorité est aux écrivains. Le premier invariant est linéaire et déduit du flot $PE+PL$, les deux autres se montrent par induction. Notons E l'ensemble des marquages vérifiant ces invariants (et donc qui contient tous les marquages accessibles). Nous allons montrer que ν est une $(E, \{M_0\})$-norme.

- $\nu(M) = 0 \Leftrightarrow M(LA) = M(EA) = M(EX) = M(PL) = M(LX) = 0 \Leftrightarrow M = M_0$, car $M(PL) = 0 \Rightarrow M(PE) = 1$ d'après l'invariant I.1.

- Si $\nu(M) > 0$, alors il existe $p \in \{LA, EA, EX, PL, LX\}$ telle que $M(p) > 0$. On distingue donc cinq cas, en supposant à chaque fois que les places de poids inférieur sont vides:

  1. Si $M(LX) > 0$, on franchit fl.
  2. Si $M(PL) > 0$ et $M(LX) = 0$: si $M(LA) = 0$, on franchit fp, sinon on franchit tl. Le dernier cas diminue le marquage de LA et augmente celui de LX, mais comme on adopte l'ordre lexicographique ν diminue.
  3. Si $M(EX) > 0$ et $M(PL) = M(LX) = 0$, on franchit fe.
  4. Si $M(EA) > 0$ et $M(EX) = M(PL) = M(LX) = 0$, on franchit de.
  5. Si $M(LA) > 0$ et $M(EA) = M(EX) = M(PL) = M(LX) = 0$, on franchit dl.

I-3 Composition de normes

L'intérêt de définir des compositions de norme est double:

- Pour un système donné $(N; M_0)$, décomposer hiérarchiquement la vérification d'un espace d'accueil E en plusieurs "sous-vérifications," au moyen d'un ensemble de normes $(\nu_i)_{i=1,n}$ qui, composées d'une certaine façon, donnent une norme associée à E.

- Pour un système $(N; M_0)$, lui-même construit hiérarchiquement par composition d'un ensemble de sous-systèmes $(N_i; M_{0i})_{i=1,n}$ ayant chacun une norme ν_i, construire une norme ν sur $(N; M_0)$ par composition des $(\nu_i)_{i=1,n}$.

Dans cette section, nous définissons deux opérations de compositions: l'ordonnancement et la somme de deux normes. La section suivante présente une méthode d'utilisation illustrée d'exemples.

La première opération de composition de norme correspond à la preuve par raffinements successifs ou par disjonction de cas. On a une norme ν_1 qui prouve qu'on peut aller de J_0 vers J_1, et une norme ν_2 qui prouve qu'on peut aller de J_1 vers J_2: on construit une norme ν qui prouve qu'on peut aller de J_0 vers J_2. Cette opération justifie le choix de l'ordre lexicographique sur $\mathbf{N}^k$ (cf. la preuve).

Proposition I-1 (Ordonnancement de normes) *Si $\nu_i \in \mathcal{N}(N, J_{i-1}, J_i)$ pour $i = 1, 2$ sont deux normes telles que $\nu_i : \mathbf{N}^P \to \mathbf{N}^{k_i}$, alors la fonction $\nu : \mathbf{N}^P \to \mathbf{N}^{k_1+k_2}$ définie par*

$$\nu(m) = (\nu_1(m), \nu_2(m))$$

est une (J_0, J_2)-norme sur N. On note $\nu = (\nu_1, \nu_2)$.

Preuve Il faut vérifier les conditions de la définition d'une norme.
D'abord $J_0 \supseteq J_1 \supseteq J_2$ et pour tout $m \in J_0$, $\nu(m) = 0$ ssi $\nu_1(m) = 0$ et $\nu_2(m) = 0$, c-à-d $m \in J_1 \cap J_2 = J_2$.

Il reste à prouver qu'on peut faire décroître ν si $\nu(m) > 0$. Si $m \in J_0$ et $\nu(m) > 0$, on distingue deux cas:

1. $\nu_1(m) > 0$: comme ν_1 est une (J_0, J_1)-norme, il existe $m' \in J_0$, tel que $m \xrightarrow{\sigma} m'$ et $\nu_1(m') < \nu_1(m)$. On en déduit $\nu(m') < \nu(m)$, puisque $<$ est l'ordre lexicographique (peu importe la relation de $\nu_2(m)$ et $\nu_2(m')$).

2. $\nu_1(m) = 0$ et $\nu_2(m) > 0$. $\nu_1(m) = 0$ entraîne $m \in J_1$. ν_2 étant une (J_1, J_2)-norme, il existe $m' \in J_1$, $m \xrightarrow{\sigma} m'$ et $\nu_2(m') < \nu_2(m)$. Comme $m' \in J_1$, $\nu_1(m') = 0$ et donc $\nu(m') < \nu(m)$.

□

La deuxième opération de composition est une somme de normes et correspond par rapport à la combinaison de marquages de places à donner le même coefficient à deux places: on veut composer deux normes sans les ordonner. Cette opération ne peut se faire que pour des normes indépendantes, c-à-d telles que la diminution de l'une n'augmente pas l'autre.

Définition I-5 (Normes indépendantes) *Deux normes ν_1 et ν_2 de même domaine J sont dites indépendantes si pour tout $\{i, j\} = \{1, 2\}$ on a*

$$\begin{array}{c} m \in J \wedge \nu_i(m) > 0 \\ \Downarrow \\ \exists m' \in J, \exists \sigma, [(m \xrightarrow{\sigma} m') \wedge (\nu_i(m') < \nu_i(m)) \wedge (\nu_j(m') \leq \nu_j(m))] \end{array}$$

Proposition I-2 (Somme de normes) *Si deux normes $\nu_i \in \mathcal{N}(N, J, K_i)$ sont indépendantes, alors $\nu = \nu_1 + \nu_2$ est une $(J, K_1 \cap K_2)$-norme. La somme sur $\bigcup_{k \in \mathbf{N}} \mathbf{N}^k$ est définie par (si $k \geq k'$)*

$$(a_1, \ldots, a_k) + (b_1, \ldots, b_{k'}) = (a_1, \ldots, a_{k-k'}, a_{k-k'+1} + b_1, \ldots, a_k + b_{k'})$$

Preuve Si $m \in J$, $\nu(m) = 0 \Leftrightarrow (\nu_1(m) = 0 \wedge \nu_2(m) = 0) \Leftrightarrow m \in K_1 \cap K_2$.

Si $\nu(m) > 0$ pour $m \in J$, il existe i tel que $\nu_i(m) > 0$. Alors il existe $m' \in J$, tel que $m \xrightarrow{\sigma} m'$, $\nu_i(m') < \nu_i(m)$, et $\nu_j(m') \leq \nu_j(m)$.

On en déduit $\nu(m') < \nu(m)$. □

Quand $J \supseteq R(\Sigma)$, alors K_1 et K_2 sont des espaces d'accueil; si en plus, la somme des deux normes est définie, $K_1 \cap K_2$ est aussi un espace d'accueil. Donc une condition suffisante pour que l'intersection de deux espaces d'accueil soit un espace d'accueil, est que la somme de leurs normes associées existe.

Ainsi cette opération fournit une méthode de vérification que l'intersection de deux espaces d'accueil en est un. (Rappelons que la propriété d'espace d'accueil est stable par réunion mais non par intersection.)

I-4 Vérification de normes

Dans [23] et [12], nous avons présenté une étude de cas de conception descendante de protocole, accompagnée d'une preuve descendante d'espace d'accueil. L'objectif de cette étude de cas était d'illustrer une méthode de *réutilisation de preuve* dans la conception progressive de systèmes.

La réutilisation de preuve, dans la conception hiérarchique, signifie qu'on tente de prouver des propriétés du système de niveau n en s'appuyant sur les preuves des sous-systèmes de niveau $n-1$.

Dans le cas des normes, le problème est le suivant: on a deux systèmes (N_i, M_{0i}) ayant chacun une norme ν_i; on obtient $(N; M_0)$ par composition de $(N_1; M_{01})$ et $(N_2; M_{02})$, et on voudrait savoir s'il existe une norme ν sur $(N; M_0)$ qui soit la composition de ν_1 et ν_2, ou de ν_1' et ν_2' qui sont obtenues par transformation des ν_i.

Plutôt que de poser des restrictions sur la nature de la composition des réseaux et des normes pour avoir cette propriété de "conservation", l'idée est alors d'associer à chaque norme un objet représentant sa preuve, et tester si cette preuve est encore applicable après la composition, et tester si la composition de normes est valide. Si oui, il faut que les preuves soient composables pour continuer ce processus.

Dans cette section, nous donnons un sens précis à une preuve de norme, et nous montrons que la preuve d'une composition de ν_1 et ν_2 s'obtient par une certaine combinaison de leurs preuves. La méthode d'utilisation de ces notions est illustrée par des exemples.

Nous définissons aussi une notion d'espace d'accueil plus précise, A-espace d'accueil, qui donne des informations sur une preuve possible de la propriété d'espace d'accueil. Cette notion apparaîtra dans les chapitres suivants sur le remplacement de sous-réseaux et la composition de réseaux.

I-4.1 Preuve de norme

On a vu que, fondamentalement, une norme est un ordre sur les éléments de J tel qu'il existe une séquence de n'importe quel élément vers un élément de rang strictement inférieur. Nous représentons alors la preuve d'une norme en regroupant les éléments admettant la même séquence qui fait décroître la norme.

Ainsi une preuve est un ensemble de couples (E, σ), où E est un sous-ensemble de J, et σ une séquence franchissable en tout marquage de E menant à un élément de rang strictement inférieur.

Définition I-6 (Preuve d'une norme) *Une preuve d'une (J,K)-norme ν est un ensemble inclus dans $(\mathcal{B}(J) \times T^*)$, noté V tel que*

- $\{E; \exists \sigma, (E, \sigma) \in V\}$ *est une partition de* $J \setminus K$
- *si* $(E, \sigma) \in V$ *alors* $\forall m \in E, \exists m' \in J, m \xrightarrow{\sigma} m' \wedge \nu(m') < \nu(m)$

L'ensemble des preuves de ν est noté $PR(\nu)$.

Une preuve de la norme ν de l'exemple des lecteurs-écrivains (figure I-1) est (pour abréger les notations, l'ensemble $\{M; \mathcal{P}(M)\}$ sera noté $\mathcal{P}(M)$):

$$\left\{ \begin{array}{l} (M(LX) > 0, fl), \\ (M(PL) > 0 \wedge M(LX) = M(LA) = 0, fp), \\ (M(PL) > 0 \wedge M(LX) = 0 \wedge M(LA) > 0, tl), \\ (M(EX) > 0 \wedge M(PL) = M(LX) = 0, fe), \\ (M(EA) > 0 \wedge M(EX) = M(PL) = M(LX) = 0, de), \\ (M(LA) > 0 \wedge M(EA) = M(EX) = M(PL) = M(LX) = 0, dl) \end{array} \right\}$$

I-4.2 Composition de preuves

Il est possible de construire une preuve de (ν_1, ν_2) et de $\nu_1 + \nu_2$ par une certaine combinaison de leurs preuves V_1 et V_2. En plus, il est possible, sur les preuves de ν_1 et ν_2, de tester une condition suffisante de l'indépendance de ν_1 et ν_2, et donc de l'existence de $\nu_1 + \nu_2$.

Dans le cas de l'ordonnancement de deux normes, une preuve de la composition est obtenue en prenant la réunion des deux preuves.

Proposition I-3 (Preuve de (ν_1, ν_2)) *Si $(\nu_i)_{i=1,2}$ sont deux normes telles que $\nu = (\nu_1, \nu_2)$ soit définie, alors*

$$\forall i \in \{1,2\}, V_i \in PR(\nu_i) \Rightarrow V = (V_1 \cup V_2) \subseteq PR(\nu)$$

Preuve On note $d(X) = \{E; \exists\sigma, (E,\sigma) \in X\}$. On suppose que ν_i est une (J_{i-1}, J_i)-norme. Puisque $d(V_i)$ est une partition de $J_{i-1} \setminus J_i$, et $J_0 \supseteq J_1 \supseteq J_2$, $d(V_1) \cup d(V_2)$ est une partition de $(J_0 \setminus J_1) \cup (J_1 \setminus J_2) = J_0 \setminus J_2$.

Il reste à montrer que si $(E,\sigma) \in V$, alors pour tout $m \in E$, $m \xrightarrow{\sigma} m'$ et $\nu(m') < \nu(m)$.

Si $(E,\sigma) \in V_1$, alors $m \xrightarrow{\sigma} m'$ tel que $\nu_1(m') < \nu_1(m)$ et donc $\nu(m') < \nu(m)$.

Si $(E,\sigma) \in V_2$, alors $m \xrightarrow{\sigma} m'$ tel que $\nu_1(m) = \nu_1(m') = 0$ et $\nu_2(m') < \nu_2(m)$: d'où $\nu(m') < \nu(m)$. □

La somme de deux normes n'est définie que si elles sont indépendantes. Une condition suffisante d'indépendance peut être testée sur des preuves de ν_1 et ν_2, et alors une preuve de $\nu = \nu_1 + \nu_2$ est obtenue à partir de celles de ν_1 et de ν_2.

Proposition I-4 (Preuve de $\nu_1 + \nu_2$) *Soit ν_1 et ν_2 deux normes telles que $\nu_i \in \mathcal{N}(N, J, K_i)$, et $V_i \in PR(\nu_i)$.*

Si pour tout $i \neq j$, $\forall (E,\sigma) \in V_i$, $\forall m \in E$, $m \xrightarrow{\sigma} m'$ tel que $\nu_j(m') \leq \nu_i(m)$ alors ν_1 et ν_2 sont indépendantes.

De plus, $V_i \cup \{(E \cap K_i, \sigma); (E,\sigma) \in V_j\} \in PR(\nu_1 + \nu_2)$.

Preuve Puisque $J \setminus K_i = \{m \in J; \nu_i(m) > 0\}$, et l'ensemble des E tels que $(E,\sigma) \in V_i$ est une partition de $J \setminus K_i$, l'indépendance de ν_1 et ν_2 découle immédiatement de la définition.

Prouvons, par exemple, que $V = V_1 \cup \{(E \cap K_1, \sigma); (E,\sigma) \in V_2\} \in PR(\nu_1 + \nu_2)$. $\{E; \exists\sigma, (E,\sigma) \in V_1\}$ est une partiton de $J \setminus K_1$ et $\{(E \cap K_1); (E,\sigma) \in V_2\}$ est une partition de $(J \setminus K_2) \cap K_1 = K_1 \setminus K_2$: donc $\{E; (E,\sigma) \in V\}$ est une partition de $(J \setminus K_1) \cup (K_1 \setminus K_2) = J \setminus (K_1 \cap K_2)$.

On vérifie facilement que si $(E,\sigma) \in V$, alors pour tout $m \in E$, $m \xrightarrow{\sigma} m'$ avec $\nu(m') < \nu(m)$, où $\nu = \nu_1 + \nu_2$. □

Exemples

Dans les exemples suivants, P_i est l'ensemble des places du réseau N_i, et $P = P_1 \cup P_2$.

Dans la figure I-2, $\nu_1 = (B,C)$ est une (J_1, K_1)-norme linéaire de N_1 où

$$J_1 = \mathbf{N}^{P_1} \bigwedge K_1 = \{M \in \mathbf{N}^{P_1}; M(B) = M(C) = 0\}$$

Une preuve V_1 de cette norme est

$$V_1 = \{(E_{11}, t_2), (E_{21}, t_3)\}$$

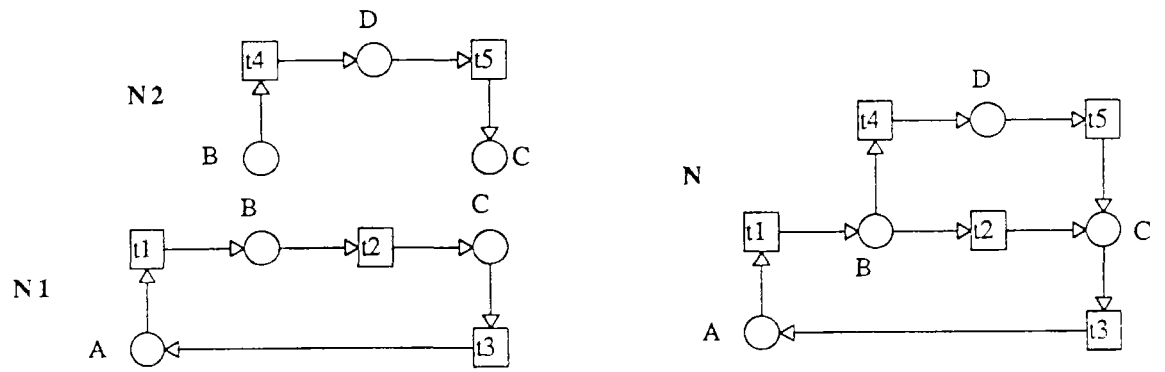


Figure I-2: Ordonnancement de normes associé à la composition de réseaux

où

$$E_{11} = \{M \in \mathbf{N}^{P_1}; M(B) > 0 \wedge M(C) = 0\} \bigwedge E_{21} = \{M \in \mathbf{N}^{P_1}; M(C) > 0\}$$

$\nu_2 = (D)$ est une (J_2, K_2)-norme de N_2 où

$$J_2 = \mathbf{N}^{P_2} \bigwedge K_2 = \{M \in \mathbf{N}^{P_2}; M(D) = 0\}$$

Une preuve V_2 de cette norme est

$$V_2 = \{(E_{12}, t_5)\}$$

où

$$E_{12} = \{M \in \mathbf{N}^{P_2}; M(D) > 0\}$$

On compose N_1 et N_2 par fusion de places et on obtient le réseau N. Dans N_1, on a construit une norme ν_1 et sa preuve montrant qu'il est possible de vider les places B et C. Dans N_2, on a procédé de même avec la norme ν_2 pour montrer qu'on peut vider D.

La question est alors de savoir s'il existe des transformations des preuves de ν_1 et ν_2 pour obtenir une norme ν' sur N avec sa preuve montrant qu'on peut vider les places B, C et D. Nous n'avons pas encore une réponse générale à ce problème, mais nous allons montrer que c'est possible dans cet exemple.

Nous montrons qu'il est d'abord possible de vider D en réutilisant ν_2 puis de vider B et C en réutilisant ν_1.

D'abord, on cherche une transformation de la preuve V_2 pour obtenir une norme sur N associée au vidage de la place D. Il est facile de vérifier que V_2' définie par

$$V_2' = \{(E, \sigma); \exists (G, \sigma) \in V_2 \wedge E = G\uparrow^P\}$$

est une preuve de la norme $\nu_2' : \mathbf{N}^P \to \mathbf{N}$ définie par $\nu_2'(M) = \nu_2(M{\downarrow_{P_2}})$ (il suffit de vérifier que les séquences restent franchissables et diminuent la norme). C'est alors une (J_2', K_2')-norme sur N où $J_2' = J_2{\uparrow^P}$ et $K_2' = K_2{\uparrow^P}$.

Maintenant, nous transformons V_1 en supposant que D est vide: on ne se contente pas d'étendre les marquages de V_1 à P, mais on prend leur intersection avec

$$F = \{M \in \mathbf{N}^P; M(D) = 0\} = K_2'$$

Ce qui donne

$$V_1' = \{(E, \sigma); \exists (G, \sigma) \in V_1 \wedge E = G{\uparrow^P} \cap F\}$$

On vérifie alors que c'est une preuve de $\nu_1' : \mathbf{N}^P \to \mathbf{N}$ définie par $\nu_1'(M) = \nu_1(M{\downarrow_{P_1}})$, qui est une (J_1', K_1')-norme sur N où $J_1' = J_1{\uparrow^P} \cap F$ et $K_1' = K_1{\uparrow^P} \cap F$.

Finalement, on a $K_2' = J_1'$, et donc, $\nu' = (\nu_2', \nu_1')$ est définie, et une de ces preuves est obtenue par $V' = V_1' \cup V_2'$.

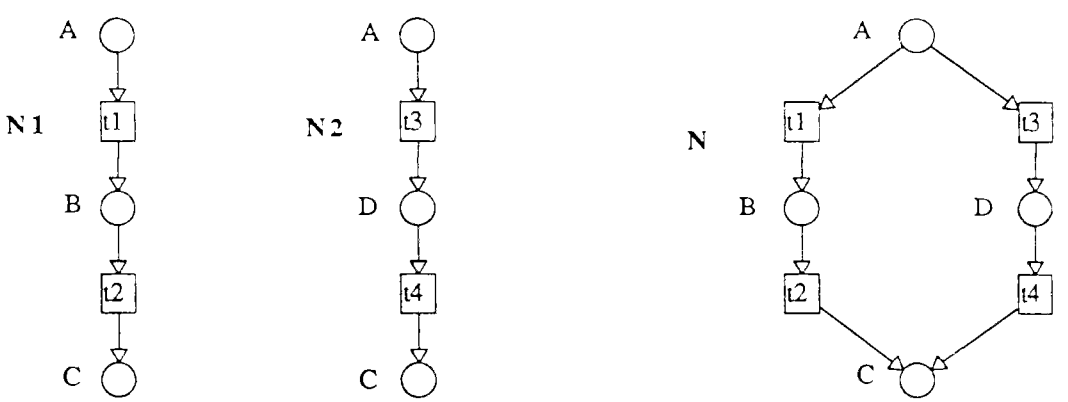


Figure I-3: Somme de normes associée à la composition de normes

Le deuxième exemple (figure I-3) illustre l'utilisation de la somme de normes. $\nu_1 = (A, B)$ est une (J_1, K_1)-norme sur N_1, où

$$J_1 = \mathbf{N}^{P_1} \bigwedge K_1 = \{M \in \mathbf{N}^{P_1}; M(A) = M(B) = 0\}$$

Une preuve de cette norme est

$$V_1 = \{(E_{11}, t_1), (E_{21}, t_2)\}$$

où

$$E_{11} = \{M \in \mathbf{N}^{P_1}; M(A) > 0 \wedge M(B) = 0\} \bigwedge E_{21} = \{M \in \mathbf{N}^{P_1}; M(B) > 0\}$$

$\nu_2 = (A, D)$ est une (J_2, K_2)-norme sur N_2, où

$$J_2 = \mathbf{N}^{P_2} \bigwedge K_2 = \{M \in \mathbf{N}^{P_2}; M(A) = M(D) = 0\}$$

Une preuve de cette norme est

$$V_2 = \{(E_{12}, t_3), (E_{22}, t_4)\}$$

où

$$E_{12} = \{M \in \mathbf{N}^{P_2}; M(A) > 0 \wedge M(D) = 0\} \bigwedge E_{22} = \{M \in \mathbf{N}^{P_2}; M(D) > 0\}$$

La composition par fusion de places de N_1 et N_2 donne le réseau N. Là aussi on voit aisément que

$$V_i' = \{(E, \sigma); \exists (G, \sigma) \in V_i \wedge E = G\uparrow^P\}$$

est une preuve de la norme $\nu_i' : \mathbf{N}^P$ définie par $\nu_i'(M) = \nu_i(M\downarrow_{P_i})$ appartenant à $\mathcal{N}(N, J_i', K_i')$, où $J_i' = J_i\uparrow^P$ et $K_i' = K_i\uparrow^P$. $J_1' = J_2'$ et la condition d'indépendance peut être testée et vérifiée sur V_1' et V_2': donc $\nu' = \nu_1' + \nu_2'$ est une (J', K')-norme sur N, avec $J' = J_1' = J_2' = \mathbf{N}^P$ et

$$K' = K_1' \cap K_2' = \{M \in \mathbf{N}^P; M(A) = M(B) = M(D) = 0\}$$

Une preuve de ν' est construite à partir de V_1' et V_2' comme indiqué dans la proposition I-4.

I-4.3 A-espace d'accueil

Quand un réseau admet un espace d'accueil, il est assez indifférent de savoir comment on l'atteint tant qu'on s'intéresse aux propriétés du réseau isolé. Mais si on compose le réseau avec un environnement ou si on le raffine, on souhaite savoir ce qui est conservé de cette propriété d'accueil, et alors il est utile de connaître l'influence de certaines places ou certaines transitions sur la preuve.

Nous définissons la notion de A-espace d'accueil qui précise qu'il est possible de se restreindre à des transitions de A pour atteindre cet espace, et par conséquent la propriété d'accueil est indépendante du marquage des places adjacentes à $T \setminus A$.

Ce n'est pas une nouvelle notion d'espace d'accueil mais une indication sur une preuve possible d'un espace d'accueil. Elle sera souvent utilisée dans les chapitres suivants.

Définition I-7 (*A*-espace d'accueil) *H est un A-espace d'accueil de $(N; M_0)$ où $A \subseteq T$ ssi c'est un espace d'accueil accessible en franchissant uniquement des transitions de A, i.e.,*

$$\forall M \in R(N; M_0), \exists w \in A^*, M \xrightarrow{w} M' \in H$$

Noter la place de l'adverbe "uniquement:" ...accessible en franchissant uniquement..., et non pas ...uniquement accessible en franchissant... Autrement dit, un espace d'accueil est un A-espace d'accueil, *s'il existe* une preuve de norme associée à cet espace ne contenant que des séquences dans A^*. Ceci est exprimé par la notion de (J, A, K)-norme.

Définition I-8 *Une (J, A, K)-norme, où $A \subseteq T$, est une (J, K)-norme telle qu'il existe $V \subseteq PR(\nu)$, et $\forall (E, \sigma) \in V$, $\sigma \in A^*$.*

Conclusion

Nous avons défini des normes, munies d'opérations de composition, qui sont des nouveaux outils de vérification modulaire et hiérarchique des espaces d'accueil. Nous avons associé aux normes des objets représentant leurs preuves, telles que, la preuve de la composition de deux normes soit obtenue par combinaison de leurs preuves respectives.

L'objectif à long terme de ce travail est de formaliser la méthode de réutilisation de preuve esquissée et illustrée par des exemples dans la quatrième section: nous avons montré qu'il était possible d'obtenir une norme d'un réseau construit par composition de deux réseaux ayant chacun une norme ν_i, en transformant leurs preuves et en les composant.

Ceci devrait déboucher sur une mise en œuvre d'une méthode de "verification assistée" d'espace d'accueil: un système expert serait capable d'enregistrer les preuves (ce qui suppose qu'on sache les représenter en machine), et lors d'une transformation de modèles, de proposer une transformation de preuve et de vérifier sa validité dans le nouveau modèle (il pourraitt aussi se faire indiquer la transformation par l'utilisateur). Ces représentations devraient être étendues aux réseaux colorés paramétrés pour une plus grande applicabilité.

Chapitre II

Raffinements de sous-réseaux

Introduction

Le raffinement et l'abstraction sont des méthodes complémentaires dans la conception et l'analyse des systèmes. La conception hiérarchique consiste à partir d'un modèle abstrait qu'on raffine progressivement en remplaçant certaines parties du modèle par des sous-modèles plus détaillés. L'opération inverse (abstraction) est utile quand on veut analyser et comprendre un système déjà réalisé, et alors on construit un modèle abstrait de son comportement. Une telle méthode de conception hiérarchique doit s'appuyer sur une méthode d'analyse hiérarchique: si le modèle $\mathcal{M}$ est transformé par une opération bien définie *op*, les propriétés de $op(\mathcal{M})$ devraient être déduites de celles de $\mathcal{M}$ et *op*.

L'opération étudiée dans ce chapitre est le remplacement d'un sous-réseau à frontière de places (sous-réseau ouvert) par un réseau: si le réseau de remplacement est plus détaillé que le sous-réseau c'est un raffinement, sinon c'est une abstraction. Si $N\,[N_1 \leftarrow N_2]$ est le réseau obtenu après remplacement du sous-réseau N_1 par le réseau N_2 dans N, nous voulons déduire $N\,[N_1 \leftarrow N_2] \equiv N$ de $N_1 \equiv' N_2$, pour certaines relations d'équivalence $\equiv$ et $\equiv'$.

On rencontre naturellement les sous-réseaux ouverts quand on modélise un système distribué comme un ensemble d'acteurs communicant par envoi de messages. Par exemple, cette méthode d'analyse peut être appliquée aux réseaux HOOD[22] qui possèdent une interface de places, ou à la méthode de conception hiérarchique de Girault[24,25] fondée sur les "acteurs abstraits" où un serveur abstrait est un réseau ayant une interface de places, et les raffinements consistent à remplacer de tels réseaux.

Un sous-réseau ouvert est engendré par un sous-ensemble de transitions, alors qu'un sous-réseau fermé (ie à frontière de transitions) est engendré par un sous-

ensemble de places.

Le remplacement de sous-réseaux à frontière de transitions et la composition de réseaux par fusion de transitions ont été largement étudiés au moyen de notions d'équivalences fondées sur l'étiquetage des transitions et l'observation des langages des réseaux (André[3], Baumgarten[6], Bourguet-Rouget[9], Valmari[59], Vogler[61] et voir Pomello[46] pour une synthèse de ces notions d'équivalence). Ces notions sont la plupart du temps inspirées des modèles algébriques tels que CCS et CSP, et jouissent de propriétés mathématiques élégantes et maniables.

Le remplacement de sous-réseaux à frontière de places et la composition par fusion de places sont un peu plus laborieux et ne sont pas des opérations libres puisque des restrictions sont nécessaires pour obtenir des propriétés de fermeture ou assurer l'existence de certaines constructions mathématiques (pour obtenir, dans le cadre de la théorie des catégories, que la composition par fusion de places soit une somme, Winskel[66] considère des réseaux 1-bornés).

Le raffinement de transition est un cas particulier du remplacement d'un sous-réseau à frontière de places: on remplace le sous-réseau engendré par cette transition. Cette opération a été étudiée par Valette[57], Suzuki et Murata[55], Vogler[62], van Glabbeek et Goltz[27], soit en considérant la conservation de propriétés soit en considérant des notions d'équivalence qui sont des congruences pour de tels raffinements.

Nous ne cherchons pas à définir une notion d'équivalence conservée par de tels raffinements mais une équivalence entre un réseau et son raffinement. Dans [60], Vogler remplace un sous-réseau, engendré par une transition, par des réseaux particuliers appelés "modules": nous considérons des opérations de remplacement plus générales et la notion d'équivalence de [60] est inspirée de celle d'André[3] et donc fondée sur l'étiquetage des transitions.

Dans la première section, nous donnons les motivations des différentes notions à travers un exemple, avant de définir les objets de base étudiés. Le point important est que nous ne considérons pas de simples réseaux P/T, mais des réseaux où on distingue un ensemble de places appelées places d'interface, et un ensemble donné de marquages de "places internes" appelés états stables: ces objets sont appelés réseaux à interface ouverte (open interface nets ou OI-réseaux). Ces OI-réseaux n'ayant pas assez de propriétés comportementales, nous définissons les systèmes à interface ouverte qui sont des OI-réseaux ayant un marquage d'interface pour lequel l'ensemble des états stables est un espace d'accueil.

Dans la deuxième section, nous définissons deux équivalences de transformation d'états: quand on raffine des actions, on change le niveau d'abstraction des événements, et si on recherche une équivalence entre un réseau et son raffinement,

il est plus approprié de comparer les transformations d'états effectuées par l'action avec celles effectuées par son raffinement plutôt que de comparer leurs comportements exprimés en termes de transitions observables (qui est indiqué quand on raffine des états et qu'on remplace des sous-réseaux fermés). Ainsi on est amené à étiqueter les places et à étudier des équivalences de transformation d'états[48].

L'équivalence de fonctionalité stable (SF-éq) est définie sur les OI-systèmes, et la conservation (dans un sens restreint) de blocage et de la propriété d'espace d'accueil est montrée. L'SF-équivalence étant insuffisante pour faire du remplacement de sous-réseaux, une équivalence plus forte, l'équivalence de transformation d'états stables (SST-éq), est définie sur les OI-réseaux. Ces deux notions d'équivalences sont des bisimulations définies uniquement sur les états stables.

En fait, ces équivalences suivent la ligne de recherche explorée par l'équipe italienne de Milan[19] où l'EF-équivalence est définie sur les systèmes "S-observables", et par Pomello et Simone[47] où un préordre et une équivalence de transformations d'états sont définis. Les états stables sont une généralisation des marquages observables. Un résultat similaire à celui visé ici a été établi pour les réseaux SA (superposed automata) 1-bornés et l'EF-équivalence[20] (le raffinement fonctionnel des réseaux SA 1-bornés est un cas particulier de remplacement de sous-réseaux à frontière de places). La principale différence entre l'SST-équivalence et l'EF-équivalence est la façon dont la simulation d'une transformation d'état est faite (cf. définitionII-6); et l'EF-équivalence est définie au moyen d'un isomorphisme entre des algèbres (engendrées par les marquages observables) alors que l'SST-équivalence est définie au moyen d'une bisimulation.

Dans la troisième section, l'opération de remplacement est définie sur les OI-réseaux et les OI-systèmes. Notons $rep(N)$ le résultat du remplacement d'un sous-réseau de N par un réseau SST-équivalent. Pour un OI-réseau OIN, $rep(N) \equiv_{SST} OIN$, mais l'ensemble des OI-systèmes n'est pas fermé par rep. Une opération restreinte—remplacement robuste (rep_r)—est définie telle que l'ensemble des OI-systèmes soit fermé par rep_r, et $rep_r(OIS) \equiv_{SF} OIS$ si OIS est un OI-système.

Dans la quatrième section, une opération d'expansion d'interface est définie sur les OI-systèmes, donnant lieu à la définition d'un préordre associé à l'SST-équivalence; quelques conservations de propriétés sont établies.

Dans la cinquième section, un exemple illustre l'utilisation de ces notions dans la conception hiérarchique de protocoles.

II-1 Réseaux à interface ouverte

II-1.1 Exemple d'introduction

Considérons le modèle client-serveur de base suivant (Figure II-1): un client oisif (CI) envoie une requête (R) et attend l'acquittement (CW); il y a n clients dans le système. Sur réception d'une requête le serveur oisif (SI) exécute la requête (SX); puis il envoie un acquittement au client (A) et redevient oisif. Quand le client reçoit l'acquittement il devient oisif.

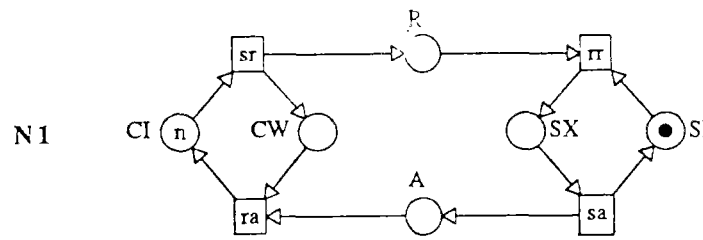


Figure II-1: Modèle Client-Serveur

On ajoute un tampon pour recevoir les requêtes quand le serveur est occupé et uniquement dans ce cas: on obtient le réseau $N2$ de la figure II-2. BB est le nombre de requêtes tamponnées et FB est le nombre de places vides. Le serveur consulte le tampon (SC) avant de redevenir oisif.

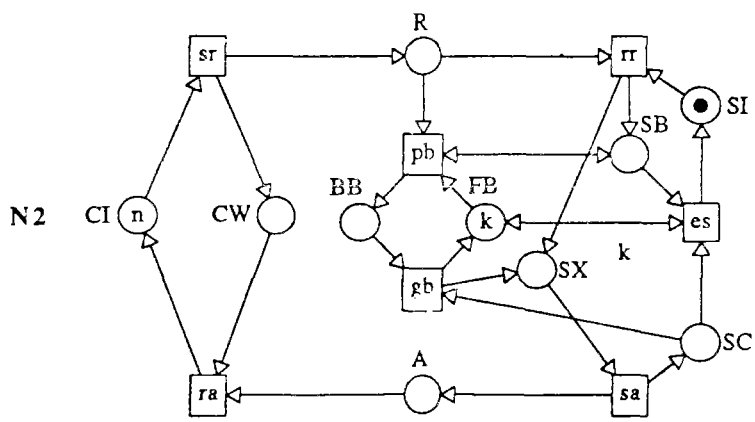


Figure II-2: Ajout d'un tampon

Si on se restreint à l'étiquetage des transitions et aux sous-réseaux à frontière de transitions, on est amené à regarder $N2$ comme le résultat du remplacement

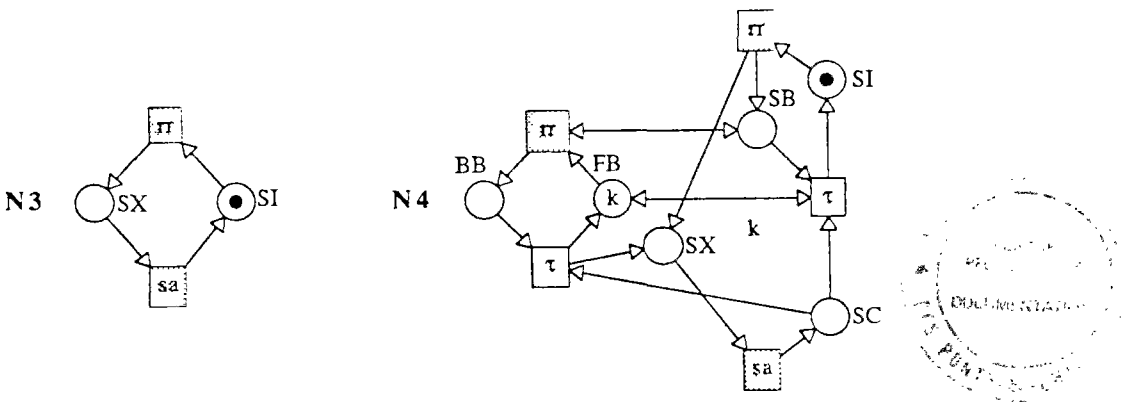


Figure II-3: Remplacement d'un sous-réseau à frontières de transitions

du sous-réseau $N3$ engendré par les places SX et SI (figure II-3) par le réseau $N4$ (les transitions rr et pb ont la même étiquette rr).

Dans $N4$ la séquence $rr.rr$ est franchissable alors qu'elle ne l'est pas dans $N3$ (idem pour $N1$ et $N2$); mais toutes les notions d'équivalence fondées sur les événements exigent au moins l'égalité des langages: donc ces réseaux ne vérifient aucune de ces équivalences. Cet argument n'est plus valable si on inclut la transition sr dans le réseau remplacé: mais nous ne voulons pas inclure une transition du client quand nous remplaçons le serveur.

C'est pourquoi nous définissons des équivalences fondées sur les transformations d'états et considérons cette transformation de réseau comme le remplacement de $N5$, le sous-réseau engendré par $\{rr, sa\}$, par $N6$ (figure II-4) car ces deux réseaux effectuent la même transformation sur $\{R, A\}$: une requête consommée de R est transformée en un acquitement délivré dans A.

Par dualité avec le paradigme de l'étiquetage des événements et l'observation des séquences d'événements, on pourrait penser étiqueter les places et observer l'évolution des marquages des places observables, et exiger que le réseau de remplacement présente la même évolution de marquages que le sous-réseau remplacé. Dans notre exemple les places observables—qu'on appelle places d'interface—sont R et A.

Ce paradigme observationnel est trop restrictif: le réseau $N7$ (figure II-4) est une abstraction de $N5$ et il semble raisonnable de vouloir remplacer $N5$ par $N7$; mais ces deux réseaux ne présentent pas la même évolution de marquages sur $\{R, A\}$: dans $N5$, à partir de l'état $M(R) = 1, M(A) = 0$, on peut atteindre l'état $M(R) = M(A) = 0$, et ceci est impossible dans $N7$ à cause de la différence d'atomicité entre ces deux réseaux. Cependant, si on considère uniquement les états de $N5$ vérifiant $M(SX) = 0$ alors on observe les mêmes transformations

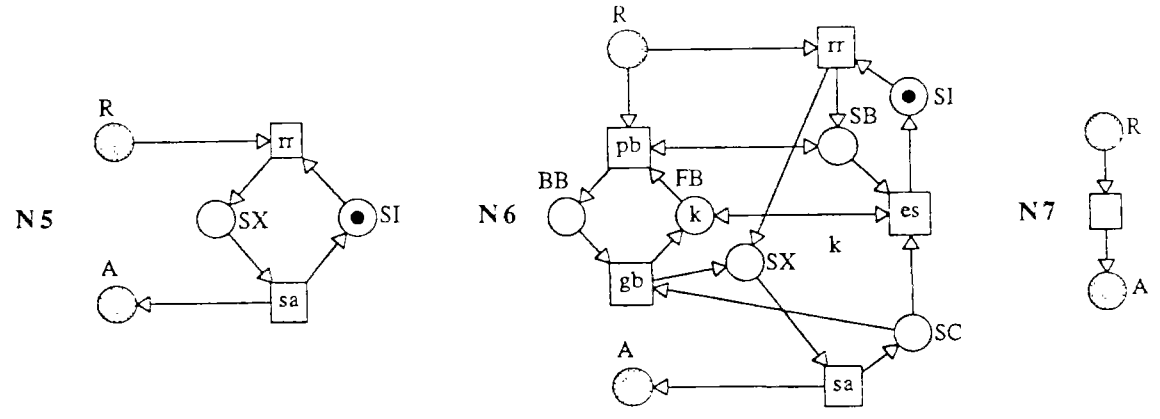


Figure II-4: Remplacement d'un sous-réseau à frontières de places

d'états sur $\{R, A\}$: nous appellerons de tels états des états stables; dans $N2$, les états stables sont ceux vérifiant $M(SI) = 1, M(SC) = M(SX) = M(BB) = 0, M(FB) = k$.

Il existe une séquence infinie de $N2$ qui n'amène jamais le système dans un état stable mais il est toujours possible d'atteindre un tel état. Donc la comparaison de deux réseaux par rapport à leurs états stables n'est pas une notion observationnelle puisqu'elle ne peut pas être vérifiée par un observateur extérieur: l'équivalence que nous définissons, fondée sur les transformations d'états stables (SST-éq), doit être considérée comme une méthode de preuve et d'analyse hiérarchique de réseaux conçus hiérarchiquement et non une notion observationnelle.

Les états stables sont les seuls où l'état de l'interface est définitif et significatif: ils correspondent à des états internes où aucune "action observable" (c-à-d, dont les conséquences sur l'interface sont observables) n'est en cours.

II-1.2 OI-réseaux

Nous avons exposé l'intuition qui sous-tend la définition d'une classe de réseaux avec un ensemble de places distinguées—les places d'interface, et un ensemble d'états distingués—les états stables. Donc les objets considérés ne sont pas de simples réseaux de Petri mais des objets de cette classe appelée réseaux à interface ouverte. Cette "interface" est soit la frontière d'un sous-réseau soit les places "observables" de tout un système.

Définition II-1 (Réseaux à interface ouverte) *Un* réseau à interface ouverte *(OI-réseau) est un quadruplet* $OIN = (N; ITF; STB; M_0)$ *où*

- $N = (P, T; W)$ *est un réseau* P/T

- $ITF \subseteq P$ *est appelé l'ensemble des places d'interface, et* $INR = P \setminus ITF$ *est appelé l'ensemble des places internes*
- $STB \subseteq \mathrm{N}^{INR}$ *est appelé l'ensemble des états stables*
- $M_0 \in STB$

L'ensemble des réseaux à interface ouverte est noté $\mathcal{OIN}$*, et* $\mathcal{OIN}_{ITF}$ *désigne l'ensemble des réseaux dont l'interface est* ITF*.*

Un réseau à interface ouverte est une structure sans aucune propriété comportementale: les places d'interface ne sont pas marquées et les états stables sont un ensemble arbitraire d'états internes.

De façon analogue à la définition des systèmes P/T qui sont des réseaux P/T avec un marquage et un comportement, nous définissons les systèmes à interface ouverte qui sont des réseaux à interface ouverte avec un marquage de l'interface et une propriété comportementale des états stables. Ceci permettra de séparer ce qui est structurel de ce qui est comportemental dans les définitions des équivalences et de l'opération de remplacement.

Définition II-2 (Système à interface ouverte) *Un* système à interface ouverte *(OI-système) est un couple* $OIS = (OIN; m_0)$ *où*

- $OIN = (N; ITF; STB; M_0) \in \mathcal{OIN}$
- $m_0 \in \mathrm{N}^{ITF}$
- $STB\uparrow^P$ *est un espace d'accueil de* $(N; M_0 + m_0)$

$\mathcal{OIS}$ *désigne l'ensemble des systèmes à interface ouverte, et* $\mathcal{OIS}_{ITF}$ *l'ensemble des systèmes dont l'interface est* ITF*.*

Définition II-3 (Etats stables accessibles) *Soit* $OIS = (OIN; m_0)$ *un OI-système.* L'ensemble des états stables accessibles *de* OIS *est*

$$RSS(OIS) = \{M \in R(N; M_0 + m_0); M{\downarrow_{INR}} \in STB\}$$

Dans un système à interface ouverte, il est toujours possible d'atteindre un état stable et il est donc sensé de comparer les transformations d'états stables de tels systèmes.

La Figure II-5 montre la différence entre un OI-réseau et un OI-système . OIN_1 et OIN_2 sont dans $\mathcal{OIN}$. Pour tout marquage m_0 de l'interface, $(OIN_1; m_0)$ est dans $\mathcal{OIS}$, mais $(OIN_2; m_0)$ est un IO-système seulement si $m_0(p_1) \leq m_0(p_2)$ ($STB_2 = \{M; M(p_5) = 0\}$: si t_2 est franchie à partir de m_0 tel que $m_0(p_1) = 1$ et $m_0(p_2) = 0$, il est impossible d'atteindre un état stable.)

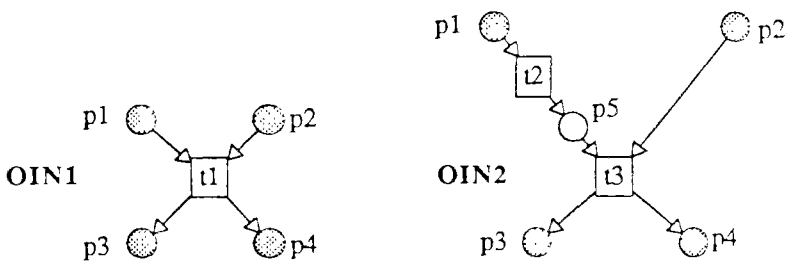


Figure II-5: Différence entre un OI-réseau et un OI-système

II-2 Equivalences

Dans cette section deux notions d'équivalence sont définies: SF-équivalence sur les OI-systèmes et SST-équivalence sur les OI-réseaux.

II-2.1 SF-équivalence

Le comportement d'un OI-système est caractérisé par ses états stables et la relation de transition entre ces états: c'est ce que nous appelons ses fonctionalités stables. C'est pourquoi nous définissons l'équivalence entre deux OI-systèmes au moyen d'une bisimulation qui est une correspondance entre leurs états stables respectifs, telle que des états stables correspondants aient même partie observable (même marquage de l'interface) et permettent les mêmes transitions vers d'autres états stables équivalents.

Définition II-4 (SF-équivalence sur $\mathcal{OIS}$) *Soit $OIS_i \in \mathcal{OIS}_{ITF}$, $i = 1,2$. Alors $\mathcal{R}_{SF} \subseteq RSS_1 \times RSS_2$ (RSS est défini dans Définition II-3) est une* bisimulation de fonctionalité stable *(SF-bisimulation) ssi:*

1. $M_{01} + m_{01}\mathcal{R}_{SF}M_{02} + m_{02}$
2. $M_1\mathcal{R}_{SF}M_2 \Rightarrow$
   - **a)** $M_1{\downarrow_{ITF}} = M_2{\downarrow_{ITF}}$
   - **b)** $M_1 \xrightarrow{\sigma_1} M_1' \in RSS_1 \Rightarrow \exists\sigma_2,\ M_2 \xrightarrow{\sigma_2} M_2' \in RSS_2,$ *et* $M_1'\mathcal{R}_{SF}M_2'$
   - **c)** *idem que b) mais en permutant 1 et 2*

On dit que OIS_1 et OIS_2 sont SF-équivalents et l'on note $OIS_1 \equiv_{SF} OIS_2$ ssi il existe une SF-bisimulation de OIS_1 vers OIS_2.

$\mathcal{R}_{SF}$ n'est pas unique en général, mais nécessairement, $dom(\mathcal{R}_{SF}) = RSS_1$ et $cod(\mathcal{R}_{SF}) = RSS_2$.

Nous allons maintenant montrer que l'SF-équivalence conserve des propriétés de blocage et d'espace d'accueil.

Un blocage dans la terminologie classique est un état qui n'a pas de successeur. Comme dans les OI-systèmes on ne s'intéresse qu'aux états stables, une première définition du blocage d'un OI-système serait un état stable à partir duquel il est impossible d'atteindre un autre état stable; mais avec une telle définition, nous n'obtenons pas la conservation de blocage par l'SF-équivalence (voir exemple ci-dessous).

Nous définissons alors un I-blocage comme étant un état stable à partir duquel il est impossible d'atteindre un état stable montrant un marquage d'interface différent.

Donc un I-blocage d'un OI-système peut être un blocage actif: il peut y avoir des séquences franchissables qui modifient le marquage de l'interface ou qui mènent à un état stable différent mais ayant le même marquage d'interface, mais il n'y a pas de séquence menant à un état stable avec un marquage d'interface différent.

Définition II-5 (I-blocage des OI-systèmes) *Un* $OIS \in \mathcal{OIS}$ a un I-blocage *ssi*

$$\exists M \in RSS(OIS), \forall \sigma \in T^*, M \xrightarrow{\sigma} M' \in RSS(OIS) \Rightarrow M{\downarrow_{ITF}} = M'{\downarrow_{ITF}}$$

Un tel marquage M *est appelé un I-blocage de* OIS.

Dans la figure II-6, OIS_1 et OIS_2 sont SF-éq, où $ITF = \{p_0, p_1\}$ et $STB_2 = \{M; M(p_2) = 0\}$.

Dans OIS_1, une fois qu'on a atteint le marquage stable M tel que $M(p_0) = 0$ et $M(p_1) = 1$, plus aucune transition n'est franchissable. Donc ce marquage est un I-blocage de OIS_1.

Dans OIS_2, à partir de M tel que $M(p_0) = M(p_2) = M(p_3) = 0$ et $M(p_1) = 1$, on peut atteindre M^k tel que $M(p_0) = M(p_2) = 0$, $M(p_3) = k$ et $M(p_1) = 1$, pour tout k. Ce sont les seuls états stables accessibles à partir de M. Mais tous ces marquages ont la même restriction sur ITF: donc ce sont tous des I-blocages de OIS_2.

Corollaire II-1 (I-blocage et SF-éq)

$$OIS_1 \equiv_{SF} OIS_2 \Rightarrow (OIS_1 \textit{ a un I-blocage} \Leftrightarrow OIS_2 \textit{ a un I-blocage})$$

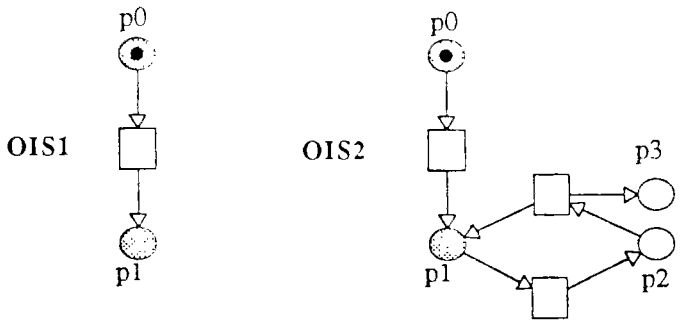


Figure II-6: I-blocage des OI-systèmes

Corollaire II-2 (Conservation d'e.a par SF-éq) *Si* $OIS_1 \equiv_{SF} OIS_2$ *et* $H_1 \subseteq RSS_1$ *est un espace d'accueil de* OIS_1, *alors* $H_2 = \{M_2 \in RSS_2; \exists M_1 \in H_1 \wedge M_1 \mathcal{R}_{SF} M_2\}$ *est un espace d'accueil de* OIS_2.

Il n'y a pas de conservation "stricte" d'espace d'accueil: premièrement ça doit être un espace d'accueil stable, et à un état d'accueil peut correspondre un espace d'accueil: mais si M_1 est un état d'accueil stable, alors l'espace d'accueil correspondant H_2 vérifie $H_2{\downarrow_{ITF}} = \{M_1{\downarrow_{ITF}}\}$.

L'SF-équivalence est suffisante pour comparer deux OI-systèmes mais non pour faire du remplacement: si dans un OI-système on remplace un sous-réseau par un réseau SF-équivalent, on n'obtient pas forcément un système SF-équivalent au premier. La cause en est que l'SF-équivalence ne prend pas en compte les interactions possibles du sous-réseau avec son environnement et qui peuvent survenir au cours de la transformation d'un état stable en un autre.

Les deux exmples suivants (figures II-7 et II-8) montrent comment le problème du remplacement nous a amené à définir une notion d'équivalence plus forte. Dans les commentaires des exemples, $\otimes$ indique la composition de réseaux par fusion de places, et $a_1 p_1 + \cdots + a_n p_n$ indique le marquage M tel que $M(p_i) = a_i$.

La figure II-7 donne le premier exemple. Notre but est de remplacer un sous-réseau à interface ouverte par un réseau équivalent. OIS_1 et OIS_2 sont SF-équivalents. $ITF = \{p_0, p_2, p_3, p_4\}$, $STB_1 = \{M; M(p_1) = 0\}$ et $STB_2 = \{M; M(p_1') = 0\}$. S'ils ont le même environnement N, alors dans $OIS_2 \otimes N$, $t_4 t_8 t_5 t_7$ est franchissable pour $M(p_0) = 1$ et $p_3 + p_5 + p_6$ est atteint, alors que ce marquage n'est pas accessible dans $OIS_1 \otimes N$.

La figure II-8 donne le deuxième exemple. Dans OIS_1, $STB_1 = \{M; M(p_1) = 0\}$ et dans OIS_2, $STB_2 = \{M; M(p_1') = 0\}$; OIS_1 et OIS_2 sont SF-équivalents, mais s'ils sont composés avec le même environnement N, alors dans $OIS_1 \otimes N$, $2p_0 \xrightarrow{\sigma_1} p_4$ où $\sigma_1 = t_1 t_2 t_7 t_3$, mais il n'y a pas de séquence σ_2 dans $OIS_2 \otimes N$ telle que $2p_0 \xrightarrow{\sigma_2} p_4$.

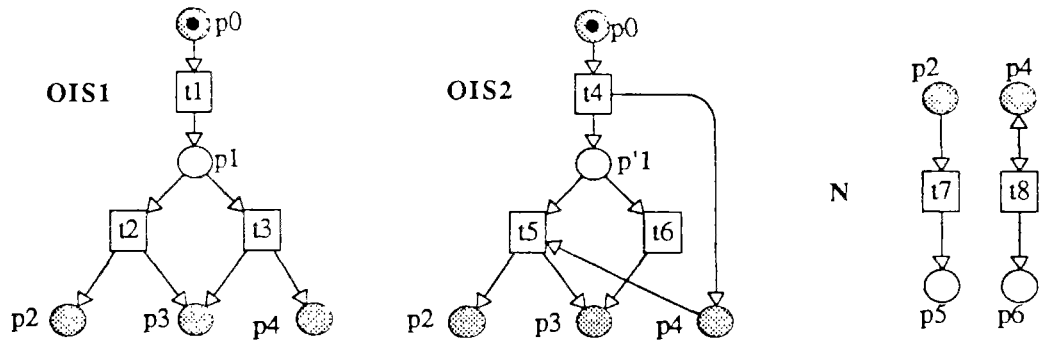


Figure II-7: Insuffisance de l'SF-équivalence

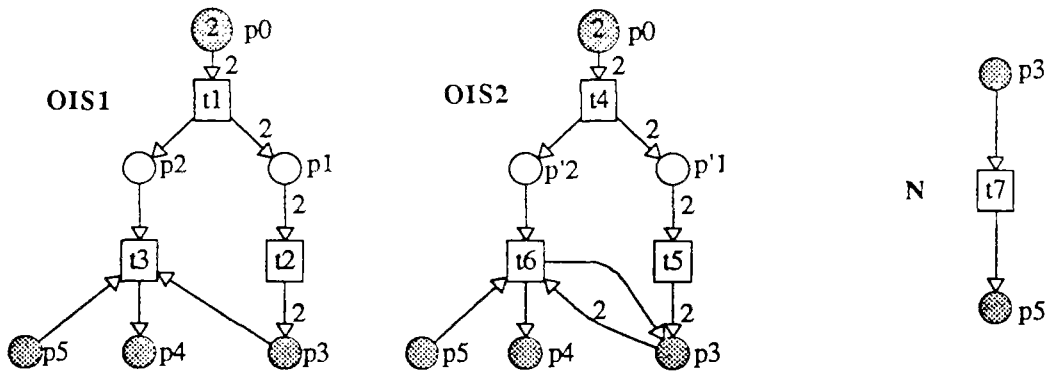


Figure II-8: Insuffisance de l'SF-équivalence

Nous définissons une notion d'équivalence plus forte sur les OI-réseaux et donc indépendante du marquage initial (et donc de l'environnement): équivalence de transformation d'états stables (SST-éq).

II-2.2 SST-équivalence

L'SST-équivalence prend en compte en plus les transformations de l'interface *pendant* la transition d'un état stable vers un autre. Considérons la transformation de l'interface par une séquence de franchissements: les deux réseaux peuvent avoir des degrés différents d'atomicité et donc interagir avec leur environnement durant une transformation produisant des marques et consommant dans ITF.

Il ne suffit plus que les séquences effectuent la même transformation d'états stables, il faut aussi qu'à chaque séquence dans un réseau corresponde une séquence dans l'autre qui permet les mêmes interactions avec tout environnement. Ceci mène à la définition d'une simulation de séquence notée $\leq_{ITF}$.

Définition II-6 (Simulation de séquence) *Soit* $OIN_i \in \mathcal{OIN}_{ITF}$ *pour* $i = 1,2$, *et* $\sigma_i \in T_i^*$.

On dit que σ_2 simule σ_1 *(par rapport à ITF), et on écrit* $\sigma_1 \leq_{ITF} \sigma_2$ *ssi*

(i) $\forall p \in ITF, C_2(p,\sigma_2) = C_1(p,\sigma_1)$

(ii) $(\sigma_1 = t_1 \ldots t_n, t_k \in T_1) \Rightarrow (\exists (w_i)_{i=1,n} \in (T_2^*)^n)$ *tel que*

1. $\sigma_2 = w_1 \ldots w_n$
2. $\forall p \in ITF, \forall k \in 1..n, C_2(p, w_1 \ldots w_k) \geq C_1(p, t_1 \ldots t_k)$
3. $\forall p \in ITF, \forall k \in 1..n, W_2(p, w_k) \leq W_1(p, t_k)$

(i) σ_1 et σ_2 effectuent la même transformation de l'interface.
(ii) exprime la différence d'atomicité et l'interaction avec l'environnement durant la transformation: il existe une décomposition de σ_2 telle que, après le kième pas elle a produit au moins autant que σ_1, et pour chaque pas elle consomme au plus autant que σ_1. Noter que w_i peut être le mot vide.

Nous définissons l'SST-équivalence de façon analogue à l'SF-équivalence au moyen d'une bisimulation liant les états stables, mais les séquences qui effectuent des transformations équivalentes d'états stables doivent vérifier la relation de simulation.

Notation Si $P' \subseteq P$, alors $N \setminus P' = (P'', T; W'')$ est le réseau N privé des places de P', défini par $P'' = P \setminus P'$ et $W' = W{\downarrow}_{((P'' \times T) \cup (T \times P''))}$.

Définition II-7 (SST-équivalence sur $\mathcal{OIN}$) *Soit $OIN_i \in \mathcal{OIN}_{ITF}$ pour $i = 1, 2$, et $\longrightarrow_{i'}$ indique le franchissement dans $(N_i \setminus ITF; M_{0i})$.*

Alors $\mathcal{R}_{SST} \subseteq STB_1 \times STB_2$ est une bisimulation de transformation d'état stable *(SST-bisimulation) ssi:*

1. $M_{01} \mathcal{R}_{SST} M_{02}$
2. $M_1 \mathcal{R}_{SST} M_2 \Rightarrow$
   a) $M_1 \xrightarrow{\sigma_1}_{1'} M_1' \in STB_1 \Rightarrow \exists \sigma_2,\ M_2 \xrightarrow{\sigma_2}_{2'} M_2' \in STB_2,\ \sigma_1 \leq_{ITF} \sigma_2$ *et* $M_1' \mathcal{R}_{SST} M_2'$
   b) *idem que a) mais en permutant 1 et 2.*

On dit que OIN_1 et OIN_2 sont SST-équivalents, *et l'on note $OIN_1 \equiv_{SST} OIN_2$ ssi il existe une SST-bisimulation de OIN_1 vers OIN_2.*

C'est une équivalence car $\leq_{ITF}$ est transitive sur T^*. Uniquement les séquences menant d'un état stable à un autre sont considérées. Noter que les séquences sont celles des OI-réseaux privés de leurs places d'interface dans la définition de l'SST-équivalence, mais ITF est présent à travers $\leq_{ITF}$: c'est ce qui fait que deux OI-réseaux SST-équivalents ont le même comportement dans n'importe quel environnement.

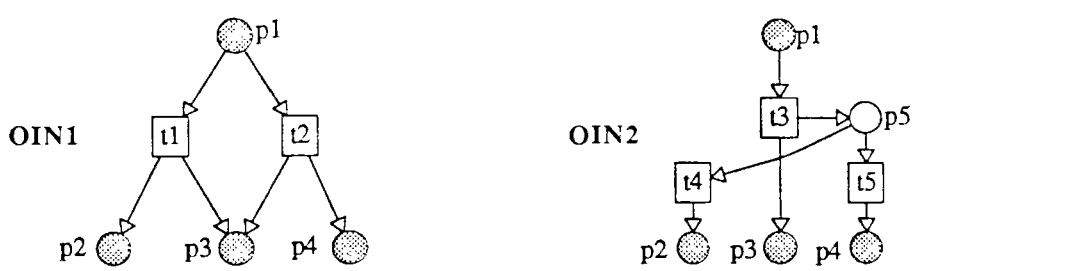


Figure II-9: Deux OI-réseaux SST-equivalents

OIN_1 et OIN_2 (Figure II-9) sont SST-equivalents: $ITF = \{p_1, p_2, p_3, p_4\}$, $STB_1 = \mathrm{N}^{\emptyset}$ et $STB_2 = \{M; M(p_5) = 0\}$. A $\sigma_1 = t_1$ correspond $\sigma_2 = t_3 t_4$ et vice versa, la décomposition de σ_1 est $t_1 \lambda$.

Il est aisé de vérifier que l'SST-équivalence est plus forte que l'SF-équivalence: deux OI-systèmes qui ont des OI-réseaux sous-jacents SST-équivalents et même marquage d'interface, sont SF-équivalents.

Le lemme suivant établit que si σ_2 simule σ_1, alors σ_2 consomme moins de marques que σ_1 de l'interface.

Lemme II-1 *Si* $\sigma_1 \leq_{ITF} \sigma_2$ *alors* $\forall p \in ITF$, $W_2(p, \sigma_2) \leq W_1(p, \sigma_1)$

Preuve Il est aisé de voir que la Définition II-6(ii) implique $W_2(p, \sigma_2) \leq W_1(p, \sigma_1)$ pour tout $p \in ITF$: on prouve $W_2(p, w_1 \ldots w_k) \leq W_1(p, t_1 \ldots t_k)$ par récurrence sur k en utilisant la définition de W. □

Proposition II-1 (SST-éq et SF-éq) *Si* $OIS_i = (OIN_i; m_{0i})$ *sont deux OI-systèmes, alors*

$$OIN_1 \equiv_{SST} OIN_2 \bigwedge m_{01} = m_{02} \Rightarrow OIS_1 \equiv_{SF} OIS_2$$

Preuve Pour alléger les notations, $\longrightarrow$ indiquera le franchissement dans tous les réseaux, privés ou non de leur places d'interface.

$\mathcal{R}_{SST}$ est une SST-bisimulation de OIN_1 vers OIN_2 (cf. Définition II-7). On définit $\mathcal{R}_{SF}$ par $M_1 \mathcal{R}_{SF} M_2$ ssi $M_1{\downarrow}_{ITF} = M_2{\downarrow}_{ITF}$ et $M_1{\downarrow}_{INR_1}\ \mathcal{R}_{SST} M_2{\downarrow}_{INR_2}$.

Si $M_1 \xrightarrow{\sigma_1} M_1' \in RSS_1$, alors $M_1{\downarrow}_{INR_1} \xrightarrow{\sigma_1} M_1'{\downarrow}_{INR_1} \in STB_1$.

Par application de la définition de $\mathcal{R}_{SST}$ on a $\exists \sigma_2, M_2{\downarrow}_{INR_2} \xrightarrow{\sigma_2} M_2'' \in STB_2$, $\sigma_1 \leq_{ITF} \sigma_2$, et $M_1'{\downarrow}_{INR_1}\ \mathcal{R}_{SST} M_2''$.

D'après le lemme II-1, $W_2(p, \sigma_2) \leq W_1(p, \sigma_1)$ pour tout $p \in ITF$.

Donc $M_2 \xrightarrow{\sigma_2} M_2'$ tel que $M_2'{\downarrow}_{INR_2} = M_2''$, et $M_2'{\downarrow}_{ITF} = M_1'{\downarrow}_{ITF}$ à cause de la définition II-6(i). □

On a vu que les réseaux des figures II-7 et II-8 sont SF-équivalents mais non interchangeables. Nous allons maintenant montrer qu'ils ne sont pas SST-équivalents, et mettre en évidence le rôle de la définition de la simulation de séquence (Définition II-6.

OIN_1 et OIN_2 de la Figure II-7 ne sont pas SST-equivalents (OIN_i est l'OI-réseau sous-jacent de OIS_i). Pour $\sigma_2 = t_4 t_5$ il n'y a pas de σ_1 tel que $\sigma_2 \leq_{ITF} \sigma_1$: le seul candidat admissible est $\sigma_1 = t_1 t_2$, mais il n'y a aucune décomposition de $\sigma_1 = w_1 w_2$ telle que $C_1(p_4, w_1) \geq C_2(p_4, t_4)$. En effet, le but de la condition (ii)2 de la définition II-6 est d'empêcher de tels réseaux d'être équivalents.

L'exemple de la Figure II-8 justifie (ii)3 de la Définition II-6. Ces deux IO-réseaux ne sont pas SST-équivalents car pour $\sigma_1 = t_1 t_2 t_3$, il n'y a pas de σ_2 tel que $\sigma_1 \leq_{ITF} \sigma_2$. $\sigma_2 = t_4 t_5 t_6$ ne convient pas car nécessairement n'importe quelle décomposition de σ_2 associera t_6 à t_3 mais $W_2(p_3, t_6) > W_1(p_3, t_3)$.

II-3 Remplacements de sous-réseaux

Dans cette section nous définissons les opérations de remplacement sur $\mathcal{OIN}$ et $\mathcal{OIS}$. Le remplacement d'un OI-sous-réseau par un OI-réseau SST-équivalent ne

pose aucun problème; mais le résultat d'un tel remplacement dans un OI-système n'est pas en général un OI-système. C'est pourquoi nous définissons aussi une classe restreinte de remplacements sur $\mathcal{OIS}$ pour obtenir la fermeture.

D'abord, nous rappelons les définitions et notations de sous-réseau ouvert et de remplacement de sous-réseau pour les réseaux P/T. Un sous-réseau ouvert est un sous-réseau engendré par un sous-ensemble de transitions, et donc sa frontière est composée de places.

Définition II-8 (Sous-réseau ouvert) *Si N est un réseau P/T et $T_1 \subseteq T$, alors le sous-réseau ouvert de N engendré par T_1 est un réseau défini par $N(T_1) = (P_1, T_1, W_1)$ où $P_1 = {}^\bullet T_1^\bullet$, et $W_1 = W{\downarrow}_{((P_1 \times T_1) \cup (T_1 \times P_1))}$.*

Si N_1 est un sous-réseau de N, la frontière de N_1 dans N est $bd_N(N_1) = P_2 \cap P_1$, où $P_2 = {}^\bullet(T \setminus T_1)^\bullet$.

Définition II-9 (Remplacement de sous-réseau) *Soit $N = (P, T; W)$, et $N(T') = (P', T'; W')$ le sous-réseau de N engendré par T'. N'' est tel que $T'' \cap T = \emptyset$ et $P'' \cap P = P'' \cap P' = bd_N(N(T'))$.*

Alors le remplacement de $N(T')$ par N'' donne un réseau défini par

$$N\,[N(T') \leftarrow N''] = ((P \setminus P') \cup P''; (T \setminus T') \cup T''; (W \setminus W') \cup W'')$$

(considérer les fonctions de valuations comme des multi-ensembles)

II-3.1 Remplacements sur $\mathcal{OIN}$

Maintenant, nous donnons les définitions relatives aux OI-réseaux: ce qu'est un OI-sous-réseau et ce qu'est un remplacement d'un tel sous-réseau.

Graphiquement, un OI-sous-réseau est un sous-réseau ouvert dont la frontière (resp. l'ensemble des places internes) est un sous-ensemble de l'interface (resp. l'ensemble des places internes) de l'OI-réseau d'origine; son ensemble d'états stables est défini par restriction. Si l'ensemble des places internes du réseau d'origine est INR, on note INR_1 celui du sous-réseau et $INR_0 = INR \setminus INR_1$: nous exigeons en plus que toute combinaison de deux états stables sur INR_0 et INR_1 soit un état stable sur INR (cf. justification après la définition du remplacement).

Définition II-10 (Sous-réseau à interface ouverte) *Soit un réseau à interface ouverte $OIN = (N; ITF; STB; M_0)$ et un sous-ensemble de transitions $T_1 \subseteq T$ tels que $bd_N(N(T_1)) \subseteq ITF$, alors le sous-réseau de OIN engendré par T_1 est un réseau à interface ouverte défini par $OIN(T_1) = (N_1; ITF_1; STB_1; M_{01})$ tel que:*

- $N_1 = N(T_1)$

- $ITF_1 = ITF \cap P_1$ *et* $INR_1 = P_1 \setminus ITF_1$
- $STB_1 = STB{\downarrow_{INR_1}}$
- $M_{01} = M_0{\downarrow_{INR_1}}$
- $STB = ((STB{\downarrow_{INR_0}}){\uparrow^{INR}}) \cap (STB_1{\uparrow^{INR}})$

L'ensemble de tels sous-réseau de OIN *est noté* $\mathcal{SN}(OIN)$

Définition II-11 (Remplacement sur $\mathcal{OIN}$) *Soit* $OIN \in \mathcal{OIN}$, $OIN_1 \in \mathcal{SN}(OIN)$ *et* $OIN_2 \in \mathcal{OIN}$ *tels que* $N\,[N_1 \leftarrow N_2]$ *soit défini.*

Alors le remplacement de OIN_1 *par* OIN_2 *dans* OIN *donne un réseau à interface ouverte défini par:*

$$OIN' = OIN\,[OIN_1 \leftarrow OIN_2] = (N'; ITF'; STB'; M'_0)$$

où

- $N' = N\,[N_1 \leftarrow N_2]$
- $ITF' = ITF$ *(et* $INR' = INR_0 \cup INR_2$ *où* $INR_0 = INR \setminus INR_1$*)*
- $STB' = ((STB{\downarrow_{INR_0}}){\uparrow^{INR'}}) \cap (STB_2{\uparrow^{INR'}})$
- $M'_0(p) = \begin{cases} M_0(p) & si\ p \in INR_0 \\ M_{02}(p) & si\ p \in INR_2 \end{cases}$

La figure II-10 résume l'opération de remplacement et rappelle les notations utilisées dans ce chapitre. Les ellipses représentent des ensembles de places et les rectangles des ensembles de transitions; un arc d'un ensemble A vers un ensemble B signifie qu'il peut exister un arc d'un nœud de A vers un nœud de B.

OIN_1 et un OI-sous-réseau de OIN engendré par T_1. Ses places internes (INR_1) sont connectées uniquement à des transitions de T_1. Son interface (ITF_1) est la frontière de ce sous-réseau (il peut exister des arcs entre ITF_1 et T_0) et doit être incluse dans l'interface de OIN notée ITF.

Le remplacement de OIN_1 par OIN_2 qui a la même interface ($ITF_2 = ITF_1$) donne OIN'.

C'est pour que le remplacement d'un sous-réseau à interface ouverte par lui-même donne le même OI-réseau d'origine que nous avons imposé la contrainte sur les états stables dans la définition II-10.

Le théorème de remplacement est le résultat fondamental de ce chapitre: le remplacement dans OIN d'un OI-sous-réseau par un OI-réseau SST-équivalent donne un OI-réseau SST-équivalent à OIN.

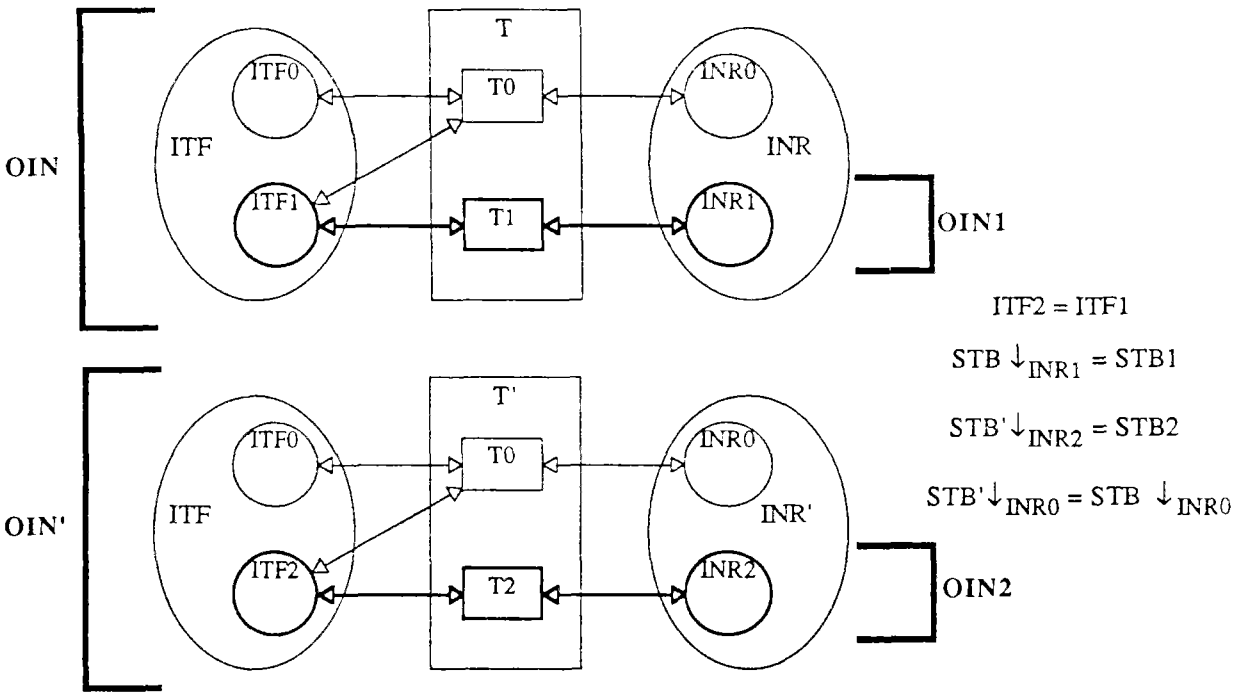


Figure II-10: Remplacement de OIN_1 par OIN_2

Pour prouver ce théorème, il faut construire une SST-bisimulation de OIN vers OIN' (cf. définition II-7), qui relie les états stables STB de OIN à ceux STB' de OIN'.

Le lemme suivant établit un résultat technique dont l'interprétation intuitive n'est pas aisée: il existe une bisimulation liant les états de OIN à ceux de OIN', dont la restriction resp. à INR_1 et INR_2 sont resp. dans STB_1 et STB_2. Les états stables de resp. OIN et OIN' vérifient cette propriété (de restriction).

Lemme II-2 *Si* $OIN_2 \equiv_{SST} OIN_1$, $OIN' = OIN[OIN_1 \leftarrow OIN_2]$, $\mathcal{R}_{SST} \subseteq STB_1 \times STB_2$ *est une SST-bisimulation de* OIN_1 *vers* OIN_2.

On note $XSTB = STB_1{\uparrow}^{INR}$ *et* $XSTB' = STB_2{\uparrow}^{INR'}$.

On définit $\mathcal{R} \subseteq XSTB \times XSTB'$ *par:*

$$M\mathcal{R}M' \stackrel{\text{def}}{\Longleftrightarrow} M{\downarrow}_{INR_1}\ \mathcal{R}_{SST} M'{\downarrow}_{INR_2} \bigwedge M{\downarrow}_{INR_0} = M'{\downarrow}_{INR_0}$$

Alors $M\mathcal{R}M' \Rightarrow$

a) $M \xrightarrow{\sigma} Q \in XSTB \Rightarrow \exists \sigma' \in T'^*,\ M' \xrightarrow{\sigma'} Q' \in XSTB',$ *et* $\sigma \leq_{ITF} \sigma'$ *et* $Q\mathcal{R}Q'$

b) *vice versa*

Preuve Notation: $INR_0 = INR \setminus INR_1$, $ITF_0 = ITF \setminus ITF_1$. $T_0 = T \setminus T_1$ et les noms primés désignent les objets de OIN'. Pour alléger les notations, $\longrightarrow$ indiquera le franchissement dans tous les réseaux privés de leurs places d'interface.

Supposons que $M\mathcal{R}M'$ et $\exists\sigma, M \xrightarrow{\sigma} m \in XSTB$: on veut prouver que $\exists\sigma', M' \xrightarrow{\sigma'} m' \in XSTB'$, $\sigma \leq_{ITF} \sigma'$ et $m\mathcal{R}m'$.

Une séquence $\sigma \in T^*$ est un entrelacement de deux séquences $\sigma_0 \in T_0^*$ et $\sigma_1 \in T_1^*$. En utilisant l' SST-équivalence entre OIN_1 et OIN_2, on peut associer $\sigma_2 \in T_2^*$ à σ_1 tel que $\sigma_1 \leq_{ITF_1} \sigma_2$; la décomposition de σ_2 donnée par $\leq_{ITF_1}$ est utilisée pour construire σ' en entrelaçant σ_0 et σ_2.

La preuve précise et formelle est la suivante.

Si $M_1 = M{\downarrow_{INR_1}} \in STB_1$ et $m_1 = m{\downarrow_{INR_1}} \in STB_1$, alors $M_1 \xrightarrow{\sigma_1} m_1$ où $\sigma_1 = \sigma{\downarrow_{T_1}}$.

Puisque $M_1 \mathcal{R}_{SST} M_2$ où $M_2 = M'{\downarrow_{INR_2}}$ (par déf. de $\mathcal{R}$), alors $\exists\sigma_2 \in T_2^*$ telle que $M_2 \xrightarrow{\sigma_2} m_2 \in STB_2$, $\sigma_1 \leq_{ITF_1} \sigma_2$ et $m_1 \mathcal{R}_{SST} m_2$.

La séquence σ de $OIN \setminus ITF$ peut être écrite $\sigma = (x_i y_i)_{i=1,n}, x_i \in T_0^* \wedge y_i \in T_1^*$. Alors σ' est définie par $\sigma' = (x_i y_i')_{i=1,n}$ où $\sigma_2 = (y_i')_{i=1,n}$ et $y_i' = w_k \ldots w_l \in T_2^*$ si $y_i = t_k \ldots t_l$ (notations de la Définition II-6: se rappeler que $\sigma_1 = (y_i)_{i=1,n}$ et $\sigma_1 \leq_{ITF_1} \sigma_2$).

Maintenant, on va montrer que $M' \xrightarrow{\sigma'} m' \in XSTB'$, $\sigma \leq_{ITF} \sigma'$ et $m\mathcal{R}m'$. $M' \xrightarrow{\sigma'} m' \in XSTB'$ et $m\mathcal{R}m'$ sont faciles à vérifier car quand les places d'interface sont retirées les entourages de T_0 et T_1 (resp. T_2) ne partagent aucune place: donc $M' \xrightarrow{\sigma'} m'$ car $M{\downarrow_{INR_0}} \xrightarrow{\sigma'{\downarrow_{T_0}}}$ et $M{\downarrow_{INR_2}} \xrightarrow{\sigma'{\downarrow_{T_2}}}$; et m' vérifie $m'{\downarrow_{INR_0}} = m{\downarrow_{INR_0}}$ et $m{\downarrow_{INR_1}} = m_1 \mathcal{R}_{SST} m_2 = m'{\downarrow_{INR_0}}$.

Il reste à vérifier $\sigma \leq_{ITF} \sigma'$. Que $\forall p \in ITF, C'(p,\sigma') = C(p,\sigma)$ est évident. D'abord, on donne la décomposition de σ': si $\sigma = t_1 \ldots t_q$ alors $\sigma' = u_1 \ldots u_q$ où $u_i = t_i$ if $t_i \in T_0$, et $u_i = w_i$ (défini ci-dessus) si $t_i \in T_1$. Alors les conditions de la définition II-6 sont facilement vérifiées en utilisant les définitions de C et W. □

Théorème II-1 (Conservation d'SST-équivalence)

$$OIN_2 \equiv_{SST} OIN_1 \Rightarrow OIN\,[OIN_1 \leftarrow OIN_2] \equiv_{SST} OIN$$

Preuve On doit montrer qu'il existe une SST-bisimulation $\mathcal{R}'_{SST} \subseteq STB \times STB'$ de OIN vers $OIN' = OIN\,[OIN_1 \leftarrow OIN_2]$. Utilisant les notations du lemme précédent, on remarque que $STB \subseteq XSTB$ et $STB' \subseteq XSTB'$. Si $M\mathcal{R}M'$ et $M \in STB$ alors $M' \in STB'$ (cf. Définition II-11). Alors on définit $\mathcal{R}'_{SST}$ par: $M\mathcal{R}'_{SST}M'$ ssi $M\mathcal{R}M'$ et $M \in STB$: on déduit du lemme que $\mathcal{R}'_{SST}$ est une SST-bisimulation. □

II-3.2 Remplacements sur $\mathcal{OIS}$

L'étape suivante est de définir l'opération de remplacement sur les OI-systèmes: c'est un remplacement effectué sur l'OI-réseau sous-jacent.

Définition II-12 (Remplacement sur $\mathcal{OIS}$) *Soit* $OIS = (OIN; m_0) \in \mathcal{OIS}$, *et soit* $(OIN_i)_{i=1,2} \in \mathcal{OIN}^2$ *tels que* $OIN_1 \in \mathcal{SN}(OIN)$ *et* $OIN\,[OIN_1 \leftarrow OIN_2]$ *soit défini. Alors le résultat du remplacement de* OIN_1 *par* OIN_2 *dans* OIS *est le couple défini par*

$$OIS\,[OIN_1 \leftarrow OIN_2] = (OIN\,[OIN_1 \leftarrow OIN_2]\,; m_0)$$

Malheureusement, $\mathcal{OIS}$ n'est pas fermé par cette opération: en général, $OIS\,[OIN_1 \leftarrow OIN_2]$ n'est pas dans $\mathcal{OIS}$ car la propriété d'espace d'accueil n'est pas conservée par remplacement (voir exemple ci-dessous). Mais on a quand même le corollaire suivant qui se déduit de la proposition II-1 et du théorème II-1.

Corollaire II-3 (Remplacement sur $\mathcal{OIS}$ et SF-équivalence)

$$\begin{array}{c}(OIN_2 \equiv_{SST} OIN_1) \wedge (OIS\,[OIN_1 \leftarrow OIN_2] \in \mathcal{OIS}) \\ \Downarrow \\ (OIS\,[OIN_1 \leftarrow OIN_2] \equiv_{SF} OIS)\end{array}$$

Les OI-réseaux de la figure II-11 sont SST-équivalents, pour $ITF = \{p_1, p_2\}$, et $STB_1 = \{M; M(p_3) = 1 \wedge M(p_4) = 0\}$, $STB_2 = \{M; M(p_5) = 1 \wedge M(p_6) = 0 \wedge M(p_7) = 0\}$. Quand ils sont composés avec l'environnement N, $(OIN_1 \otimes N; p_1)$ est un OI-système, mais $(OIN_2 \otimes N; p_1)$ ne l'est pas: après franchissement de t_5, il n'est plus possible d'atteindre un état stable; pour cela, il est nécessaire de franchir $t_5 \in ITF^\bullet$. (Remarquer qu'à t_2t_1, on fera correspondre t_4t_3 sans jamais faire intervenir t_5, mais à $t_5t_5t_6$ on fera correspondre $t_2t_1\lambda$.)

Nous avons eu donc l'idée de considérer des classes restreintes de réseaux pour obtenir la propriété de fermeture pour le remplacement dans $\mathcal{OIS}$. Les restrictions concernent les transitions franchies pour atteindre un état stable: l'ensemble des états stables est un $T \setminus ITF^\bullet$-espace d'accueil.

Définition II-13 (OI-réseaux robustes) *Un réseau à interface ouverte* est robuste *ssi* $\forall m_0 \in \mathbf{N}^{ITF}$, $STB\uparrow^P$ *est un* $T \setminus ITF^\bullet$*-espace d'accueil de* $(N; M_0 + m_0)$.

La définition des réseaux robustes revient à dire que STB est un $(T \setminus ITF^\bullet)$-espace d'accueil de $(N; M_0)$ privés de ses places d'interface.

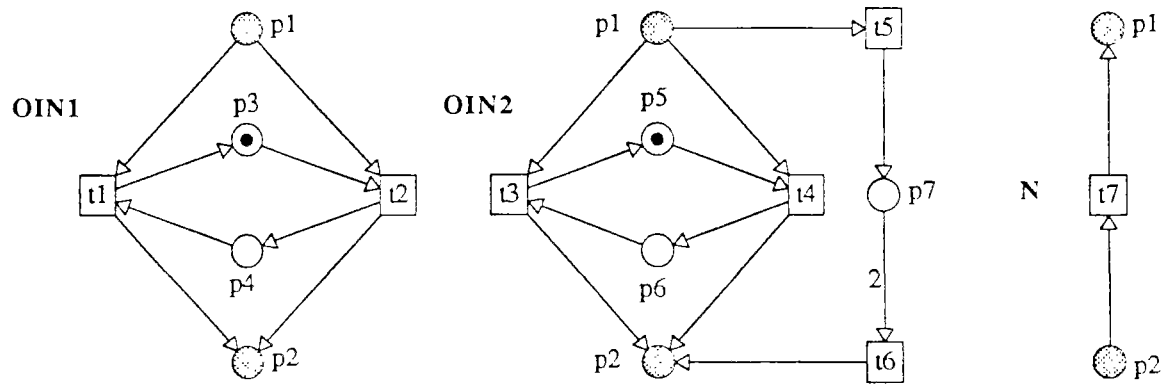


Figure II-11: $\mathcal{OIS}$ n'est pas fermé par remplacement

Proposition II-2 *Soit* $OIN = (N; ITF; STB; M_0)$ *un réseau à interface ouverte.*

Alors, OIN *est robuste ssi* STB *est* $(T \setminus ITF^\bullet)$*-espace d'accueil de* $(N \setminus ITF; M_0)$.

Dans un OI-réseau robuste, pour n'importe quel marquage initial de ITF, il est toujours *possible* d'atteindre un état stable par une séquence ne contenant pas des transitions du post-ensemble de ITF, mais l'existence d'autres séquences n'est pas exclue. Donc une transformation d'état stable ne peut commencer que si toutes les ressources nécessaires sont disponibles, et ces ressources sont consommées dès la première transition.

Tous les exemples de ce chapitre sont des OI-réseaux robustes excepté OIN_2 de la figure II-5 et OIN_2 de la figure II-11.

Les réseaux réentrants définis dans un chapitre ultérieur sont une classe particulière d'OI-réseaux robustes.

Définition II-14 (Remplacement Robuste) *Un remplacement de* OIN_1 *par* OIN_2 *est un* remplacement robuste *ssi* OIN_2 *est un réseau à interface ouverte robuste et* $OIN_2 \equiv_{SST} OIN_1$.

Cette opération est notée $OIS[OIN_1 \leftarrow OIN_2]_r$.

Lemme II-3 *Si* $OIN_1 \equiv_{SST} OIN_2$ *et si* $STB_2\uparrow^{P'}$ *est un espace d'accueil de* $OIS' = OIS\,[OIN_1 \leftarrow OIN_2]$, *alors* $OIS' \in \mathcal{OIS}$.

Preuve Nous utilisons les notations du lemme II-2. Nous voulons prouver la propriété d'espace d'accueil d'un OI-système: si $M'_0 + m'_0 \xrightarrow{\sigma'} Q'$ alors il existe $\sigma'' \in T'^*$ telle que $Q' \xrightarrow{\sigma''} Q'' \in STB'$.

Puisque $STB_2\uparrow^{P'}$ est un espace d'accueil de OIS', il existe $u_2 \in T'^*$ telle que$Q' \xrightarrow{u_2} X'$ et $X'\downarrow_{INR_2} \in STB_2$.

Donc si $v' = \sigma' u_2$, $M'_0 + m'_0 \xrightarrow{v'} X'$ et $X'\downarrow_{INR_2} \in STB_2$.

D'après le lemme II-2, il existe $v \in T^*$ telle que
$M_0 + m_0 \xrightarrow{v} X$ et $X\downarrow_{INR} \mathcal{R} X'\downarrow_{INR'}$ ($m_0 = m'_0$, cf. définition II-12).

Or $OIS \in \mathcal{OIS}$ et donc il existe $z \in T^*$ telle que $X \xrightarrow{z} Y$ et $Y\downarrow_{INR} \in STB$: il découle du lemme II-2 qu'il existe $z' \in T'^*$ tele que $X' \xrightarrow{z'} Y'$ et $Y\downarrow_{INR} \mathcal{R} Y'\downarrow_{INR'}$: donc $Y'\downarrow_{INR'} \in STB'$. D'où $\sigma'' = u_2 z'$. □

Théorème II-2 (Fermeture pour le remplacement robuste)

$$OIS[OIN_1 \leftarrow OIN_2]_r \in \mathcal{OIS} \text{ et donc } OIS[OIN_1 \leftarrow OIN_2]_r \equiv_{SF} OIS$$

Preuve Nous allons montrer que $STB_2\uparrow^{P'}$ est un espace d'accueil de OIS', c-à-d, si $M'_0 + m'_0 \xrightarrow{\sigma'} Q'$, il existe $\sigma'' \in T'^*$ telle que $Q' \xrightarrow{\sigma''} Q'' \in STB_2\uparrow^{P'}$.

On définit $\sigma_2 = \sigma'\downarrow_{T_2}$ et $m_{02} \in \mathbf{N}^{ITF_2}$ par $m_{02}(p) = \sum_{t \in T_0} \bar{\sigma}'(t) W(t,p)$ où $\bar{\sigma}'(t)$ est le nombre d'occurrences de t dans σ'.

Alors, dans OIN_2, $M_{02} + m_{02} \xrightarrow{\sigma_2} Q_2$, $Q_2\downarrow_{INR_2} = Q'\downarrow_{INR_2}$ et $Q_2\downarrow_{ITF_2} \geq Q'\downarrow_{ITF_2}$.

Puisque OIN_2 est un OI-réseau robuste, il existe $u_2 \in (T_2 \setminus ITF_2^\bullet)^*$ telle que $Q_2 \xrightarrow{u_2} X_2$ et $X_2\downarrow_{INR_2} \in STB_2$. D'où, $\sigma'' = u_2$. □

II-4 Expansion et préordre

Le résultat obtenu dans la section précédente est trop restrictif pour une conception hiérarchique: un réseau et son raffinement sont équivalents.

En particulier, ceci implique qu'ils ont les mêmes places d'interface; mais il n'est possible de remplacer que des sous-réseaux dont la frontière est incluse dans l'interface du réseau d'origine.

En plus de l'opération de remplacement, un autre raffinement utile d'un OI-système consiste à étendre son interface: ainsi, des parties cachées deviennent visibles et peuvent être remplacées.

Dans cette section, nous définissons l'expansion d'un OI-système ce qui donne lieu à la définition d'un préordre. Il s'avère que ce préordre est associé à l'SF-équivalence et conserve les propriétés de I-blocage et d'espace d'accueil.

Définition II-15 (Expansion des OI-systèmes) *Soit* $(OIS_i)_{i=1,2} \in \mathcal{OIS}^2$. *On dit que* OIS_2 *est une* expansion *de* OIS_1 *à* $ITF_2 \supseteq ITF_1$, *et on note* $OIS_2 = OIS_1\uparrow^{ITF_2}$, *ssi*

- $N_1 = N_2$
- $ITF_1 \subseteq ITF_2$ *(donc* $ITF_2 \setminus ITF_1 = INR_1 \setminus INR_2$*)*
- $STB_2 = STB_1{\downarrow}_{INR_2}$
- $M_{02} = M_{01}{\downarrow}_{INR_2}$
- $m_{02}(p) = \begin{cases} m_{01}(p) & si\ p \in ITF_1 \\ M_{01}(p) & si\ p \in ITF_2 \setminus ITF_1 \end{cases}$

Définition II-16 (SF-Préordre) *Soit* $(OIS_i)_{i=1,2} \in \mathcal{OIS}^2$. *Alors on dit que* OIS_1 est plus petit que OIS_2 par rapport à la fonctionalité stable, *et l'on note*

$$OIS_1 \preceq_{SF} OIS_2$$

ssi il existe $(OIS'_i)_{i=1,2}$ *tels que*

$$(OIS_i \equiv_{SF} OIS'_i) \bigwedge (OIS'_2 = OIS'_1{\uparrow}^{ITF_2})$$

L'SF-préordre est fondé sur l'expansion et sur l'équivalence: si OIS_1 est plus petit que OIS_2, alors OIS_2 a une interface plus grande et montre plus de "choses" que OIS_1, mais ils sont équivalents sur une "partie commune." Si l'SF-préordre avait été défini d'abord au moyen d'une certaine SF-bisimulation puis l'SF-équivalence par la fermeture symétrique, on aurait pas obtenu cette équivalence sur une partie commune qui implique la conservation de certaines propriétés.

Heureusement, l'SF-équivalence est compatible avec l'SF-préordre, comme le montre la proposition suivante.

Proposition II-3 (SF-équivalence est associée au SF-preordre)

$$OIS_1 \preceq_{SF} OIS_2 \bigwedge OIS_2 \preceq_{SF} OIS_1 \Leftrightarrow OIS_1 \equiv_{SF} OIS_2$$

Preuve
$\Rightarrow$) Si $OIS_1 \preceq_{SF} OIS_2 \wedge OIS_2 \preceq_{SF} OIS_1$ alors $ITF_1 = ITF_2$ et ceci implique $OIS'_1 = OIS'_2$ (notation de la Définition II-16).
$\Leftarrow$) considérer $OIS'_1 = OIS'_2 = OIS_2$ □

Proposition II-4 (SF-préordre et conservation de fonctionalité) *Si deux systèmes à interface ouverte sont tels que* $OIS_1 \preceq_{SF} OIS_2$, *alors il existe une* simulation de fonctionalité stable *de* OIS_1 *vers* OIS_2, *ie, une relation* $S_{SF} \subseteq RSS_1 \times RSS_2$ *vérifiant:*

1. $M_{01} + m_{01} \mathcal{S}_{SF} M_{02} + m_{02}$

2. $M_1 \mathcal{S}_{SF} M_2 \Rightarrow$
   a) $M_1{\downarrow_{ITF_1}} = M_2{\downarrow_{ITF_1}}$
   b) $M_1 \xrightarrow{\sigma_1} M_1' \in RSS_1 \Rightarrow \exists \sigma_2,\ M_2 \xrightarrow{\sigma_2} M_2' \in RSS_2,\ et\ M_1' \mathcal{S}_{SF} M_2'$

De plus, $cod(\mathcal{S}_{SF})$ est un espace d'accueil de OIS_2.

Preuve Nous gardons les notations de la définition II-16. Soit $\mathcal{R}^1_{SF} \subseteq RSS_1 \times RSS_1'$ une SF-bisimulation de OIS_1 vers OIS_1' et soit $\mathcal{R}^2_{SF} \subseteq RSS_2' \times RSS_2$ une SF-bisimulation de OIS_2' vers OIS_2. Alors on définit $\mathcal{S}_{SF}$ par $\mathcal{S}_{SF} = \mathcal{R}^1_{SF} \circ \mathcal{R}^2_{SF}$: c'est bien défini puisque $RSS_1' \subseteq RSS_2'$.

La propriété d'espace d'accueil est prouvée en appliquant le corollaire II-2 et le fait que RSS_1' est un espace d'accueil de OIS_2'. □

Il faut noter que les deux corollaires suivants ne sont pas des conséquences directes de l'SF-simulation mais sont spécialement dûs au fait que $cod(\mathcal{S}_{SF})$ est un espace d'accueil de OIS_2.

La figure II-6 montre un contre-exemple: quand $ITF_9 = ITF_{10} = \{p_0, p_1\}$, OIS_9 et OIS_{10} sont SF-équivalents et tous les deux admettent p_1 comme I-blocage. Mais si $ITF_{10} = \{p_0, p_1, p_2\}$, alors $OIS_9 \preceq_{SF} OIS_{10}$, et OIS_{10} n'a plus de I-blocage.

Puisque un I-blocage peut être un blocage actif, quand on étend l'interface, des modifications inobservables deviennent visibles: donc l'SF-préordre conserve l'absence de I-blocage dans un seul sens (du plus petit vers le plus grand).

Corollaire II-4 (Conservation du non-blocage par SF-préordre)

$$OIS_1 \preceq_{SF} OIS_2 \Rightarrow (OIS_1\ \textit{est sans I-blocage}\ \Rightarrow OIS_2\ \textit{est sans I-blocage}\ )$$

Corollaire II-5 (Conservation d'ea par SF-préordre) *Si $OIS_1 \preceq_{SF} OIS_2$ et $H_1 \subseteq RSS_1$ est un espace d'accueil de OIS_1, alors $H_2 = \{M_2 \in RSS_2; \exists M_1 \in H_1 \wedge M_1 \mathcal{S}_{SF} M_2\}$ est un espace d'accueil de OIS_2.*

II-5 Modélisation et analyse hiérarchiques

Les notions définies dans les sections précédentes munissent la méthode de conception hiérarchique d'une méthode d'analyse hiérarchique. Le système est modélisé par un OI-système abstrait qu'on raffine successivement. Deux opérations de raffinement sont disponibles: le remplacement d'un OI-réseau par un OI-réseau SST-équivalent, et l'expansion de l'interface. Si $ref(OIS)$ est le raffinement de OIS, dans le premier cas, $ref(OIS) \equiv_{SF} OIS$, et dans le deuxième cas,

$ref(OIS) \preceq_{SF} OIS$. Donc, si OIS_0 est le premier modèle abstrait et OIS_n le modèle obtenu après n raffinements, $OIS_0 \preceq_{SF} OIS_n$.

Le modèle client-serveur présenté dans l'introduction nous servira d'exemple. Nous commençons par le modèle de base OIS_0 (figure II-12): les places d'interface sont les places ombrées, et $STB_0 = \{M; M(SX) = 0, M(SI) = 1\}$.

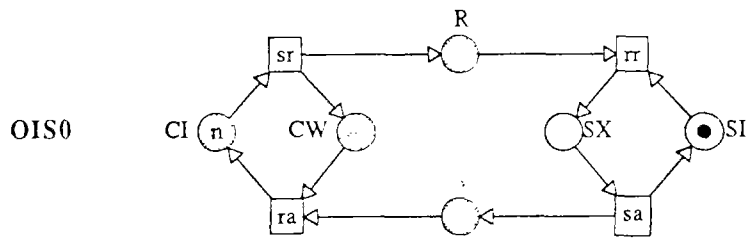


Figure II-12: Modèle de base

Le premier raffinement est l'ajout du tampon: l'OI-réseau $OIN_0(\{rr, sa\})$ est remplacé donnant OIS_1 (Figure II-13), où $ITF = \{CI, CW, R, A\}$, $STB_0 = \{M; M(SX) = 0 \wedge M(SI) = 1\}$ et $STB_1 = \{M; M(BB) = 0, M(FB) = k, M(SB) = M(SX) = M(SC) = 0, M(SI) = 1\}$. C'est un remplacement robuste et donc $OIS_0 \equiv_{SST} OIS_1$.

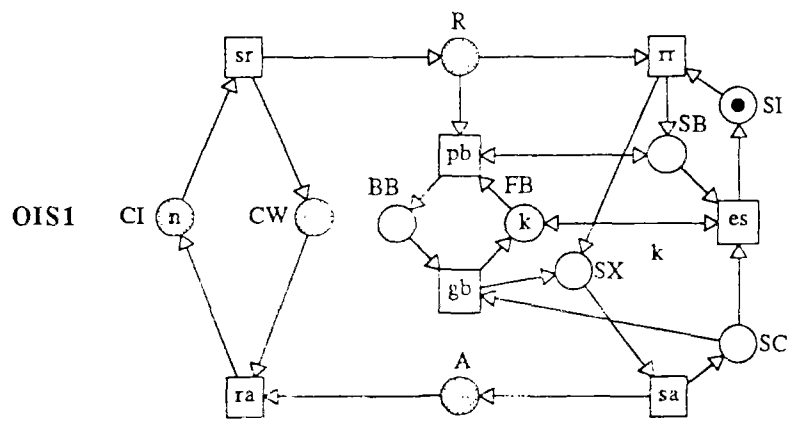


Figure II-13: OIS_1 est obtenu après un remplacement sur OIS_0

En fait, le tampon n'est pas un composant passif de stockage mais la requête est prétraitée avant d'être stockée dans un tampon à x places.

Donc on veut remplacer l'OI-sous-réseau $OIN_1(\{pb, gb\})$. Mais la frontière de ce sous-réseau n'est pas incluse dans l'interface de OIS_1.

C'est pourquoi on étend l'interface et l'on obtient OIS_2 (Figure II-14): $OIS_2 =$

$OIS_1{\uparrow}^{ITF_2}$, où $ITF_2 = \{CI, CW, R, A, FB, SB, SX, SC\}$, et donc $OIS_0 \preceq_{SF} OIS_2$.

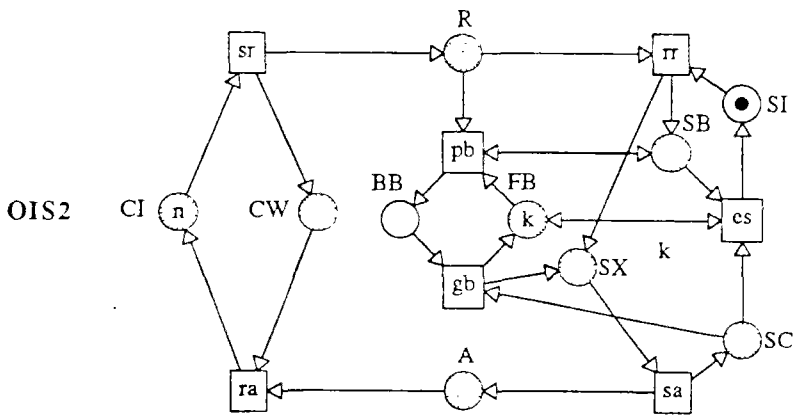


Figure II-14: OIS_2 est une expansion de OIS_1

Finalement, on remplace l'OI-sous-réseau $OIN_2(\{pb, gb\})$ pour obtenir OIS_3 (Figure II-15): une requête est prétraitée dans PX, puis tamponnée dans $BB2$.

Les états stables sont

$$STB_3 = \{M; M(PI) = 1, M(PX) = 0, M(FB2) = x, M(BB2) = 0, M(SI) = 1\}$$

Ce n'est pas un remplacement robuste cette fois-ci: nous devons donc vérifier qu'il est toujours possible de vider PX et $BB2$, ce qui signifie que $STB{\uparrow}^{P_3}$ est un espace d'accueil de OIS_3 (ou' P_3 est l'ensemble des places de OIS_3 et $STB = \{M; M(PX) = M(BB2) = 0 \wedge M(PI) = 1 \wedge M(FB2) = x\}$). Ceci peut se faire, par exemple, au moyen d'une norme et des flots du réseau.

D'après le lemme II-3, on a $OIS_3 \in \mathcal{OIS}$, et donc $OIS_2 \equiv_{SF} OIS_3$.

Il résulte des relations entre les OI-systèmes que $OIS_0 \preceq_{SF} OIS_3$. Donc OIS_3 est sans I-blocage puisque OIS_0 est sans I-blocage

De plus M_{00}, l'état initial de OIS_0, est un état d'accueil et donc $H_3 = \{M \in RSS_3; M{\downarrow}_{ITF_0} = M_{00}{\downarrow}_{ITF_0}\}$ est un espace d'accueil de OIS_3.

Mais d'après les flots de ce réseau, on a $H_3 = \{M_{03}, M_3\}$, où M_{03} est le marquage initial de OIS_3, et M_3 est le marquage tel que M_3 soit égal à M_{03} sur $P_3 \setminus \{SB, SC, SI\}$ et $M_3(SC) = M_3(SB) = 1 \wedge M_3(SI) = 0$.

Donc M_{03} est un état d'acueil de OIS_3 puisque $M_3 \xrightarrow{es} M_{03}$.

Cet exemple montre comment alterner les remplacements et les expansions dans un processus de modélisation et analyse hiérarchiques.

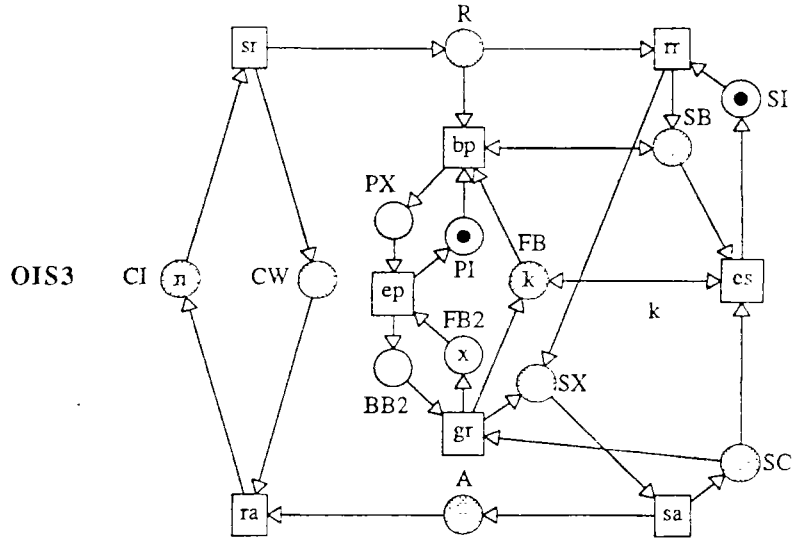


Figure II-15: OIS_3 est obtenu après un remplacement sur OIS_2

Quand le remplacement est robuste, l'analyse est réellement "modulaire" puisqu'il suffit de comparer le sous-réseau remplacé et le réseau de remplacement, et de vérifier que ce dernier est robuste.

Mais quand le réseau de remplacement n'est pas robuste, il y a une vérification supplémentaire d'espace d'accueil à mener sur le réseau entier.

D'autre part, le calcul des flots complète cette méthode en permettant de préciser les espaces d'accueil qu'on obtient par remplacement. C'est en utilisant les flots que nous avons pu réduire H_3 en M_{03}.

Conclusion

Nous avons défini un ensemble de notions pour faire de l'analyse hiérarchique de réseaux par remplacement de sous-réseaux ouverts (à frontière de places). A notre connaissance, notre SST-équivalence est la première notion basée sur l'observation d'états à considérer le problème à ce niveau de généralité.

Ceci définit un bon cadre théorique mais la complexité de cette équivalence risque d'être rédhibitoire en pratique: il faut trouver une bisimulation et vérifier $\leq_{ITF}$. Il faut donc définir des classes particulières de réseaux où la vérification de l'équivalence serait plus simple: ceci est l'objet du chapitre sur les réseaux réentrants.

La définition de transformations de réseaux qui conservent l'SST-équivalence est une autre voie que nous n'avons pas encore explorée. Il semble que l'application

des transformations de Berthelot[8] à des transitions non-adjacentes aux places d'interface doive conserver l'équivalence.

D'un autre côté, cette équivalence est un peu trop fine, car elle n'identifie que des réseaux interchangeables dans n'importe quel environnement: ceci réduit son domaine d'application. Des travaux ultérieurs doivent considérer une équivalence par rapport à un certain environnement: prise en compte des marquages possibles de l'interface et de la connexion des réseaux à leur environnement, et différenciation des places d'interface (actuellement l'interface est traitée de façon uniforme).

Chapitre III

Composition Asynchrone

Introduction

Dans le chapitre précédent, nous avons étudié un concept de modularité "verticale", à savoir le raffinement hiérarchique par remplacement de modules par des modules équivalents. Ici nous considérons un concept de modularité "horizontale", c-à-d, la composition des modèles. Il existe deux façons naturelles de composer des réseaux de Petri: la fusion de transitions et la fusion de places.

La fusion de transitions correspond à la synchronisation d'événements par (multi-)rendez-vous. Elle a déjà fait l'objet de nombreux travaux qui posent le problème de déterminer le comportement d'un système à partir des comportements de ses composants, ou de trouver sous quelles conditions des propriétés du système (vivacité, caractère borné) peuvent être déduites de celles de ses composants. Mazurkievicz[41] montre que la composition des ensembles de traces de deux réseaux est égal à l'ensemble des traces du réseau composé; Winskel[66] étudie le problème dans le cadre des catégories où la composition de deux réseaux par fusion de transitions correspond au produit; Souissi[51,52] étudie la conservation de la vivacité dans la composition de deux réseaux à travers un medium de communication et la composition par rendez-vous multiples ordonnés; Valmari[59] propose une méthode de composition des graphes d'états correspondant à la composition par fusion de transitions: on associe à chaque composant son graphe qu'on réduit par des transformations conservant une certaine relation d'équivalence, puis on compose les graphes réduits. (voir aussi l'introduction du chapitre précédent.)

Ici nous étudions la composition par fusion de places que nous appelons composition asynchrone par opposition à la fusion de transitions qui est une synchronisation d'événements.

L'étude de la composition par fusion de places (notée $\otimes$) est plus difficile que

celle de la composition par fusion de transitions (notée |) car, si Σ_1 et Σ_2 sont deux systèmes P/T, les projections du graphe des marquages de $\Sigma_1|\Sigma_2$ sont inclus dans les graphes de marquages de Σ_1 et Σ_2, cette propriété n'étant pas vérifiée par $\Sigma_1 \otimes \Sigma_2$.

Cette opération admet plusieurs interprétations: le séquencement, l'alternative ou l'itération sont obtenues par des fusions de places quand les places sont considérées comme les préconditions et les postconditions d'une action (Kotov[37]); quand on fusionne des places représentant des ressources des systèmes, la fusion s'interprète comme un partage de ressources ou un multiplexage si les places représentent l'état repos d'une unité.

Dans le cas général, nous obtenons essentiellement la conservation des flots, et celle des espaces d'accueil quand leur accessibilité est indépendante des places partagées. En particulier, ce résultat implique que la composition de deux réseaux à interface ouverte robuste est un OI-réseau robuste.

La diificulté du problème a déjà été mise en évidence par d'autres travaux. Dans [53], Souissi étudie la conservation de la vivacité lors de la composition de réseaux par fusion de places, et est amené à supposer des conditions de monotonie de vivacité des réseaux. Petrucci[45] présente un algorithme de construction du graphe des marquages de la composition de deux réseaux: dans le cas de la composition par fusion de places, l'algorithme est plus complexe et construit implicitement plusieurs graphes pour chacun des composants.

La relation entre un réseau composé et ses composants étant assez décevante dans le cas général, nous étudions la composition de réseaux appartenant à des classes plus restreintes, et nous nous intéressons à des propriétés particulières: nous considérons le problème du partage de ressources et la preuve modulaire de la propriété "il est toujours possible de libérer les ressources."

Dans [33], Holt développe une théorie des systèmes et des catégories de ressources en vue d'étudier les propriétés de blocages. Il distingue deux types de ressources: les renouvelables et les consommables.

Les ressources consommables sont disponibles en nombre infini et ne sont pas réutilisables (messages, signaux): leur partage est modélisé par la fusion de places représentant des canaux, et donc initialement vides.

Les ressources renouvelables sont disponibles en nombre fini et le processus qui les utilise les libère quand il n'en a plus besoin (imprimantes, variable de sémaphore); leur partage est modélisé par la fusion de places bornées par leur marquage initial.

Quelques analyses du problème d'allocation de ressources au moyen des réseaux de Petri existent centrées autour de l'algorithme d'Habermann[28] et celui du banquier[21]. Lautenbach et Thiagarajan[39] ont considéré des processus séquen-

tiels modélisés par des réseaux cycliques, et ont montré que l'ajout de certaines places empêchent le franchissement des séquences qui ne sont pas "fiables" (qui mènent à un blocage) et celles-ci seulement. Tazza[56] reprend le problème sur une classe de réseaux plus générale, où la stratégie d'allocation n'est plus optimale (c-à-d, certaines séquences fiables sont aussi interdites): il étudie alors quantitativement la déviation par rapport à l'optimum. Hauschildt et Valk[32] étudient à partir d'une représentation au moyen d'un réseau de Petri de l'algorithme du banquier, et dérivent des formules caractérisant les états fiables (qui ne mènent pas à un blocage).

Nous n'étudions pas de stratégie d'allocation de ressources, mais des conditions que doivent vérifier les réseaux dans leurs demandes et libérations de ressources pour éviter l'interblocage. Le résultat principal de ce chapitre est une solution du problème d'interblocage dans les systèmes à ressources renouvelables, basée sur un ordonnancement des ressources. Notre solution diffère de la solution classique en ce qu'il existe des choix dans les suites de demandes de ressources.

Dans la première section, nous examinons les propriétés élémentaires de cette composition: relation entre les langages des composants et celui du composé, ainsi que les propriétés relatives aux flots. Nous énonçons un premier résultat sur la conservation d'espace d'accueil dans certaines conditions.

Dans la deuxième section, la classe des réseaux à ressources renouvelables est définie: il s'agit de réseaux où on distingue un ensemble de places (ressources) bornées par leur marquage initial. Nous montrons alors que la relation entre les comportements des composants et celui du composé devient plus étroite.

Dans la troisième section, nous définissons la classe des réseaux à ressources ordonnées: ce sont des réseaux à ressources renouvelables, où les ressources sont ordonnées et il est toujours possible de libérer une ressource de rang k sans demander des ressources de rang $j \leq k$. Nous montrons alors que dans un tel réseau, il est toujours possible de libérer toutes les ressources. De plus, cette classe est fermée par composition ce qui implique que cette propriété est conservée.

L'originalité de cette solution réside dans le fait que nous considérons des processus avec choix, car sinon, ces hypothèses d'ordonnancement coïncident avec celles de la solution classique quand les processus sont linéaires.

La quatrième section esquisse rapidement une extension des différentes opérations et définitions aux réseaux colorés.

III-1 Composition par fusion de places

Cette section étudie certaines propriétés générales de la composition par fusion de places: quelques résultats sur relations entre un système et ses composants en ce qui concerne les séquences franchissables, les invariants, les places implicites et les espaces d'accueil.

III-1.1 Définition et comportement

Nous appelons la composition par fusion de places composition asynchrone par opposition à la fusion des transitions qui est une synchronisation d'événements. Graphiquement, l'opération consiste à superposer les places communes aux deux réseaux; pour les réseaux marqués, les places fusionnées doivent avoir le même marquage initial, car les deux systèmes partagent les mêmes ressources et n'apportent pas chacun ses ressources. La figure III-1 résume l'opération et fixe les notations.

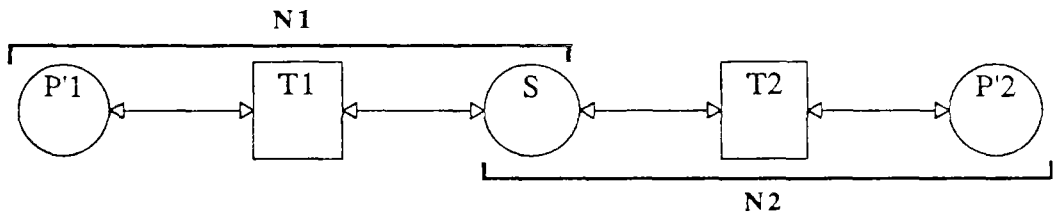


Figure III-1: Composition asynchrone

Définition III-1 (Composition asynchrone de réseaux)
La composition asynchrone *de deux réseaux* $N_i = (P_i, T_i; W_i)$, $i = 1, 2$, *est définie par:*

$$N_1 \otimes N_2 = (P, T; W)$$

où

- $P = P_1 \cup P_2$
- $T = T_1 \uplus T_2$ *(T est l'union disjointe de T_1 et T_2)*
- $W = W_1 \cup W_2$

Définition III-2 (Composition asynchrone de systèmes)
La composition asynchrone de deux systèmes P/T, $(N_i; M_{0i})$, $i = 1, 2$, *tels que*

$$M_{01}\downarrow_{P_1 \cap P_2} = M_{02}\downarrow_{P_1 \cap P_2}$$

est définie par

$$(N_1; M_{01}) \otimes (N_2; M_{02}) = (N_1 \otimes N_2; M_0)$$

où

$$M_0{\downarrow_{P_i}} = M_{0i}{\downarrow_{P_i}}$$

Dans la suite, on notera $P_1 \cap P_2 = S$ *et* $P_i' = P_i \setminus S$.

La proposition suivante montre la différence fondamentale entre la composition par fusion de transitions et celle par fusion de places.

Dans la composition par fusion de transitions (qu'on notera |), si $M \xrightarrow{\sigma} M'$ dans $N_1|N_2$ alors $M_i \xrightarrow{\sigma_i} M_i'$ où $M_i = M{\downarrow_{P_i}}$, $\sigma_i = \sigma{\downarrow_{T_i}}$ et $M_i' = M'{\downarrow_{P_i}}$; autrement dit la projection du graphe des marquages de $\Sigma_1|\Sigma_2$ sur les places et les transitions de Σ_i est un sous-graphe du graphe de Σ_i (où $\Sigma_i = (N_i; M_{0i})$).

Cette relation est fausse dans la composition par fusion de places.

La figure III-2 montre un exemple de client-serveur ayant un compteur qui mémorise le nombre de requêtes traitées; CTR est la place modélisant le compteur.

Dans $\Sigma_1 \otimes \Sigma_2$, on a pour tout n, $CI + SI \xrightarrow{\sigma} CI + SI + n.CTR$ pour $\sigma = (er.rr.ea.ra)^n$; or dans Σ_2 rien n'est franchissable à partir de SI mais $SI + n.R \xrightarrow{\sigma_2} SI + n.CTR + n.A$ où $\sigma_2 = (rr.ea)^n$.

Donc le marquage qui permet le franchissement de $\sigma{\downarrow_{T_i}}$ dans N_i dépend de M et de σ, et M et M_i sont égaux uniquement sur P_i privé des places partagées.

Cela signifie que le comportement du système résultant de la composition ne se déduit pas du comportement des composants, d'où la difficulté d'étudier cette composition et le besoin de mettre des restrictions pour obtenir une relation entre le réseau composé et ses composants. Ce sera l'objet de la section suivante avec les réseaux à ressources renouvelables.

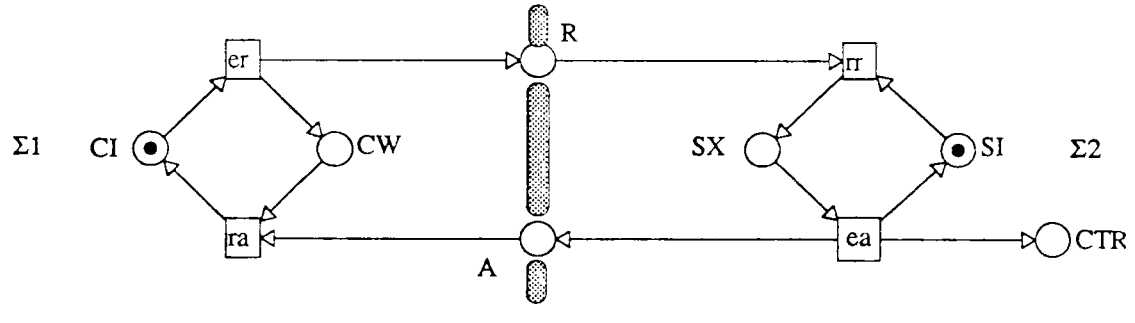


Figure III-2: Composition et comportement

Proposition III-1 (Composition et comportement) *Soit Σ_1 et Σ_2 deux systèmes P/T, $\sigma \in L(\Sigma_1 \otimes \Sigma_2)$ et $M_0 \xrightarrow{\sigma} M$.*
En général

$$\sigma\downarrow_{T_i} \notin L(\Sigma_i)$$

mais il existe $M'_{0i} \in \mathbf{N}^{P_i}$ dépendant de σ, tel que

$$M'_{0i} \xrightarrow{\sigma_i} M'_i, \text{ où } \sigma_i = \sigma\downarrow_{T_i}$$

$$M'_i\downarrow_{P'_i} = M\downarrow_{P'_i} \bigwedge M'_i\downarrow_S \geq M\downarrow_S$$

(idem pour M'_{0i} et M_0).

Preuve Si $M_0 \xrightarrow{\sigma} M$, on définit M'_{0i} en ajoutant au marquage initial de Σ_i toutes les productions dans les places partagées des transitions de σ appartenant à T_j (où {i,j}={1,2}):

$$\forall p \in P_i, M'_{0i}(p) = M_{0i}(p) + \sum_{t \in T_j} \bar{\sigma}(t) W(t,p)$$

où $\bar{\sigma}(t)$ est le nombre d'occurrences de t dans σ.

Il est alors facile de vérifier que $M'_{0i} \xrightarrow{\sigma_i} M'_i$ et que les relations entre M'_i et M sont vraies. □

III-1.2 Invariants structurels

Les invariants inductifs sont des outils de preuve importants. Certains de ces invariants se déduisent de la structure du réseau, c-à-d de la matrice d'incidence ou des matrices *Pre* et *Post*.

Proposition III-2 (Invariants structurels) *Soit X_i la matrice d'incidence, de Pre ou de Post de N_i, et X la matrice correspondante de $N_1 \otimes N_2$. Soit $\mathcal{R} \subseteq \mathbf{Z} \times \mathbf{Z}$, $a_i \subseteq \mathbf{Z}^{P'_i}$, $a \subseteq \mathbf{Z}^S$ et $b \in \mathbf{Z}^{T_1}$.*
Alors pour tout $d \in \mathbf{Z}$ on a:

(i)

$$\forall i \in \{1,2\}, \forall t \in T_i, \left(\sum_{p \in P'_i} a_i(p) X_i(p,t) + \sum_{p \in S} a(p) X_i(p,t)\right) \mathcal{R}\ d$$

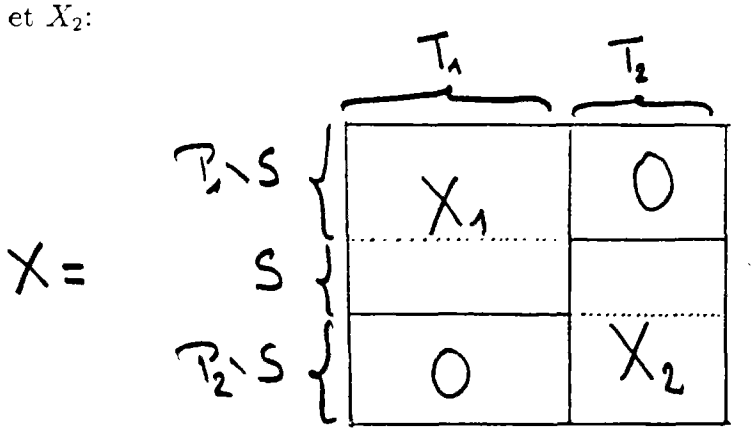
$$\Updownarrow$$

$$\forall t \in T_1 \cup T_2, \left(\sum_{p \in P'_1} a_1(p) X(p,t) + \sum_{p \in S} a(p) X(p,t) + \sum_{p \in P'_2} a_2(p) X(p,t)\right) \mathcal{R}\ d$$

(ii)

$$\forall p \in P_1, \left(\sum_{t \in T_1} b(t)X_1(p,t)\right) \mathcal{R}\ d \Rightarrow \forall p \in P_1 \cup P_2 \left(\sum_{t \in T_1} b(t)X(p.t)\right) \mathcal{R}\ d$$

Preuve La preuve est immédiate comme le montre la construction de X à partir de X_1 et X_2:



□

(i) concerne les invariants de place: si $\mathcal{R}$ est l'égalité et $d = 0$, alors $f \in \mathbf{N}^P$ est un flot de $N_1 \otimes N_2$ si et seulement si $f{\downarrow_{P_i}}$ est un flot de N_i pour $i = 1, 2$. Autrement dit, on peut composer deux flots s'ils ont mêmes coefficients sur les places partagées. En particulier, les flots de N_i dont le support ne contient pas des places partagées, sont conservés dans $N_1 \otimes N_2$. Si $\mathcal{R}$ est la relation $\geq$ (ou $\leq$), il s'agit des invariants d'inégalité.

(ii) concerne les invariants de transition ou "rythmes": un rythme de N_1 est évidemment un rythme de $N_1 \otimes N_2$ mais il peut exister des rythmes de $N_1 \otimes N_2$ dont les projections sur T_1 et T_2 ne soient pas des rythmes. Il en est de même pour les séquences croissantes.

Dans l'exemple de la figure III-3, $A + B + C$ est un flot de N_1 et $B + C + D$ un flot de N_2: comme les coefficients sur $\{B, C\}$ sont les mêmes dans les deux réseaux, on en déduit que $A + B + C + D$ est un flot de $N_1 \otimes N_2$.

$2A+B+2C$ est un flot de N_3 et $B+C+D$ un flot de N_4: comme les coefficients de C sont différents dans les deux réseaux, on ne peut pas en déduire de flot; mais $A + B + C \geq 0$ est vrai dans N_3 et donc on en déduit que $A + B + C + D \geq 0$ est vrai dans $N_3 \otimes N_4$, c-à-d, $M(A) + M(B) + M(C) + M(D) \geq k$ est un invariant inductif de $N_3 \otimes N_4$.

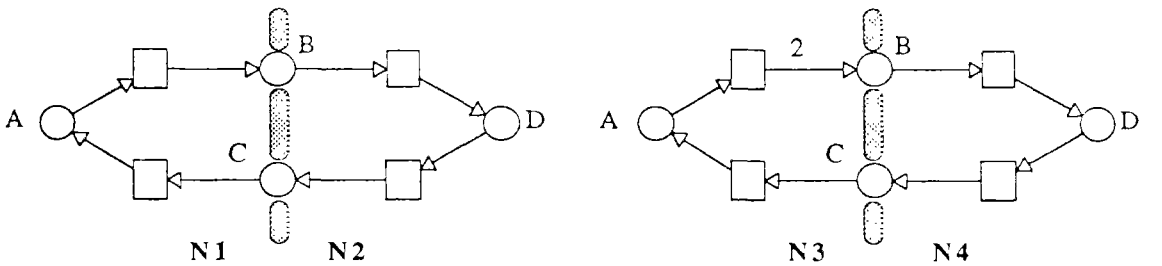


Figure III-3: Invariants structurels

III-1.3 Conservation de place implicite

Les places implicites (structurelles) étai. déterminées par des propriétés linéaires de la matrice d'incidence et du marquage initial, le résultat précédent permet de déduire des conditions de conservation de place implicite lors de la composition. La définition de place implicite donnée ici est inspirée de Silva[17].

Définition III-3 (Place implicite) *Soit $(N; M_0)$ un système P/T et $p_0 \in P$. On dit que p_0 est une place implicite par rapport à (I, f) où $I \subseteq P \setminus \{p_0\}$ et $f \in \mathbf{N}^{I \cup \{p_0\}}$ ssi*

$$\forall t \in T, f(p_0)C(p_0, t) - \sum_{p \in I} f(p)C(p, t) \geq 0$$

$$\forall t \in T, f(p_0)M_0(p_0) - \sum_{p \in I} f(p)M_0(p) \geq f(p_0)W(p_0, t) - \sum_{p \in I} f(p)W(p, t)$$

Quand une place partagée est implicite pour les deux réseaux, et qu'elle l'est par rapport aux mêmes places partagées avec les mêmes coefficients dans les deux réseaux, cette place est implicite dans le réseau obtenu par composition. Donc cette place partagée peut être supprimée, ce qui simplifie l'analyse du réseau composé.

Corollaire III-1 (Conservation de place implicite) *Soit $\Sigma_i(N_i; M_{0i})$ deux systèmes P/T, et $s_0 \in S = P_1 \cap P_2$.*

Si s_0 est implicite par rapport à (I, f) où $I \subseteq S$, dans Σ_1 et Σ_2, alors s_0 est implicite par rapport à (I, f) dans $\Sigma_1 \otimes \Sigma_2$.

Preuve On suppose que $s_0 \in S$ est implicite par rapport à (I, f) dans les deux réseaux (et donc $I \subseteq S$).

$$\forall i \in \{1, 2\}, \forall t \in T_i, f(s_0)C_i(s_0, t) - \sum_{p \in I} f(p)C_i(p, t) \geq 0$$

Par application de la proposition III-2, on obtient

$$\forall t \in T_1 \cup T_2, f(s_0)C(s_0, t) - \sum_{p \in I} f(p)C(p, t) \geq 0$$

De même, de

$$\forall i \in \{1,2\}, \forall t \in T_i, f(s_0)M_{0i}(p_0) - \sum_{p \in I} f(p)M_{0i}(p) \geq f(s_0)W_i(p_0,t) - \sum_{p \in I} f(p)W_i(p,t)$$

en notant que

$$d = f(s_0)M_{01}(p_0) - \sum_{p \in I} f(p)M_{01}(p) = f(s_0)M_{02}(p_0) - \sum_{p \in I} f(p)M_{02}(p)$$

car $M_{01}{\downarrow_S} = M_{02}{\downarrow_S}$, on déduit

$$\forall t \in T_1 \cup T_2, f(s_0)M_0(p_0) - \sum_{p \in I} f(p)M_0(p) \geq f(s_0)W(p_0,t) - \sum_{p \in I} f(p)W(p,t)$$

□

III-1.4 Conservation d'espaces d'accueil

La dernière propriété examinée est celle d'espace d'accueil. Pour avoir une conservation de cette propriété lors de la composition, nous supposons une certaine indépendace vis-à-vis des places partagées et donc de leurs transitions sorties. Nous considérons donc des $(T \setminus S^\bullet)$-espaces d'accueil, c-à-d accessibles sans franchir des transitions en sortie des places partagées, et accessibles pour une classe de marquages initiaux, c-à-d si l'environnement avec lequel est composé le réseau vérifie certaines propriétés.

Proposition III-3 (Conservation d'espaces d'accueil) *Soit $(N_i; M_{0i})$ deux systèmes P/T, $MI_i \subseteq \mathbb{N}^{P_i}$ et $E_i \subseteq \mathbb{N}^{P_i'}$.*

On note $(N; M_0) = (N_1; M_{01}) \otimes (N_2; M_{02})$.

Si

$$\forall M_i \in MI_i, {E_i}{\uparrow^{P_i}} \text{ est un } (T_i \setminus S^\bullet)\text{-espace d'accueil de } (N_i; M_i)$$

et

$$\forall \sigma \in L((N; M_0)), \exists M'_{0i} \in MI_i, (M'_{0i}{\downarrow_{P_i'}} = M_{0i}{\downarrow_{P_i'}}) \wedge (\sigma{\downarrow_{T_i}} \in L((N_i; M'_{0i})))$$

alors

$${E_1}{\uparrow^P} \cap {E_2}{\uparrow^P} \text{ est un } (T \setminus S^\bullet)\text{-espace d'accueil de } (N_1; M_{01}) \otimes (N_2; M_{02})$$

Notons que l'existence d'un M'_{0i} tel que $\sigma\downarrow_{T_i} \in L((N_i; M'_{0i}))$ est vraie d'après la proposition III-1, mais la contrainte supplémentaire est $M_i \in MI_i$.

Preuve Soit $\sigma \in T^*$ tel que $M_0 \xrightarrow{\sigma} M$. On veut montrer qu'il existe s tel que $M \xrightarrow{s} m \in E_1\uparrow^P \cap E_2\uparrow^P$.

Par hypothèse, il existe $M'_{01} \in MI_1$ tel que $M'_{01}\downarrow_{P'_1} = M_{01}\downarrow_{P'_1}$ et $M'_{01} \xrightarrow{\sigma_1}_1 M'_1$. Comme $E_1\uparrow^{P_1}$ est un $(T_1 \setminus S^\bullet)$-espace d'accueil de $(N_1; M'_{01})$, il existe $s_1 \in (T_1 \setminus S^\bullet)^*$ telle que $M'_1 \xrightarrow{s_1}_1 M''_1 \in E_1\uparrow^{P_1}$. Or $M\downarrow_{P'_1} = M'_1\downarrow_{P'_1}$ et donc $M \xrightarrow{s_1} Q_1$ où $Q_1\downarrow_{P_2} = M\downarrow_{P_2}$ et $Q_1\downarrow_{P'_1} \in E_1$.

De même, à partir de Q_1, il existe une séquence $s_2 \in (T_2 \setminus S^\bullet)^*$ telle que $Q_1 \xrightarrow{s_2} Q_2$, avec $Q_2\downarrow_{P'_1} = Q_1\downarrow_{P'_1}$ et $Q_2\downarrow_{P'_2} \in E_2$: donc $m = Q_2 \in E_1\uparrow^P \cap E_2\uparrow^P$ et $s = s_1 s_2$. □

Corollaire III-2 (Composition d'OI-réseaux robustes)
Soit $(N_i; ITF_i; STB_i; M_{0i})$ *deux réseaux à interface ouverte robuste tels que* $(P_1 \cap P_2) \subseteq (ITF_1 \cap ITF_2)$.
Alors $(N; ITF; STB; M_0)$ *est un OI-réseau robuste où*

- $(N; M_0) = (N_1; M_{01}) \otimes (N_2; M_{02})$
- $ITF = ITF_1 \cup ITF_2$
- $STB = STB_1\uparrow^P \cap STB_2\uparrow^P$ *où* $P = P_1 \cup P_2$

Preuve On applique la proposition précédente, en prenant $MI_i = \{M_{0i}\}\uparrow^{P_i}$ et $E_i = STB_i$. □

En fait, ce résultat aurait pu être déduit directement de la caractérisation des OI-réseaux robustes de la proposition II-2. Nous avons voulu montré qu'il est aussi une conséquence de la proposition III-3.

III-2 Réseaux à ressources renouvelables

Dans cette section nous étudions les propriétés de la composition de réseaux par fusion de places marquées et bornées par leur marquage initial. Dans ce cas, le comportement du système global est la composition des comportements de ses composants.

Nous commençons par définir une classe de réseaux où on distingue un ensemble de places bornées par leur marquage initial. Ces places sont interprétées comme modélisant des ressources renouvelables, d'où le nom de ces réseaux.

Définition III-4 (Réseaux à ressources renouvelables) *Un* réseau à ressources renouvelables *est un triplet* $RRN = (N; M_0; RES)$ *où*

- $(N; M_0) = (P, T; W; M_0)$ *est un système place/transition*
- $RES \subseteq P$ *est l'ensemble des ressources de* RRN *vérifiant: pour tout* $r \in RES$, r *est bornée par* $M_0(r)$ *dans* $(N; M_0)$.

L'ensemble des réseaux à ressources renouvelables est noté $\mathcal{RRN}$.

La composition de deux réseaux à ressources renouvelables est un réseau à ressources renouvelables et donc un nombre arbitraire de réseaux peut être composé.

La composition de deux réseaux à ressources renouvelables se fait par fusion de places appartenant uniquement à RES. Dans ces hypothèses, la projection d'une séquence du langage de la composition des deux réseaux est une séquence du langage d'un composant (contrairement au cas général: cf. la proposition III-1).

Définition III-5 (Composition sur $\mathcal{RRN}$) *La composition asynchrone de deux réseaux à ressources renouvelables tels que*

$$RES_1 \cap RES_2 = P_1 \cap P_2$$

est définie par

$$(N_1; M_{01}; RES_1) \otimes (N_2; M_{02}; RES_2) = (N; M_0; RES)$$

où

- $(N; M_0) = (N_1; M_{01}) \otimes (N_2; M_{02})$
- $RES = RES_1 \cap RES_2 = P_1 \cap P_2$

Proposition III-4 (Fermeture de $\mathcal{RRN}$ par composition) *Soient deux réseaux à ressources renouvelables* $(N_i; M_{0i}; RES_i)$.

Alors $(N_1; M_{01}; RES_1) \otimes (N_2; M_{02}; RES_2) \in \mathcal{RRN}$ *et*

$$M_0 \xrightarrow{\sigma} M \Rightarrow \exists \sigma_i, \exists M_i, M_{0i} \xrightarrow{\sigma_i} M_i$$

$$\sigma_i = \sigma\downarrow_{T_i} \bigwedge M_i\downarrow_{P_i'} = M\downarrow_{P_i'} \bigwedge M_i\downarrow_{RES} \geq M\downarrow_{RES}$$

et plus précisément

$$M\downarrow_{RES} = M_1\downarrow_{RES} + M_2\downarrow_{RES} - M_0\downarrow_{RES}$$

Preuve On prouve par récurrence sur la longueur de σ l'assertion suivante ($\{i,j\} = \{1,2\}$):

Si $M_0 \xrightarrow{\sigma} M$, alors $M_{0i} \xrightarrow{\sigma_i} M_i$, où $\sigma_i = \sigma\downarrow_{T_i}$ et $M_i(p) = M(p)$ pour $p \in P'_i$, et $M_i(p) = M(p) - C_j(p,\sigma_j)$ pour $p \in RES$ (se rappeler que $C_j(p,\sigma_j) = M_j(p) - M_0(p) \leq 0$ sur RES, car $p \in RES$ est bornée par son marquage initial).

Pour $\sigma = \lambda$, c'est trivial.

Pas de récurrence. Supposons $M_0 \xrightarrow{\sigma} M \xrightarrow{t} M'$, $\sigma' = \sigma t$. Si $t \in T_1$, alors $M_1 \xrightarrow{t}$ et donc $M_{01} \xrightarrow{\sigma_1 t} M'_1$ et $M_{02} \xrightarrow{\sigma_2} M'_2 = M_2$ avec $M'_i\downarrow_{P'_i} = M'\downarrow_{P'_i}$; il reste à montrer $M_i(p) = M(p) - C_j(p,\sigma_j)$ pour $p \in RES$.

Si $t \notin {}^\bullet RES \cup RES^\bullet$ pas de problème.

Si $t \in {}^\bullet RES \cup RES^\bullet$, alors $M'(p) = M(p) + C_1(p,t)$ et $M'_1(p) = M_1(p) + C_1(p,t)$ pour $p \in RES$. Par conséquent, $M'_1(p) - M'(p) = M_1(p) - M(p) = C_2(p,\sigma_2) = C_2(p,\sigma'_2)$ pour $p \in RES$, et $M'_2(p) = M_2(p) = M(p) - C_1(p,\sigma_1) = M(p) + C_1(p,t) - C_1(p,\sigma_1 t) = M'(p) - C_1(p,\sigma'_1)$ pour $p \in RES$. □

Un blocage d'un réseau à ressources renouvelables n'est pas un marquage où toutes les transitions sont mortes mais un marquage qui ne permet pas de libérer toutes les ressources, ie, les ramener à leur état initial.

Donc un réseau à ressources renouvelables est dit sans R-interblocage si l'ensemble des états où les ressources sont libres est un espace d'accueil.

Définition III-6 (R-interblocage) $(N; M_0; RES) \in \mathcal{RRN}$ *est dit sans R-interblocage ssi*

$$FREE = \{M \in \mathbf{N}^P; \forall r \in RES, M(r) = M_0(r)\}$$

est un espace d'accueil de $(N; M_0)$.

III-3 Réseaux à Ressources Ordonnées

Dans cette section, nous définissons une sous-classe des réseaux à ressources renouvelables sans R-interblocage et telle que la composition de deux réseaux de cette classe appartienne à cette classe. Cette classe est l'ensemble des réseaux à ressources ordonnées.

Une solution simple et classique au problème de l'interblocage est la partition des ressources en classes ordonnées et l'obligation pour chaque processus de demander les ressources dont il a besoin en respectant l'ordre.

La solution que nous proposons consiste aussi à ordonner les ressources, mais le système n'est pas obligé de suivre un certain ordre dans ses demandes, mais

doit pouvoir toujours prendre une porte de sortie et libérer les ressources qu'il détient si jamais il y a risque d'interblocage, et dans sa "sortie," il peut continuer à demander des ressources mais en respectant l'ordre cette fois. Il existe donc des choix dans les suites de demandes de ressources, contrairement aux solutions classiques. Cette solution coïncide avec la solution classique pour les processus linéaires (séquentiels sans choix).

Cette condition est exprimée par une propriété d'espace d'accueil: il est toujours possible de libérer une ressource sans demander au cours de la libération des ressources de rang inférieur (mais avec la possibilité de demander des ressources de rang supérieur).

Définition III-7 (Réseaux à ressources ordonnées) *Un* réseau à ressources ordonnées *est un quadruplet* $(N; M_0; RES; (RES_i)_{i=1,n})$ *où*

- $(N; M_0; RES) \in \mathcal{RRN}$
- $(RES_i)_{i=1,n}$ *est une partition ordonnée de* RES
- $\forall i, \forall r \in RES_i, FREE(r) = \{M \in \mathbf{N}^P; M(r) = M_0(r)\}$ *est un* $T(r)$*-espace d'accueil, où* $T(r) = T \setminus (\bigcup_{j \leq i} RES_j^\bullet)$

L'ensemble des réseaux à ressources ordonnées est noté $\mathcal{RRO}$.

La figure III-4 montre un exemple de réseau à ressources ordonnées: il existe une séquence où S_1 est demandée avant S_2 ($t_1t_2t_3t_4$) et une autre où S_2 est demandée avant S_1 (t_5t_6); mais dans la séquence où S_2 est demandée avant S_1, il est toujours possible de continuer dans le bon ordre, en particulier de libérer S_2 en demandant S_3 (t_8t_9).

D'abord nous montrons qu'un réseau à ressources ordonnées (RRO) est sans R-interblocage (en tant que réseau à ressources renouvelables), puis que la composition de deux RRO ayant mêmes ressources avec le même ordre est un RRO, ce qui prouve l'absence d'R-interblocage lors de la composition.

Proposition III-5 ($\mathcal{RRO}$ et R-interblocage)
Si $(N; M_0; RES; (RES_i)_{i=1,n}) \in \mathcal{RRO}$, *alors* $(N; M_0; RES)$ *est sans R-interblocage.*

Preuve On montre par récurrence sur k que $E_k = \bigcap_{1 \leq j \leq k} \bigcap_{r \in RES_j} FREE(r)$ est un espace d'accueil.

$k = 1$. Comme $FREE(r)$ est un $(T \setminus RES_1^\bullet)$-espace d'accueil pour $r \in RES_1$, on peut ramener une ressource de rang 1 à son état initial sans changer l'état des autres ressources de rang 1 et donc $E_1 = \bigcap_{r \in RES_1} FREE(r)$ est un espace d'accueil.

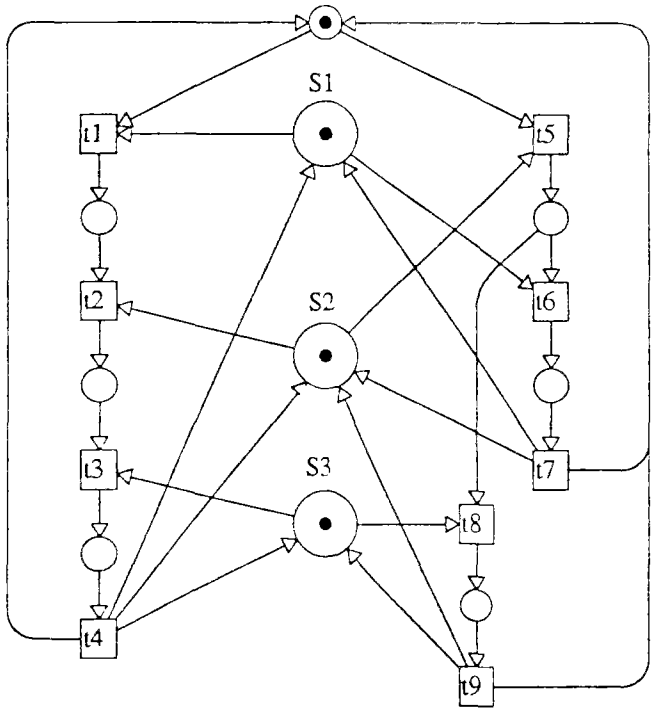


Figure III-4: Réseau à ressources ordonnées

Pas de récurrence. Si E_k est un espace d'accueil, comme

$$FREE(r) \text{ est un } (T \setminus \bigcup_{1 \leq j \leq k+1} RES^\bullet)\text{-espace d'accueil pour } r \in RES_{k+1}$$

on peut atteindre $\bigcap_{r \in RES_{k+1}} FREE(r)$ sans faire décroître le marquage des places de $\bigcup_{1 \leq j \leq k+1} RES_j$, et donc atteindre E_{k+1}. Donc E_n est un espace d'accueil. □

La composition de deux réseaux à ressources ordonnées est sans R-interblocage, car on peut toujours libérer les ressources de la classe la plus grande, disons de rang n, sans rien demander. Puis on peut libérer les ressources de rang $n-1$ sans demander de ressources de rang inférieur mais éventuellement en demandant des ressources de rang n. A la fin de cette opération, les ressources de rang $n-1$ sont libres mais celles de rang n ne le sont peut-être plus: mais on peut toujours libérer les ressources de rang n sans rien demander. Ainsi on atteint un état où les ressources des rangs n et $n-1$ sont libres. Et ainsi de suite...

Proposition III-6 ($\mathcal{RRO}$ et composition) *Soit deux réseaux à ressources ordonnées $(N_j; M_{0j}; RES; (RES_i)_{i=1,n})_{j=1,2}$ ayant mêmes ressources et mêmes partitions avec le même ordre, alors*

$$((N_1; M_{01}; RES) \otimes (N_2; M_{02}; RES); (RES_i)_{i=1,n}) \in \mathcal{RRO}$$

Preuve On montre par récurrence descendante sur $k \leq n$ que

$$E_k = \bigcap_{j=k}^{n} \bigcap_{r \in RES_j} FREE(r)$$

est un A_k-espace d'accueil où $A_k = T \setminus (\bigcup_{1 \leq l \leq k} RES_l^\bullet)$.

$k = n$. La preuve repose sur l'idée que chaque réseau peut libérer les ressources de rang n qu'il détient sans demander aucune ressource supplémentaire, et ainsi on atteint E_n.

Si $M_0 \xrightarrow{\sigma} M$, on veut montrer qu'il existe $s \in T_n^*$ telle que $M \xrightarrow{s} Q \in E_n$. D'après la proposition III-4. pour $i = 1, 2$, $M_{0i} \xrightarrow{\sigma_i}_i M_i$; puisque $(N_i; M_{0i}) \in \mathcal{RRO}$, il existe $\sigma_i' \in (T_i \setminus \bigcup_{1 \leq l \leq n} RES_l^\bullet)^*$, telle que

$$M_i \xrightarrow{\sigma_i'}_i M_i' \in FREE_i(RES_n)$$

Comme $M{\downarrow}_{P_i'} = M_1{\downarrow}_{P_i'}$ et σ_i' ne contient pas de transition $\in RES^\bullet$, alors

$$M \xrightarrow{\sigma_1'} M' \xrightarrow{\sigma_2'} Q \in FREE(RES_n)$$

D'où $s = \sigma_1'\sigma_2'$.

Pas de récurrence. Si $M_0 \xrightarrow{u} Q$, alors par hypothèse de récurrence, $Q \xrightarrow{v} M \in E_k$ avec $v \in A_k$. On note $s = uv$.

D'après la proposition III-4, pour $i = 1, 2$, $M_{0i} \xrightarrow{s_i}_i M_i$, avec $M_i \in E_k{\downarrow}_{P_i}$ et donc $M_i(r) = M_{0i}(r)$ pour $r \in \bigcup_{k \leq j \leq n} RES_j$.

Puisque les deux réseaux sont dans $\mathcal{RRO}$, pour $r_0 \in RES_{k-1}$, il existe σ_i, telle que $M_i \xrightarrow{\sigma_i}_i M_i' \in FREE_i(r_0)$, avec $\sigma_i \in A_{k_1}$.

Alors $M \xrightarrow{\sigma_1\sigma_2} M' \in FREE(r_0)$, puis par application de l'hypothèse de récurrence, il existe $w \in A_k \subseteq A_{k-1}$, telle que $M' \xrightarrow{w} M'' \in E_k \cap FREE(r_0)$: donc à partir de $M \in E_k$ on atteint $E_k \cap FREE(r_0)$ par la séquence $\sigma_1\sigma_2 w \in A_{k-1}$.

On recommence cette opération pour les autres places de RES_{k-1}, et on obtient ainsi une séquence $y \in A_{k-1}$ telle que $Q \xrightarrow{y} Q' \in E_{k-1}$. □

III-4 Composition de réseaux colorés

Le seul point à ajouter à la définition de la composition asynchrone de réseaux ordinaires pour obtenir celle des réseaux colorés est la définition des domaines de couleurs: on prend $D(p) = D_i(p)$ si $p \in P_i$, en exigeant que les places partagées

aient même domaine dans les deux réseaux, c-à-d $\forall p \in S, D_1(p) = D_2(p)$. Dans ces conditions le réseau composé déplié est égal à la composition des réseaux dépliés.

Pour les réseaux à ressources ordonnées, on prend exactement la même définition. Il reste à montrer qu'à tout réseau coloré à ressources ordonnées, on peut associer un réseau ordinaire à ressources ordonnées. Si RES est l'ensemble des ressources du réseau coloré, et $(RES_i)_{i=1,n}$ sa partition ordonnée, on prend pour le réseau déplié, $RES' = \{(p,c); p \in RES, c \in D_i(p)\}$ et $RES'_i = \{(p,c); p \in RES_i, c \in D_i(p)\}$ et on vérifie qu'alors le même ordre convient.

Le multiplexage d'unités entre plusieurs utilisateurs revient souvent à partager les places représentant l'état oisif de ces unités entre plusieurs réseaux identiques. On obtient ainsi une coloration supplémentaire du réseau d'origine: le domaine de toutes les places sauf celles partagées, devient égal à $D(p) \times \{1, \ldots, n\}$.

Conclusion

La composition par fusion de places a pour avantage essentiel de conserver les flots. Toutefois, dans le cas général, il n'y a pas de relation entre les comportements des composants et celui du réseau total. Il n'est donc possible d'obtenir des propriétes de conservation qu'en introduisant des conditions supplémentaires sur le comportement vis-à-vis des places partagées. Nous avons obtenu la conservation des espaces d'accueil (Proposition III-3) quand leur accessibilité est indépendante des places partagées, et la conservation des comportements (Proposition III-4) quand les places partagées sont bornées par leur marquage initial (réseaux à ressources renouvelables).

Nous arrivons à déduire l'absence de R-interblocage du réseau total uniquement en considérant les composants, dans la classe des réseaux à resssources ordonnées. Cette classe est assez restreinte, mais la preuve modulaire de l'absence de blocage est intrinsèquement difficile. La recherche d'autres propriétés plus faciles devrait aboutir à des classes de réseaux plus larges.

Des résultats de ce chapitre sont appliqués à la composition des réseaux réentrants, classe de réseaux définie dans le chapitre suivant.

Chapitre IV

Réseaux réentrants

Introduction

Dans le deuxième chapitre, lors de l'étude du remplacement de sous-réseaux ouverts au moyen de l'SST-équivalence, pour pallier la difficulté de vérifier l'équivalence de deux réseaux à interface ouverte, nous avons évoqué la définition de classes particulières de réseaux pour lesquelles cette vérification serait simple.

Nous avions déjà défini les réseaux à interface ouverte robustes pour obtenir la fermeture de l'ensemble des systèmes à interface ouverte par remplacement de tels sous-réseaux, et obtenir un théorème de remplacement pour les OI-systèmes. Mais cette classe ne simplifie en rien la vérification de l'équivalence.

Nous avons donc cherché à définir une sous-classe des OI-réseaux robustes ayant assez de contraintes structurelles qui normalisent le comportement pour simplifier la vérification de l'SST-équivalence, et en plus, ayant de bonnes propriétés pratiques de modélisation. Nous avons alors obtenu la classe des réseaux réentrants.

Les réseaux réentrants sont le résultat de la combinaison des OI-réseaux robustes définis dans le deuxième chapitre, et des réseaux à terminaison propre définis dans BRAMS[11].

Un OI-réseau robuste a un ensemble de places d'interface ITF, et un ensemble d'états stables STB, tels que STB soit un espace d'accueil accessible sans franchir des transitions sorties des places d'interface.

Un réseau à terminaison propre est un réseau ayant une place initiale p_i et une place finale p_f, telles que, à partir du marquage initial où seule p_i est marquée, l'état final où seule p_f est marquée soit un état d'accueil.

Nous généralisons alors les réseaux à terminaison propre en considérant une interface composée d'un ensemble INI de places initiales et un ensemble FIN de places finales. De plus, INI et FIN sont partitionnées en sous-ensembles, tels

qu'à un sous-ensemble PI de INI corresponde un sous-ensemble PF de FIN, et une marque dans une place de PI puisse aller dans n'importe quelle place de PF.

Les états stables des réseaux réentrants sont alors les états terminaux qui, au lieu d'être caractérisés par "toutes les places du réseau sont vides", sont définis par: pour chaque couple (PI, PF), il existe un flot dont le support contient $PI \cup PF$, et les états stables correspondent aux états où toutes les places internes des supports sont vides.

Intuitivement, un réseau réentrant s'interprète comme un serveur qui reçoit des requêtes dans des places initiales et délivre des réponses dans des places finales: ces places constituent l'interface du réseau. Plusieurs services sont disponibles, et pour chaque service, il existe plusieurs ports (places) où on peut mettre les requêtes et plusieurs ports de sortie, et il existe un flot dont le support contient les places d'interface correspondantes.

La réentrance (absence de blocage quel que soit le nombre des requêtes et quel que soit leur distribution) est exprimée par une propriété d'espace d'accueil: on peut toujours vider les places intermédiaires des flots.

De plus, la réponse est indéterministe, c-à-d, quel que soit l'état interne du serveur, il est toujours possible sans autre intervention de l'environnement de traiter cette requête et de lui donner n'importe quelle réponse. Mais ceci implique que tous les traitements soient indépendants: il n'y a pas de synchronisation possible entre deux exécutions d'un même service ou de deux services différents.

Finalement, à travers les réseaux réentrants, nous obtenons bien des réseaux à interface ouverte robustes dont l'SST-équivalence dépend uniquement de leur structure, ce qui supprime la complexité de vérifier l'existence d'une bisimulation. Mais évidemment, il reste la complexité de vérifier la réentrance d'un réseau, qui s'exprime par une propriété d'espace d'accueil.

Un autre avantage des réseaux réentrants est la possibilité de les composer par fusion de places, et ainsi de les construire hiérarchiquement: ceci nous permet d'appliquer certains résultats du chapitre III. Ainsi c'est une classe de réseaux qui se prêtent bien à l'analyse modulaire, que ce soit par remplacement de sous-réseaux ou par composition.

Dans[13,15], nous avons donné une définition légèrement différente des réseaux réentrants, et directement pour les réseaux colorés. Le théorème de remplacement y a été montré pour une autre équivalence, appelée OH-éq, proche de l'SF-éq avec en plus un étiquetage des transitions. Nous abandonnons l'étiquetage de transitions, car par une succession de remplacements de sous-réseaux ouverts, on peut obtenir un réseau qui n'a plus aucune transition commune avec le réseau d'origine.

La première section donne la définition d'un réseau réentrant et sa justification intuitive, et montre que les réseaux réentrants sont des réseaux à interface robustes. Deux sous-classes de réseaux réentrants sont définies: les réseaux sans boucles et ceux sans mémoire. Pour ces sous-classes, la vérification de la réentrance est plus simple que dans le cas général.

Dans la deuxième section, nous montrons que l'SST-équivalence de deux réseaux réentrants est caractérisable structurellement d'après le graphe des réseaux.

La troisième section est consacrée à la composition de réseaux réentrants par différents types de fusions de places: fusion des places d'interface, mise en anneau de réseaux réentrants et partage de places représentant des ressources (différentes des places d'interface).

IV-1 Définition et propriétés

IV-1.1 Définition et motivations

Certaines des conditions vérifiées par un réseau réentrant sont structurelles—chemins, flots, et d'autres sont comportementales—espaces d'accueil. Les commentaires suivent la définition et font le lien avec l'interprétation intuitive d'un réseau réentrant.

Définition IV-1 (Réseau réentrant) *Un* réseau réentrant *est un triplet* $RN = (N; SV; M_0)$ *où*

(i) $N = (P, T; W)$ *est un réseau P/T*

(ii) $SV \subseteq \mathcal{B}(P) \times \mathcal{B}(P)$ *et* SV *est bijective (rappelons que* $\mathcal{B}(P)$ *est l'ensemble des parties de* P*).*
On note $INI = \bigcup_{PI \in dom(SV)} PI$ *l'ensemble des places initiales,*
et $FIN = \bigcup_{PF \in cod(SV)} PF$ *l'ensemble des places finales.*
$dom(SV)$ *(resp.* $cod(SV)$*) forme une partition de* INI *(resp.* FIN*).*
$ITF = INI \cup FIN$ *est l'ensemble des places d'interface.*
INI *et* FIN *vérifient les deux conditions suivantes:*

- $^\bullet INI = \emptyset$
- *Pour tout* $PF \in cod(SV)$*, il n'existe pas de chemin entre deux places de* PF

(iii) *Si* $SV = \{(INI_i, FIN_i); i \in 1..n\}$*, il existe n flots* f_i*, tels que pour tout i, le support de* f_i*, noté* SPR_i*, vérifie* $SPR_i \cap ITF = ITF_i = INI_i \cup FIN_i$ *et ses*

coefficients sur ITF_i soient égaux:

$$\forall t \in T, \sum_{p \in ITF_i} C(p,t) + \sum_{p \in SPR_i \setminus ITF_i} c_i(p)C(p,t) = 0$$

où $c_i(p) = f_i(p)/f_i(p'), p' \in ITF_i$

(iv) *On note l'ensemble des états stables:*

$$STB = \{M \in \mathbf{N}^{P \setminus ITF}; \forall i, M{\downarrow}_{SPR_i \setminus ITF} = 0\}$$

Pour tout $m_0 \in \mathbf{N}^{ITF}$, $STB{\uparrow}^P$ *est un* $(T \setminus ITF^\bullet)$*-espace d'accueil de* $(N; M_0 + m_0)$

(v) $M_0 \in STB$

(vi) $\forall m_0 \in \mathbf{N}^{ITF}, \forall M \in ACC(N; M_0 + m_0) \cap STB, \forall (PI, PF) \in SV, \forall pi \in PI, \forall pf \in PF,$
si $M(pi) > 0$ *alors*

$$\exists s \in (T \setminus FIN^\bullet)^*, M \xrightarrow{s} M' \in STB, M'{\downarrow}_{ITF} = M{\downarrow}_{ITF} - pi + pf$$

L'ensemble des réseaux réentrants est noté $\mathcal{RN}$.

(ii) SV est un ensemble de couples d'ensembles (places initiales, places finales): chaque couple représente un service. Les ensembles des ports d'entrée et des ports de sortie des différents services sont disjoints (propriété de la partition).

Les places initiales n'ont pas de transition entrée: le serveur ne peut produire une requête ni remettre une requête qu'il a commencée à traiter dans son état initial.

Les places finales peuvent avoir des transitions sorties: le serveur peut reprendre un résultat et le retraiter mais il doit remettre le nouveau résultat sur le même port (pas de chemin entre places finales).

Ces deux conditions sont justifiées par le lemme IV-2 qui dit que, dans une séquence menant d'un état stable à un autre, tout suffixe de cette séquence diminue le marquage des places initiales et augmente celui des places finales.

(iii) Le flot représente le chemin de traitement d'une requête: les places intermédiaires du flot représentent les exécutions en cours. La propriété des coefficients implique qu'à une requête correspond une seule réponse et vice versa. Noter que les flots ne sont pas nécessairement uniques.

(iv) L'ensemble des états stables (états terminaux) est défini en fonction des supports des flots.

Le serveur ne se bloque jamais: il est toujours capable de terminer toutes les exécutions en cours (vider les places internes des supports des flots) et d'atteindre un état stable oisif, sans avoir besoin de consommer de nouvelles requêtes ni de reprendre certains résultats (sans franchir des transitions dans $ITF^{\bullet}$).

(vi) C'est une condition qui normalise le comportement des réseaux réentrants et rend donc leur comparaison plus aisée.

Quand le serveur est dans un état stable, une marque dans une place initiale peut être acheminée dans n'importe quelle place finale du service correspondant. C'est la propriété la plus contraignante.

D'abord, il ne peut y avoir de synchronisation entre les différentes requêtes, comme une requête en deux temps: demande d'accès puis envoi des paramètres de la requête.

La figure IV-1 montre un exemple: le réseau N_1 est réentrant, ses deux services sont (p_1, p_3) et (p_2, p_4). Une marque en p_1 peut toujours être acheminée vers p_3, sans besoin de marques en p_2.

Le réseau N_2 n'est pas réentrant car il existe une synchronisation entre les deux services: une fois qu'on a exécuté (p_1, p_3) deux fois, on ne peut plus recommencer tant qu'on a pas exécuté (p_2, p_4); et inversement, (p_2, p_4) n'est possible que si (p_1, p_3) a été exécuté auparavant.

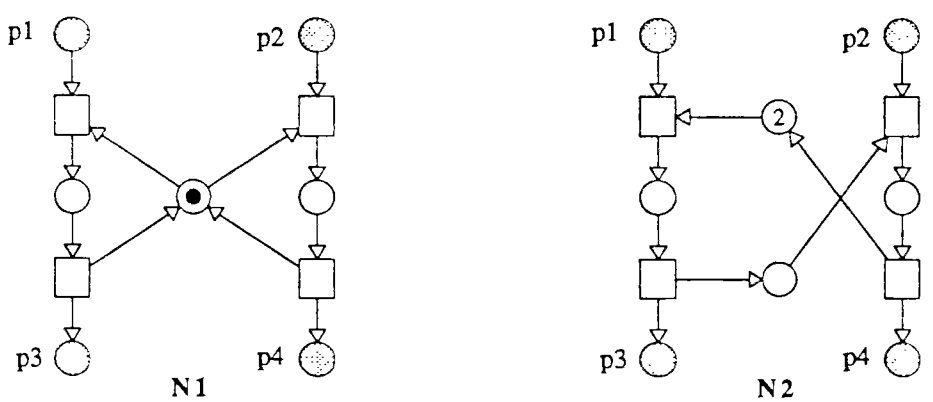


Figure IV-1: Synchronisation entre deux services

La deuxième contrainte imposée par (vi) est l'indéterminisme de la distribution des résultats: le serveur est toujours libre de donner la réponse qu'il veut à une requête donnée, alors que d'habitude on ne refuse une requête que pour certains motifs comme le manque de place dans le tampon.

La figure IV-2 montre un exemple: le réseau N_1 est réentrant, son unique service est $(p_1, \{p_2, p_3\})$; Une marque en p_1 peut aller librement en p_2 ou en p_3.

En revanche, N_2 n'est pas réentrant, car une marque en p_1 ne peut aller en p_2 que s'il y a déjà une exécution en cours: donc il y a une synchronisation entre les exécutions d'un même service.

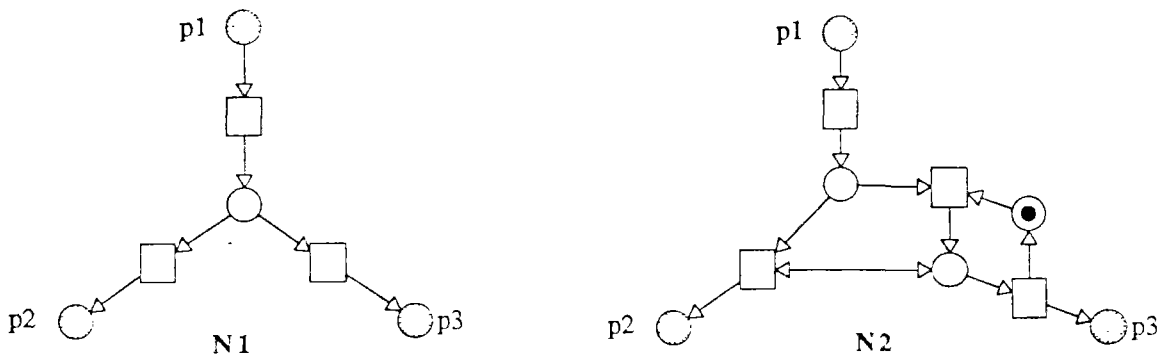


Figure IV-2: Synchronisation entre les exécutions d'un même service

IV-1.2 Robustesse des réseaux réentrants

Nous commençons par montrer qu'un réseau réentrant appartient bien à la classe des réseaux à interface ouverte robuste définie dans le deuxième chapitre (rappelons qu'ils s'agit d'OI-réseaux dont les états stables sont accessibles sans franchir des transitions sorties des places d'interface).

Dans la définition IV-1, le point (iii) énonce l'existence des flots dont les supports SPR_i contiennent les couples de sous-ensembles de l'interface, puis dans (iv), nous définissons l'ensemble STB des états stables en fonction de ces supports.

Or ces flots ne sont pas forcément uniques, et il faut donc montrer que l'ensemble STB des états stables est lui unique et ne dépend pas du choix des supports SPR_i.

Proposition IV-1 (Unicité de STB) *Un réseau réentrant est bien défini car STB est unique quand il existe et ne dépend pas du choix des SPR_i.*

Preuve On suppose qu'il existe un autre STB' associé à une autre famille de flots $(f'_i)_{i=1,n}$. On note $INR_i = SPR_i \setminus ITF_i$, et les noms primés se rapportent aux flots f'_i.

D'après les relations vérifiées par les flots ((iii) de la définition IV-1), on a

$$\forall t \in T, \sum_{p \in INR_i} c_i(p)C(p,t) = \sum_{p \in INR'_i} c'_i(p)C(p,t)$$

et comme d'après (v), $M_0 \in STB \cap STB'$, alors

$$\forall i, \sum_{p \in INR_i} c_i(p) M_0(p) = \sum_{p \in INR'_i} c'_i(p) M_0(p) = 0$$

et donc pour tout i,

$$\sum_{p \in INR_i} c_i(p) M(p) = \sum_{p \in INR'_i} c'_i(p) M(p)$$

est un invariant du réseau $(N; M_0 + m_0)$ pour $m_0 \in \mathbf{N}^{ITF}$. Par conséquent, pour $M \in ACC(N; M_0 + m_0)$, on a

$$\begin{aligned} M \in STB &\Leftrightarrow \forall i, \forall p \in INR_i, M(p) = 0 \\ &\Leftrightarrow \forall i, \forall p \in INR'_i, M(p) = 0 \\ &\Leftrightarrow M \in STB' \end{aligned}$$

et donc $STB' = STB$. □

La proposition suivante montre qu'une fois l'interface fixée, il existe une unique façon de la partitionner.

Proposition IV-2 (Unicité de SV) *Si $(N; SV; M_0)$ et $(N; SV'; M_0)$ sont deux réseaux réentrants ayant même réseau marqué sous-jacent et tels que $ITF = ITF'$, alors $SV = SV'$, c-à-d, il existe une unique partition possible de l'interface.*

Preuve Si ITF est fixé, alors INI est FIN sont déterminés de façon unique par $INI = \{p \in ITF; {}^\bullet p = \emptyset\}$ et $FIN = ITF \setminus INI$ à cause de (ii) et (vi). Il reste à montrer l'unicité de SV.

Soit SV_1 et SV_2 tels que $(N; SV_i; M_0)$ soit un réseau réentrant pour $i = 1, 2$. Soit $p \in INI$: il existe $PI_i \in dom(SV_i)$ tels que $p \in PI_i$. D'après la définition des réseaux réentrants, il existe PF_i tels que $PF_i \in cod(SV_i)$, et deux flots dont les supports SPR_i sont tels que $(PI_i \cup PF_i) \subseteq SPR_i$.

On note STB_i l'espace stable correspondant à SV_i.

Nous allons montrer que $(PI_1, PF_1) = (PI_2, PF_2)$.

On prouve d'abord $PF_1 = PF_2$. Soit $q \in PF_1$.
D'après la réentrance, il existe $\sigma \in (T \setminus ITF^\bullet)^*$ telle que $M_0 + p \xrightarrow{\sigma} M' + q$ où $M' \in STB_1 \wedge M'{\downarrow_{ITF}} = 0$.
Il existe $PF'_2 \in cod(SV_2)$ tel que $q \in PF'_2$ et un flot dont le support est $SPR'_2 \supseteq PI'_2 \cup PF'_2$ associé à STB_2.
On en déduit que $p \in SPR'_2$ (sinon q n'aurait pas pu être marquée par σ, et ne pas

oublier que $M_0 \in STB_1 \cap STB_2$), puis $p \in PI_2 \cap PI_2'$: par conséquent, $PI_2 = PI_2'$ et $PF_2 = PF_2'$, et donc $q \in PF_2$. Ainsi $PF_1 \subseteq PF_2$. On montre symétriquement que $PF_2 \subseteq PF_1$, d'où $PF_1 = PF_2$.

Montrons maintenant que $PI_1 = PI_2$. Notons $PF = PF_1 = PF_2$; on a $(PI_i, PF) \in SV_i$. Soit $p \in PI_1$ et $q \in PF$. De nouveau, l'existence d'une séquence qui amène une marque de p vers q, et l'existence des flots impliquent que $p \in PI_2$, et donc $PI_1 \subseteq PI_2$. Par symétrie, $PI_2 \subseteq PI_1$ et finalement $PI_1 = PI_2$.

Il reste à déduire que $SV_1 = SV_2$. Si $PI_1 \in dom(SV_1)$, on prend $p \in PI_1$, il existe $PI_2 \in dom(SV_2)$ tel que $p \in PI_2$. D'après ce qui précède, $PI_1 = PI_2$ et $PF_1 = PF_2$. Donc $SV_1 \subseteq SV_2$, et par symétrie $SV_1 = SV_2$. □

Corollaire IV-1 *Si $RN = (N; SV; M_0) \in \mathcal{RN}$ alors $(N; ITF; STB{\downarrow}_{P \setminus ITF}; M_0)$ est un réseau à interface ouverte robuste. L'association est bien définie et injective, et donc on peut écrire $\mathcal{RN} \subseteq \mathcal{ROI}$.*

Preuve Le fait que $(N; ITF; STB; M_0)$ soit un réseau à interface robuste découle immédiatement de la définition. L'association est bien définie à cause de l'unicité de STB, et elle est injective, car deux réseaux réentrants ayant même réseau marqué sous-jacent et même interface sont égaux (proposition IV-2). □

IV-1.3 Réseaux sans boucle et sans mémoire

Nous définissons deux sous-classes des réseaux réentrants qui permettent de réduire l'ensemble des marquages pour lequel il faut vérifier la propriété d'accueil: on la vérifie pour un marquage quelconque de INI au lieu de ITF. Ces deux sous-classes sont les réseaux sans boucle et les réseaux sans mémoire.

Les places finales des réseaux sans boucle ne peuvent avoir de transitions sorties: les marques qui atteignent les places finales ne peuvent revenir en arrière.

Définition IV-2 (Réseau sans boucle) *Un réseau réentrant $(N; SV; M_0)$ est dit sans boucle si $FIN^\bullet = \emptyset$.*

L'ensemble des réseaux réentrants sans boucle est noté $\mathcal{LCS}$.

Un réseau sans mémoire peut revenir à son état initial quand il n'y a pas d'exécution en cours (mais ses places finales peuvent avoir des transitions en sortie).

Définition IV-3 (Réseau sans mémoire) *Un réseau réentrant $(N; SV; M_0)$ est dit sans mémoire si pour tout $m_0 \in \mathbf{N}^{INI}$, $M_0\uparrow^P$ est un $(T \setminus ITF^\bullet)$-espace d'accueil de $(N; M_0 + m_0)$.*

L'ensemble des réseaux réentrants sans mémoire est noté $\mathcal{MLS}$.

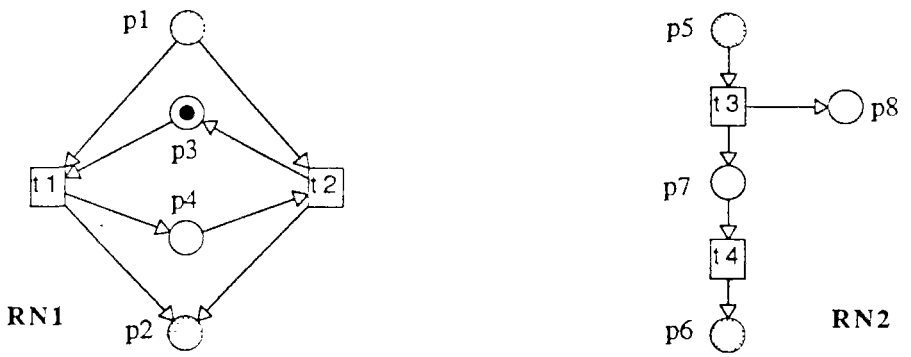


Figure IV-3: Réseaux réentrants avec mémoire

La figure IV-3 montre deux contre-exemples. Pour le réseau RN_1, $SV = \{(p_1, p_2)\}$ et $STB = \mathbf{N}^{\{p_3,p_4\}}$. Après franchissement de t_1 on atteint un état stable, mais il n'est plus possible alors de revenir au marquage initial sans franchir des transitions dans $ITF^\bullet$.

Pour le réseau RN_2, $SV = \{p_5, p_6\}$ et $STB = \{M; M(p_7) = 0\}$. Après chaque début de service (franchissement de t_3), une marque s'ajoute dans la place p_8: on ne peut pas ramener cette cette place à son état initial (marquage nul).

Ces réseaux réentrants ont en commun la propriété suivante: l'arrivée dans leurs places finales de marques qui ne sont pas passées par les places initiales ne les destabilise pas et il suffit alors de vérifier la propriété d'accueil pour des marquages appartenant à $\mathbf{N}^{INI}$ au lieu de $\mathbf{N}^{ITF}$.

Proposition IV-3 *Si $RN \in \mathcal{LLS} \cup \mathcal{MLS}$ alors il suffit de vérifier (iv) dans la définition IV-1 pour $m_0 \in \mathbf{N}^{INI}$.*

Preuve La propriété est immédiate pour $RN \in \mathcal{LLS}$.

Soit $RN \in \mathcal{MLS}$. Il faut montrer que si pour tout $m_0 \in \mathbf{N}^{INI}$, STB est un $(T \setminus ITF^\bullet)$-espace d'acceuil de $(N; M_0 + m_0)$, alors il en est de même pour $m_0 \in \mathbf{N}^{ITF}$.

Si $m_0 \in \mathbf{N}^{ITF}$, alors $m_0 = m_{01} + m_{02}$ où $m_{01} \in \mathbf{N}^{INI}$ et $m_{02} \in \mathbf{N}^{FIN}$.
On considère un marquage $m'_0 \in \mathbf{N}^{INI}$ vérifiant

$$\forall (PI, PF) \in SV, \sum_{p \in PI} m'_0(p) = \sum_{p \in PI} m_{01}(p) + \sum_{p \in PF} m_{02}(p)$$

c-à-d, on ajoute aux places initiales de chaque flot, les marques de ses places finales. Dans $(N; SV; M_0 + m'_0)$, par application de (vi) autant de fois qu'il le faut, on amène des marques de PI vers PF pour $(PI, PF) \in SV$, de façon à atteindre le marquage $M' + m_0$, où $M' \in \mathbf{N}^{P \setminus ITF}$.

Comme $RN \in \mathcal{MLS}$, il existe $\sigma \in (T \setminus ITF^\bullet)^*$ telle que $(M' + m_0 \xrightarrow{\sigma} (M_0 + m_0)$: par conséquent, si STB est un $(T \setminus ITF^\bullet)$-ea de $(N; M_0 + m'_0)$, il en est de même pour $(N; M_0 + m_0)$ puisque $M_0 + m_0 \in ACC(N; M_0 + m'_0)$. □

IV-2 Equivalence de réseaux réentrants

En tant que réseaux à interface ouverte robustes, les réseaux réentrants sont candidats à être utilisés dans une modélisation hiérarchique construite par remplacements successifs de serveurs par des serveurs SST-équivalents. Les contraintes posées sur les réseaux réentrants rendent la vérification de l'équivalence de deux réseaux simple: ils sont équivalents ssi ils rendent les mêmes services, ce qui se traduit par: ils ont la même interface et la même partition de cette interface entre les différents services. C'est la contrainte (vi) qui normalise les services en imposant la distribution indéterministe des résultats qui permet cette simplification. Dans ces conditions tout réseau réentrant admet un équivalent canonique.

Pour démontrer ce résultat (Théorème IV-1), nous avons besoin de deux lemmes qui expliquent pourquoi nous avons exclu dans la définition des réseaux réentrants les arcs entrant dans les places initiales et les chemins entre places finales.

Le premier lemme est un résultat général dans les réseaux de Petri: si une séquence diminue le marquage d'une place appartenant au support d'un flot, alors il existe un chemin de cette place vers une autre place dont le marquage augmente.

Lemme IV-1 *Si $p \in SPR$ support d'un flot, et si $M \xrightarrow{\sigma} M'$ tel que $M'(p) < M(p)$, alors il existe $p' \in SPR$, telle que $p' \neq p$, $M'(p') > M(p')$ et il existe un chemin de p vers p'.*

Preuve On va montrer que si $f : P \to \mathbf{N}$ a SPR comme support et

$$\forall t \in T, \sum_{p \in P} f(p)C(p,t) = 0$$

et $p_0 \in SPR$ telle que

$$\sum_{t \in T} a(t)C(p_0,t) < 0$$

où

$$a : T \to \mathbf{N} \text{ de support } T_a$$

alors, il existe $p_1 \in SPR$,

$$\sum_{t \in T} C(p_1, t) > 0$$

et il existe un chemin $q_0 t_1 q_1 \ldots t_n q_n$ avec $q_0 = p_0$, $q_n = p_n$, $\forall i, t_i \in T_a$, $C(q_{i-1}, t_i) < 0$ et $C(q_i, t_i) > 0$.
La preuve se fait par récurrence sur $|T_a|$.

$|T_a| = 0$: vrai par vacuité.

$|T_a| \geq 1$.

On considère S_1, l'ensemble des chemins élémentaires commençant en p_0, ne contenant que des transitions de T_a:

$$S_1 = \{s; s = q_0 t_1 q_1 \ldots t_n q_n \text{ chemin élémentaire vérifiant } \mathbf{H}\}$$

où

$$\mathbf{H} : (q_0 = p_0) \wedge (t_i \in T_a) \wedge (C(q_{i-1}, t_i) < 0) \wedge (C(q_i, t_i) > 0)$$

S_1 est non vide, car il existe $C(p_0, t) < 0$ et donc à cause du flot, il existe p' tel que $C(p', t) > 0$. On note alors

$$P_1 = \{p \in SPR; \exists s \in S_1, p \text{ apparaît dans } s\}$$

$$T_1 = \{t \in T_a; \exists s \in S_1, t \text{ apparaît dans } s\}$$

Montrons par l'absurde qu'il existe $p_1 \in P_1$ telle que

$$\sum_{t \in T} a(t) C(p_1, t_1) > 0$$

Donc on suppose $\forall p \in P_1, \sum_{t \in} a(t) C(p, t) \leq 0$.

A cause du flot, puisque $\sum_{t \in T} a(t) C(p_0, t) < 0$, il existe $p' \in SPR$ telle que $\sum_{t \in T} a(t) C(p', t) > 0$.

On en déduit $p' \notin P_1$ et $\exists t' \in T_a, C(p', t') > 0$. Nécessairement, $t' \notin T_1$ sinon il existerait un chemin de p_0 à p' et $p' \in P_1$: par conséquent, $\forall p \in P_1, C(p, t') \geq 0$ (sinon $t' \in T_1$.)

On considère $b : T \to \mathbf{N}$, définie par

$$b(t) = \begin{cases} a(t) & \text{si } t \neq t' \\ 0 & \text{si } t = t' \end{cases}$$

Alors le support de b vérifie $T_b = T_a \setminus \{t'\}$, et $\sum_{t\in T} b(t)C(p_0,t) < 0$ puisque $C(p,t') \geq 0$ pour tout $p \in P_1$.

Si on note S_1', P_1' et T_1' les ensembles correspondant à b, comme $t' \notin T_1$, alors $S_1' = S_1$, $P_1' = P_1$ et $T_1' = T_1$.

D'après l'hypothèse de récurrence appliquée à b, il existe $p_1 \in P_1' = P_1$ telle que $\sum_{t\in T} b(t)C(p_1,t) > 0$.

Il vient $\sum_{t\in T} a(t)C(p_1,t) > 0$ puisque $C(p,t') \geq 0$ pour $p \in P_1$: il y a donc contradiction. D'où, $\exists p_1 \in P_1, \sum_{t\in T} a(t)C(p_1,t) > 0$.

Finalement, pour montrer le lemme, il suffit de remarquer que $C(p,\sigma) = \sum_{t\in T} \bar{\sigma}(t)C(p,t)$ ($\bar{\sigma}(t)$ représente le nombre d'occurrences de t dans σ). □

Dans la suite, on appelle séquence stable une séquence qui transforme un état stable en un état stable.

Définition IV-4 (Séquence stable) *Soit $(N; SV; M_0)$ un réseau réentrant où $N = (P, T; W)$ et $\sigma \in T^*$. Alors σ est appelée* séquence stable *s'il existe deux marquages M et M' dans $STB\uparrow^P$ tels que $M \xrightarrow{\sigma} M'$.*

Le lemme suivant établit qu'un suffixe d'une séquence stable ne peut qu'augmenter le marquage d'une place finale. C'est pour obtenir ce résultat que nous avons exclu les chemins entre les places finales d'un même flot.

Lemme IV-2 *Soit $\sigma = t_1 \ldots t_n$ une séquence stable d'un réseau réentrant. Alors*

$$\forall k \in 1..n, \forall p \in FIN, \sum_{j=k}^{n} C(p,t_j) \geq 0$$

$$\forall (PI,PF) \in SV, \forall k \in 1..n, \begin{cases} \sum_{p\in PI} W(p,t_1 \ldots t_k) & \geq & \sum_{p\in PF} C(p,t_1 \ldots t_k) \\ \sum_{p\in PI} W(p,\sigma) & = & \sum_{p\in PF} C(p,\sigma) \end{cases}$$

(sans oublier que $W(p,\sigma) = -C(p,\sigma)$ pour $p \in INI$).

Preuve Soit $(PI,PF) \in SV$ et SPR le support du flot correspondant.

Si $\sigma = t_1 \ldots t_n$ est une séquence stable, alors $\sum_{j=k}^{n} C(p,t_j) \leq 0$ pour $p \in SPR \setminus PF$, car σ mène à un marquage stable, c-à-d vérifiant $M(p) = 0$ pour $p \in SPR \setminus PF$; et $C(p,\sigma) \leq 0$ pour $p \in PI$ car ${}^\bullet INI = \emptyset$.
Donc, $\sum_{j=k}^{n} C(p,t_i) \leq 0$ pour $p \in SPR \setminus PF$.

Supposons qu'il existe $p_0 \in PF$ telle que $\sum_{j=k}^{n} C(p_0,t_j) < 0$, alors il existe $p_1 \in SPR$ telle que $\sum_{j=k}^{n} C(p_1,t_j) > 0$ et il existe un chemin de p_0 vers p_1. Or si

$\sum_{j=k}^{n} C(p_1, t_j) > 0$, alors $p_1 \in PF$: mais il n'existe pas de chemin de $p_0 \in PF$ vers $p_1 \in PF$. D'où, $\forall p \in FIN, \sum_{j=k}^{n} C(p, t_j) \geq 0$. □

Nous sommes en mesure maintenant, d'énoncer et de prouver le théorème sur l'équivalence de deux réseaux réentrants: deux réseaux réentrants sont SST-équivalents ssi ils rendent les mêmes services, c-a-d, ils ont même interface et même partition de cette interface.

Théorème IV-1 (SST-équivalence sur $\mathcal{RN}$) *Soit $(N_i; SV_i; M_{0i})$ deux réseaux réentrants.*

$$(N_1; SV_1; M_{01}) \equiv_{SST} (N_2; SV_2; M_{02}) \Leftrightarrow SV_1 = SV_2$$

Une conséquence immédiate de ce résultat est l'existence d'un équivalent canonique d'un réseau réentrant. Comme nous prouvons le théorème en montrant qu'un réseau réentrant est équivalent à sa forme canonique, nous énonçons d'abord le corollaire suivant.

Corollaire IV-2 (Equivalent canonique) *Tout réseau réentrant admet un équivalent canonique construit comme le montre la figure IV-4: c'est un réseau ayant autant de composantes connexes que d'éléments dans $SV = \{(INI_i, FIN_i); i \in 1..n\}$. Les places de chaque composante connexe sont $INI_i \cup FIN_i \cup \{p_i\}$ telles que, il existe une transition entre chaque place de INI_i et p_i, et une transitin entre p_i et chaque place de FIN_i. Les places p_i sont appelées places internes ou intermédiaires.*

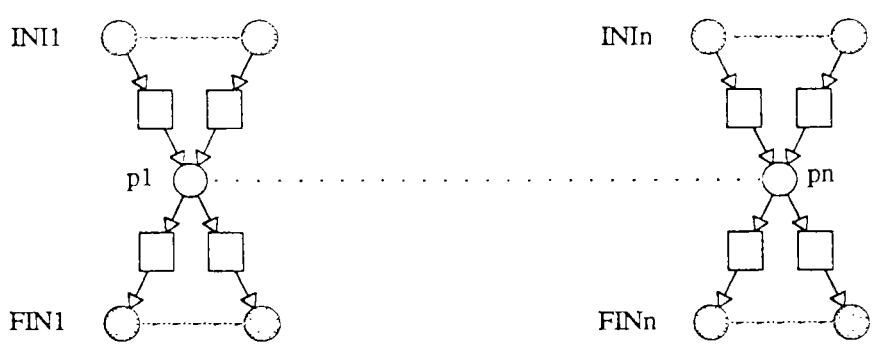


Figure IV-4: Equivalent canonique

Preuve du théorème IV-1

1) $SV_1 = SV_2 \Rightarrow RN_1 \equiv_{SST} RN_2$. On va montrer que $\mathcal{R} \subseteq STB_1 \times STB_2$ est une SST-bisimulation (tous les états stables sont équivalents).

Soit $\sigma_1 = t_1 \dots t_m$ une séquence stable de RN_1. On va construire σ_2 séquence stable de RN_2 telle que $\sigma_1 \leq_{ITF} \sigma_2$, où RN_2 est l'équivalent canonique de la figure IV-4.

Si x et y sont deux séquences, $x \subseteq y$ signifie que x est une sous-séquence de y. A chaque t_i de σ_1, on associe $w_i = u_i v_i \in T_2^*$ ainsi défini: u_i et v_i sont les plus petites séquences de T_2^* à une permutation près vérifiant:

- si pour $p \in INI$, $W_1(p, t_i) = a$, alors $(p^\bullet)^a \subseteq u_i$.
- si pour $p \in FIN$, $\forall j \leq i, \sum_{l=j}^{i} C_1(p, t_l) > 0$, on pose $b = \sum_{l=k+1}^{i} C_1(p, t_l)$, où k est le plus grand entier $< i$ vérifiant $\forall j \leq k$, $\sum_{l=1}^{k} C_1(p, t_l) > 0$. On a alors $({}^\bullet p)^b \subseteq v_i$.

On pose alors $\sigma_2 = w_1 \dots w_n$.
On a $\sigma_1 \leq_{ITF} \sigma_2$ car

- $W_2(p, w_i) = W_1(p, t_i)$, pour $p \in INI$, et $W_2(p, w_i) \leq W_1(p, t_i)$ pour $p \in FIN$.
- $C_2(p, w_i) = C_1(p, t_i)$ pour $p \in INI$; et pour $p \in FIN$, $C_2(p, w_1 \dots w_k) \geq C_1(p, t_1 \dots t_k)$, car on montre par récurrence que $C_2(p, w_1 \dots w_k) = \sup_{l \in 1..k} C_1(p, t_1 \dots t_l)$.

Il reste à montrer que σ_2 est stable et franchissable dans $RN_2 \setminus ITF$. σ_2 est franchissable car $\sum_{p \in INI_i} W_2(p, w_1 \dots w_k) \geq \sum_{p \in ITF_i} C_2(p, w_1 \dots w_k)$ à cause de ce qui précède et du lemme IV-2.

Enfin σ_2 est stable car, pour $p \in ITF$, $C_2(p, w_1 \dots w_n) = \sup_{l \in 1..n} C_1(p, t_1 \dots t_l) = C_1(p, t_1 \dots t_n)$ (lemme IV-2). Donc $C_2(p, \sigma_2) = C_1(p, \sigma_1)$ pour $p \in ITF$, et enfin $C_2(p_i, \sigma_2) = 0$ (p_i est la place intermédiaire du flot: voir figure IV-4).

Maintenant on considère une séquence stable σ_2 de RN_2, et on veut construire σ_1. L'idée est d'associer à chaque transition de σ_2 appartenant à $INI^\bullet$, l'exécution d'un service de RN_2.

Soit $\sigma_2 = t_1 \dots t_m$. A chaque consommation d'une marque d'une place de INI, on peut associer une production d'une marque dans une place de FIN. On note $A = \{j \in 1..n; t_j \in INI^\bullet\}$ et $B = \{j \in 1..n; t_j \in {}^\bullet FIN\}$. Puisque σ_2 est stable (c-à-d, $C(p_k, \sigma_2) = 0$) et d'après le lemme IV-2, il existe une bijection croissante $f : A \to B$ telle que $f(i) > i$ et si $f(i) = j$, alors il existe k, tel que $t_i \in INI_k^\bullet$ et $t_j \in {}^\bullet FIN_k$.

On construit σ_1 par récurrence. A $t_1 \dots t_k$ on associe une séquence $s_k \in T_1^*$, s_k stable franchissable et $t_1 \dots t_k \leq_{ITF} s_k$, et pour $p \in INI, W_2(p, t_1 \dots t_k) = W_1(p, s_k)$.

Pour $k = 0$, $s_0 = \lambda$.

Pas d'induction. Si $k+1 \in B$ alors $s_{k+1} = s_k$. Si $k+1 \in A$, on considère $w_{k+1} \in T_1^*$ qui amène une marque de ${}^\bullet t_{k+1}$ à $(t_{f(k+1)})^\bullet$: $s_{k+1} = s_k w_{k+1}$. Finalement, on prend $\sigma_1 = s_n$.

2) $RN_1 \equiv_{SST} RN_2 \Rightarrow SV_1 = SV_2$. A RN_i, on associe RN_{ci} qui est son équivalent canonique. Par transitivité de l'équivalence, $RN_{c1} \equiv_{SST} RN_{c2}$; or ceci implique trivialement $SV_1 = SV_2$. □

IV-3 Composition des réseaux réentrants

La preuve de la réentrance est assez difficile car elle implique la vérification d'espace d'accueil. Il est donc souhaitable de disposer d'opérateurs de composition de réseaux réentrants pour construire hiérarchiquement de grands réseaux réentrants à partir de petits réseaux. Dans cette section nous définissons d'abord deux opérations qui consistent à composer des réseaux réentrants par partage de places d'interface. Une troisième opération consiste à composer des réseaux réentrants en anneau: le réseau obtenu n'est pas réentrant mais est sans blocage. Finalement, des réseaux réentrants à ressources ordonnées sont composables par fusion des places de ressources (et non des places d'interface).

IV-3.1 Fusion de places d'interface

Les deux opérations sont la somme et le séquencement.

La somme compose deux réseaux réentrants qui rendent les mêmes services ($SV_1 = SV_2$) mais avec des traitements différents: les places initiales (resp. finales) qui se correspondent sont fusionnées ensemble.

Définition IV-5 (Somme) *Soit $RN_i = (N_i; SV_i; M_{0i})$ deux réseaux réentrants tels que $(P_1 \setminus ITF_1) \cap (P_2 \setminus ITF_2) = \emptyset$.*

La somme de RN_1 et RN_2 est définie par

$$RN_1 + RN_2 = (N; SV; M_0)$$
$$\Updownarrow$$
$$(N = N_1 \otimes N_2) \wedge (SV = SV_1 = SV_2) \wedge (M_0 = M_{01} \cup M_{02})$$

La composition séquentielle fusionne les places initiales d'un réseau avec les places finales d'un autre: les résultats du premier sont des requêtes pour le deuxième.

Définition IV-6 (Séquencement) *Soit $RN_i = (N_i; SV_i; M_{0i})$ deux réseaux réentrants tels que $(P_1 \setminus ITF_1) \cap (P_2 \setminus ITF_2) = \emptyset$.*

Le séquencement de RN_1 et RN_2 (RN_1 suivi par RN_2) est défini par

$$RN_1 \circ RN_2 = (N; SV; M_0)$$
$$\Updownarrow$$
$$(N = N_1 \otimes N_2) \wedge (SV = SV_1 \circ SV_2) \wedge (M_0 = M_{01} \cup M_{02})$$

La figure IV-5 donne le schéma de ces deux opérations. Rappelons qu'un cercle représente un ensemble de places et un rectangle un ensemble de transitions; un arc entre deux ensembles de sommets signifie qu'il peut exister un arc entre deux éléments de ces ensembles.

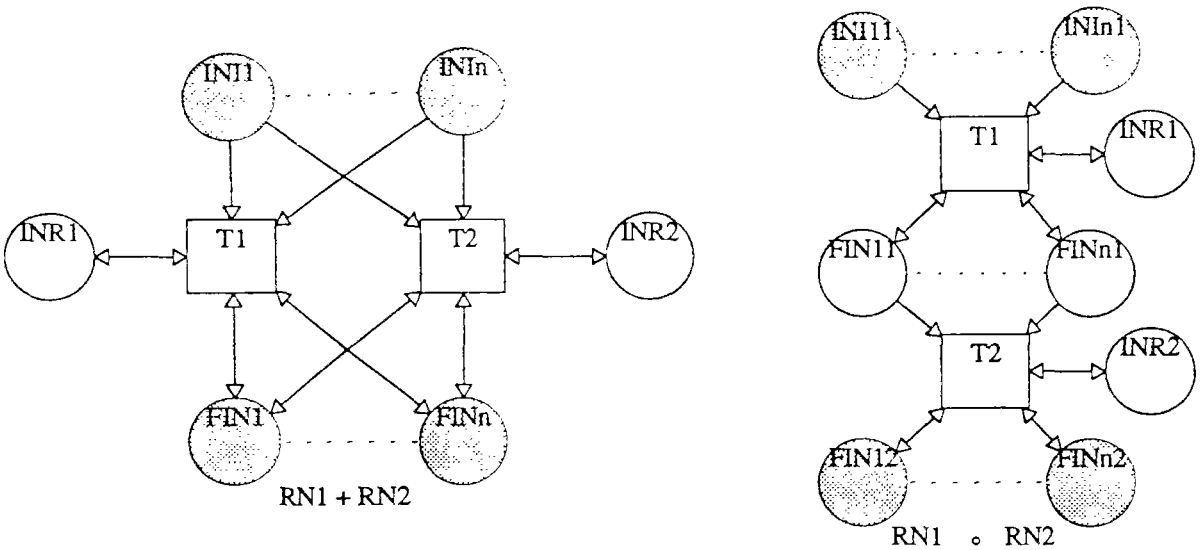


Figure IV-5: Somme et séquencement

Proposition IV-4 *Soit $RN_i = (N_i; SV_i; M_{0i})$ deux réseaux réentrants tels que $(P_1 \setminus ITF_1) \cap (P_2 \setminus ITF_2) = \emptyset$. Alors $RN_1 + RN_2$ et $RN_1 \circ RN_2$ sont des réseaux réentrants.*

Preuve Les seuls points à vérifier sont (iii) et (iv) de la définition des réseaux réentrants (les autres étant immédiats).

Le point (iii) découle de la proposition III-2, pour la somme et le séquencement.

Pour $RN_1 + RN_2$, (iv) découle du corollaire II-2.

Pour $RN_1 \circ RN_2$, la proposition III-3 implique que $STB_1{\uparrow}^P \cap STB_2{\uparrow}^P$ est un $(T \setminus (ITF_1 \cup ITF_2)^\bullet)$-ea $(P = P_1 \cup P_2)$. Il faut donc montrer que

$$STB = STB_1{\uparrow}^P \cap STB_2 extP \cap \{M; \forall p \in FIN_1, M(p) = 0\}$$

est un $(T \setminus ITF^\bullet)$-ea où $ITF = INI_1 \cup FIN_2$. Comme $FIN_1 = INI_2$, on applique (vi) de la définition des réseaux réentrants à RN_2 pour vider INI_2 et atteindre un marquage de STB. □

IV-3.2 Anneau de réseaux réentrants

L'opération $RING$ définie ci-dessous "ferme" un réseau réentrant RN en fusionnant ses places initiales avec ses places finales. Quand RN est lui-même une composition séquentielle de plusieurs réseaux réentrants, on obtient un anneau de réseaux réentrants.

Définition IV-7 (Anneau) *Soit $RN = (N; SV; M_0)$ un réseau réentrant tel que $\forall (PI, PF) \in SV, |PI| = |PF|$, et soit $f : INI \to FIN$ une bijection telle que $f(PI) = PF$ si $(PI, PF) \in SV$. Alors la fermeture (ou mise en anneau) de RN est le réseau noté $RING(RN, f) = (P', T'; W'; M'_0)$ et défini par*

- $P' = P \setminus FIN$
- $T' = T$
- $W'(x,y) = \begin{cases} W(x,y) & si\ (x,y)\ ou\ (y,x) \in (P' \setminus INI) \times T' \\ W(x,y) + W(f(x),y) & si\ x \in INI \\ W(x,f(y)) & si\ y \in INI \end{cases}$

Cette opération convient à la conception de protocoles cycliques qui sont une séquence de serveurs. La proposition montre que chaque serveur peut atteindre un état stable, et donc chaque service peut être exécuté et achevé un nombre arbitraire de fois: d'où l'absence de blocage du système.

Proposition IV-5 *Dans les hypothèses de la définition précédente, STB étant l'espace associé à RN, $STB{\uparrow}^{P'}$ est un espace d'accueil de $(N'; M'_0 + m_0)$ pour tout $m_0 \in \mathbf{N}^{INI}$.*

Preuve Soit $RN = (N; SV; M_0)$ un réseau réentrant et $RING(RN, f) = (N'; M_0)$. Il faut montrer que si $M_0 + m_0 \xrightarrow{\sigma}_{N'} M$ où $m_0 \in \mathbf{N}^{INI}$, alors il existe $u \in T'^*$ telle que $M \xrightarrow{u}_{N'} M' \in STB{\uparrow}^{P'}$.

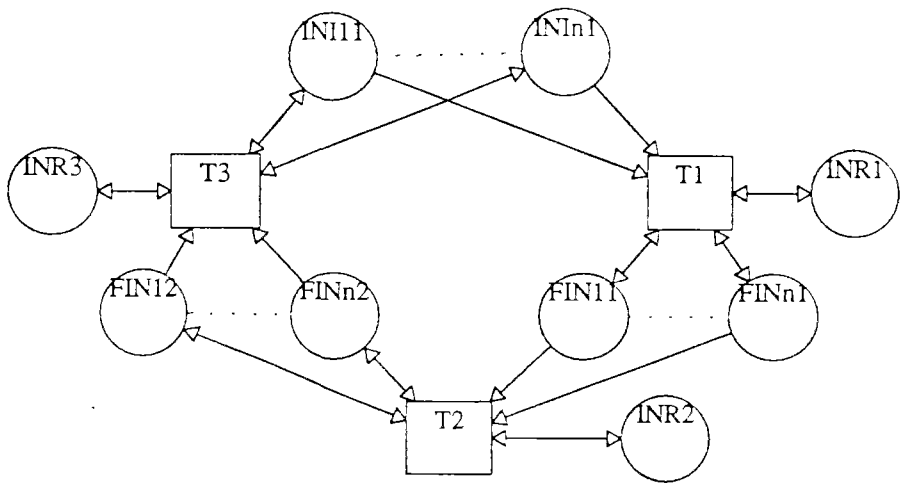


Figure IV-6: Anneau de réseaux réentrants

Dans N, on considère $q_0 \in \mathbf{N}^{ITF}$ tel que

$$q_0(p) = \begin{cases} m_0(p) + \sum_{t \in T} \bar{\sigma}(t) W'(t,p) & \text{pour } p \in INI \\ m_0(f^{-1}(p)) + \sum_{t \in T} \bar{\sigma}(t) + W'(t, f^{-1}(p)) & \text{pour } p \in FIN \end{cases}$$

où $\bar{\sigma}(t)$ désigne le nombre d'occurrences de t dans σ.
C'est-à-dire, on prend le réseau "non boucle"', en marquant de la même façon les places initiales et finales qui se correspondent.

On peut alors montrer par récurrence sur $|\sigma|$, que si $M_0 + m_0 \xrightarrow{\sigma}_{N'} M$, alors $M_0 + q_0 \xrightarrow{\sigma}_{N} Q$, avec

$$Q \geq M \text{ sur } INI \text{ et } Q \circ f^{-1} \geq M \circ f^{-1} \text{ sur } FIN$$

$$Q = M \text{ sur } P' \setminus INI = P \setminus ITF$$

D'après la réentrance, il existe $u \in (T \setminus ITF^\bullet)*$ telle que $Q \xrightarrow{u} Q' \in STB\uparrow^P$: d'où, $M \xrightarrow{u} M' \in STB\uparrow^{P'}$. □

Proposition IV-6 *Soit $(RN_i)_{i=1,n}$, n réseaux réentrants tels que*

$$RING(RN_1 \circ \cdots \circ RN_n) = (N; M_0)$$

soit défini.
Alors pour tout marquage $m_0 \in \mathbf{N}^{INI_1}$, pour tout $i \in 1..n$,

$$E_i = \{M \in \mathbf{N}^P; \forall j, \sum_{p \in FIN_{j,i}} M(p) = \sum_{p \in INI_{j,1}} m_0(p)\}$$

est un espace d'accueil de $(N; M_0 + m_0)$.

Preuve On note SPR_{ij} le support du ième flot de RN_j, $INI_{ij} = INI_j \cap SPR_{ij}$ et $FIN_{ij} = FIN_j \cap SPR_{ij}$.
Par application de la proposition précédente, $STB\uparrow^{P'}$ est un espace d'accueil, c-à-d, on peut atteindre M tel que $M(p) = 0$ pour $p \in (SPR_{ij} \setminus INI_1)$. Et donc, d'après les flots,

$$\sum_{p \in INI_{i1}} M(p) = \sum_{p \in INI_{i1}} m_0(p)$$

Alors en appliquant j fois (vi) de la définition de la rèentrance, on peut amener une marque de n'importe quelle place de $INIij$ vers n'importe quelle place de FIN_{ij}: d'où la propriété. □

IV-3.3 Partage de ressources

En raison de leurs propriétés supplémentaires différentes, il est intéressant de combiner les notions de réseaux réentrants et de réseaux à ressources ordonnées. Nous obtenons ainsi un résultat plus fort dans le cas des réseaux réentrants qui sont aussi à ressources ordonnées, tels que $STB \subseteq FREE$, et où la libération des ressources (comme la terminaison) peut se faire indépendamment de l'interface. Dans ces conditions, la composition des réseaux réentrants en tant qu'éléments de $\mathcal{RRO}$ donne un réseau réentrant.

Définition IV-8 ($\mathcal{RN} \cap \mathcal{RRO}$) *Un réseau réentrant à ressources ordonnées est un quadruplet* $(N; M_0; RES; (RES_i)_{i=1,n}; SV)$ *où*

- $(N; SV; M_0) \in \mathcal{RN}$
- *Pour tout* $m_0 \in \mathbf{N}^{ITF}$, $(N; M_0 + m_0; RES; (RES_i)_{i=1,n}) \in \mathcal{RRO}$ *tel que*
  - $\forall r \in RES, M_0(r) > 0$
  - $RES \cap ITF = \emptyset$
  - $\forall i, \forall r \in RES_i, FREE(r)$ *est un* $T(r)$*-espace d'accueil, où* $T(r) = T \setminus (\bigcup_{j \leq i} RES_j^{\bullet} \cup ITF^{\bullet})$
  - $STB \subseteq FREE$

L'ensemble de ces réseaux est noté $\mathcal{ORN}$.

Proposition IV-7 (Partage de ressources) *Soit deux réseaux réentrants à ressources ordonnées* $(N_j; M_{0j}; RES; (RES_i)_{i=1,n}; SV_j) \in \mathcal{ORN}$ *pour* $j = 1, 2$, *alors*

$$((N_1; M_{01}; RES) \otimes (N_2; M_{02}; RES); (RES_i)_{i=1,n}; SV_1 \cup SV_2) \in \mathcal{ORN}$$

Preuve On déduit de la proposition III-3 la propriétéde $T(r)$-ea de $FREE(r)$.

Il faut vérifier la réentrance du réseau composé.

Les points (i) et (ii) de la définition des réseaux réentrants sont trivialement vérifiés.

Quant au (iii), les mêmes flots sont conservés puisque, tout r étant marqué initialement et $ITF \cap RES = \emptyset$, aucune place $r \in RES$ n'appartient au support d'un flot. (v) est vrai par définition de la composition.

Il reste à vérifier (iv) et (vi).

On considère $m_0 \in \mathbf{N}^{ITF}$ où $ITF = ITF_1 \cup ITF_2$ et $(N; M_0) = (N_1; M_{01}) \otimes (N_2; M_{02})$. L'idée pour vérifier (iv) et (vi) est que si $M_0 + m_0 \xrightarrow{\sigma}_N M \in FREE$ toute séquence franchissable dans RN_i à partir de $M{\downarrow_{P_i}}$ l'est dans RN à partir de M.

Pour vérifier (iv), c-à-d que

$$STB = \{M \in \mathbf{N}^P; M{\downarrow_{P_i \setminus ITF_i}} \in STB_i\}$$

est un $(T \setminus ITF^\bullet)$-ea, on remarque qu'une fois atteint un marquage $M \in \{M; M{\downarrow_{P_1}} \in FREE_1\}$ (qui est un $(T \setminus {ITF_1}^\bullet)$-ea de $(N; M_0 + m_0)$), toutes les ressources sont disponibles. Alors, d'après la réentrance de RN_1, il existe une séquence $\sigma_1 \in (T_1 \setminus ITF^\bullet)$ telle que $M_1 \xrightarrow{\sigma_1} M_1' \in STB_1$ où $M_1 = m{\downarrow_{P_1}}$; cette séquence est franchissable dans $(N; M)$. On répète la même opération avec $M{\downarrow_{P_2}}$. □

Une interprétation de cette composition est le multiplexage d'un serveur et de ses ressources entre plusieurs clients. Le réseau réentrant de la figure IV-7 est un serveur ayant un gestionnaire de requêtes, l'unité de traitement et un tampon. Le multiplexage de ce serveur entre deux clients revient à composer deux réseaux réentrants identiques par partage du gestionnaire, de l'unité de traitement et du tampon: le réseau obtenu est encore réentrant. En plus, la propriété sur les ressources est conservée, et donc on peut composer avec un troisième réseau, etc.

IV-4 Réseaux réentrants colorés

Un réseau coloré est réentrant si son dépliage donne un réseau réentrant. Il est souhaitable de poser quelques restrictions sur la structure du réseau coloré pour que les conditions structurelles soient vérifiables sans dépliage.

Un réseau réentrant coloré est un triplet $(N; SV; M_0)$ où N est un réseau coloré et $SV \in \mathcal{B}(P) \times \mathcal{B}(P)$. On suppose en plus que si $(PI, PF) \in SV$, toutes les places de $PI \cup PF$ ont même domaine.

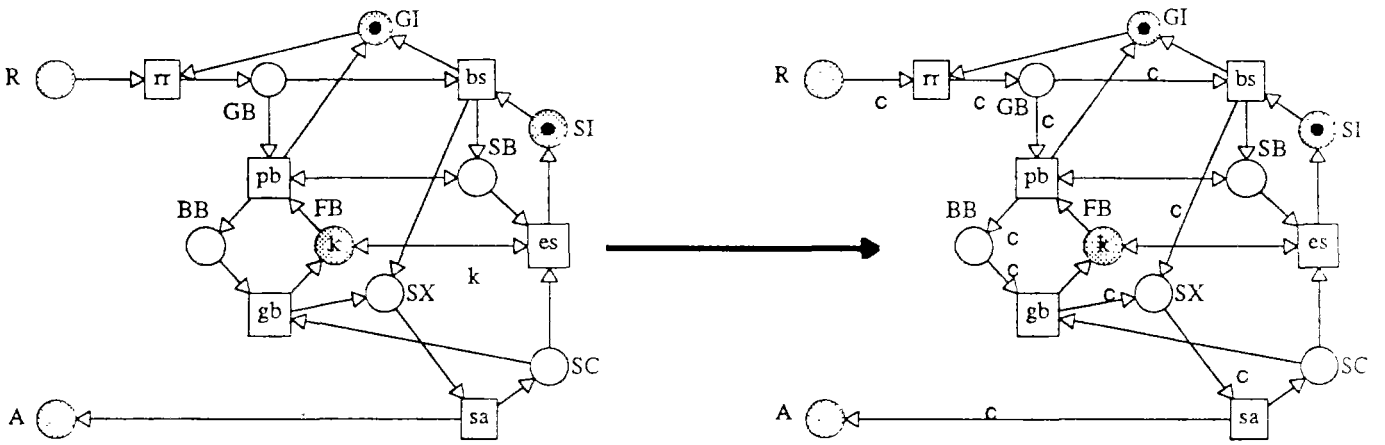


Figure IV-7: Composition de réseaux réentrants par partage de ressources

Dans un réseau coloré, un chemin d'une place vers elle-même peut se traduire dans le réseau déplié par un chemin entre deux places distinctes. Le plus simple est d'interdire les arcs sortant des places de FIN.

Il reste à fixer la forme des flots. Pour chaque flot $f_i : P \rightarrow \mathbf{N}^{D(p)}$ de support SPR_i, on suppose qu'il existe D_i, tel que

$$\forall p \in SPR_i, D(p) = D_i \times D'(p)$$

et

$$\forall t \in T, \forall c_i \in D_i, \sum_{p \in SPR_i, c' \in D'(p)} f_i(p)(c_i, c')C(p,t)(c_i, c') = 0$$

De plus, pour tout $c_i \in D_i$ et $p \in ITF_i$, $f_i(p)(c_i, .)$ est constante sur $D'(p)$.

Dans ces conditions, la figure IV-8 montre l'équivalent canonique d'un réseau réentrant coloré.

L'interprétation d'un réseau réentrant comme un serveur est à l'origine de ces contraintes. On suppose qu'une marque qui arrive dans une place initiale contient l'identité de l'emetteur plus d'autres informations. C'est pourquoi le domaine de couleur est un produit $D_i \times D'(p)$ où D_i est le domaine des identités des émetteurs. La deuxième contrainte dit qu'il y a un flot (déplié) par émetteur: on conserve uniquement le champ correspondant à l'identité de l'émetteur. La troisième condition exprime que les coefficients du flot relatifs aux places d'interface d'un flot déplié sont égaux.

Pour la composition, on suppose que les places partagées ont même domaine de couleur.

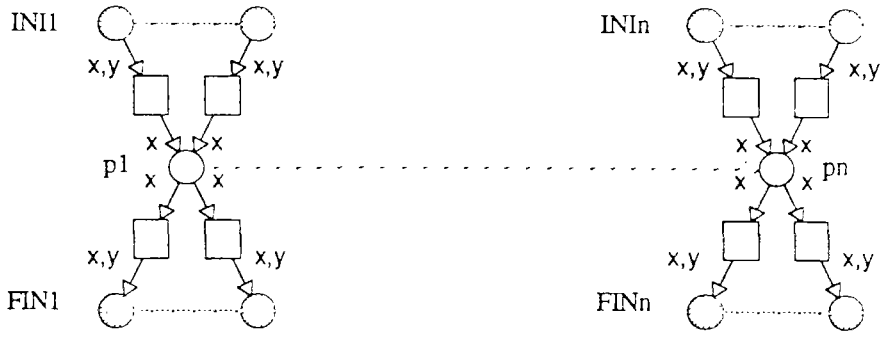


Figure IV-8: Equivalent canonique coloré

Exemple

Nous allons donner un exemple de réseau réentrant coloré et prouver sa réentrance au moyen d'une norme (en utilisant une certaine extension des normes aux réseaux colorés).

Le réseau de la figure IV-9 est un serveur muni d'un tampon pour les requêtes en attente. Quand une requête est reçue, elle est immédiatement servie si le serveur est oisif; si le serveur est occupé et le tampon non saturé la requête est mise dans le tampon.

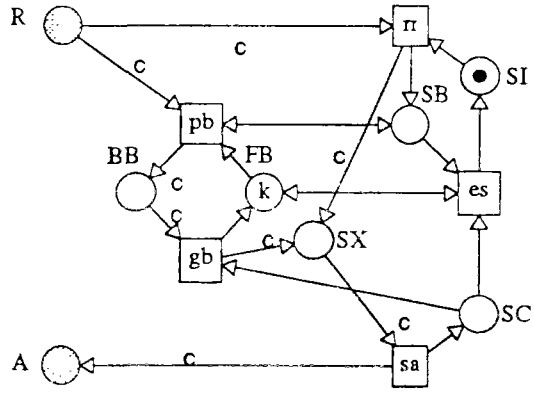


Figure IV-9: Un réseau réentrant modélisant un serveur

Dans ce réseau, $INI = \{R\}$ et $FIN = \{A\}$. Le flot correspondant à (D, A) est

$$\forall p \in \{R, BB, SX, A\}, \forall c \in D(p), f(p)(c) = 1$$

L'ensemble des états stables est

$$STB = \{M; M(BB) = M(SX) = 0\}$$

c-à-d, l'ensemble des états où il n'y a ni requêtes en attente ni requêtes en exécution. Il faut vérifier que quel que soit le marquage initial de R, on peut atteindre STB sans franchir des transitions dans $\{mt, ds\}$. Cette propriété sera prouvée à l'aide d'une norme.

Les invariants du réseau sont les suivants

$$M(SL) + \sum_{c \in D(SX)} M(SX)(c) + M(SQ) = 1 \qquad \text{(IV.1)}$$

$$\sum_{c \in D(BB)} M(BB)(c) + M(FB) = k \qquad \text{(IV.2)}$$

$$M(BB)(c)\, M(SL) = 0 \qquad \text{(IV.3)}$$

On note

$$E_0 = \{M; M \text{ vérifie } IV.1 \wedge IV.2 \wedge IV.3\}$$

On définit la fonction $\nu_1 : E_0 \rightarrow \mathbf{N}$ par

$$\nu_1(M) = \sum_{c \in D(SX)} M(SX)(c)$$

Montrons que ν_1 est une (E_0, T', E_1)-norme où $T' = T \setminus \{ds, mt\}$ et

$$E_1 = \{M; (M \in E_0) \wedge (M(SX) = 0)\}$$

Si $\nu_1(M) > 0$, on franchit la transition ep pour une couleur appropriée, et on atteint $M \in E_0$ tel que $\nu_1(M') < \nu_1(M)$.

Maintenant, on considère la fonction $\nu_2 : E_1 \rightarrow \mathbf{N}$ définie par

$$\nu_2(M) = \sum_{c \in D(BB)} M(BB)(c)$$

Montrons que ν_2 est une (E_1, T', E_2)-norme où

$$E_2 = \{M; (M \in E_1) \wedge (M(BB) = 0)\}$$

Si $\nu_2(M) > 0$ alors $M(SL) = 0$ d'après l'invariant IV.3. Mais IV.1 implique alors $M(SQ) = 1$ (ne pas oublier que $M \in E_1$.) On franchit alors $(rt, c)(ep, c')$ et on atteint un marquage $M' \in E_1$ tel que $\nu_2(M') < \nu_2(M)$.

Finalement, par application de la composition des normes, $\nu = (\nu_1, \nu_2)$ est une (E_0, T', E_2)-norme: comme E_0 contient l'ensemble des marquages accessibles de $(N; M_0 + m_0)$ quel que soit $m_0 \in \mathbf{N}^{ITF}$ et $E_2 = STB$, on en déduit que STB est un $(T \setminus ITF^\bullet)$-espace d'accueil du réseau quel que soit le marquage initial de l'interface.

Conclusion

En visant à définir une classe de réseaux qui permet d'appliquer les résultats des chapitres précédents sur le remplacement de sous-réseaux et la composition par fusion de places, nous avons défini les réseaux réentrants qui sont une sous-classe des OI-réseaux robustes généralisant les réseaux à terminaison propre.

Ainsi, l'SST-équivalence de deux réseaux réentrants est simplement vérifiable par comparaison des interfaces des réseaux. De plus, plusieurs types de fusions de places permettent de construire hiérarchiquement des réseaux réentrants.

Cette classe semble avoir de bonnes propriétés pratiques de modélisation, car un réseau réentrant s'interprète facilement comme un serveur. Mais certaines contraintes, telles l'absence de synchronisation des services, sont assez restrictive. Il est souhaitable de trouver un compromis avec une classe moins restreinte et plus pratique, où la vérification de l'équivalence serait moins triviale mais resterait plus simple que dans le cas général.

Conclusion

L'objectif de cette thèse était de contribuer au développement de méthodes modulaires pour la conception et l'analyse de réseaux de Petri complexes. Nous voulions définir des méthodes hiérarchiques de preuves de propriétés telles que les espaces d'accueil ou l'absence de blocage, associées à des méthodes de conception hiérarchique.

Notre résultat le plus important est la mise en place d'une théorie de raffinement par remplacement de sous-réseaux ouverts, fondée sur une nouvelle équiva lence bisimulationnelle (SST-éq), et sur des notions d'interface et d'états stables. La vérification de cette équivalence est assez complexe, mais la classe des réseaux réentrants, pour laquelle la vérification de l'équivalence est facile, rend cette théorie plus applicable.

Notre deuxième résultat est la définition d'un nouveau concept de norme pour la preuve des espaces d'accueil. Les opérations de composition que nous avons définies sur ces normes, permettent une vérification modulaire d'espace d'accueil. Nous avons, de plus, introduit une représentation d'une preuve de norme permettant, après un raffinement de réseau, de dériver la preuve de norme du nouveau réseau par réutilisation de celle du réseau de départ.

Pour étudier la composition par fusion de places, nous avons d'abord développé une synthèse des propriétés générales de cette opération. Cela nous a permis de dégager deux résultats nouveaux: la conservation d'espace d'accueil dans certaines conditions et des propriétés de composition des réseaux à ressources ordonnées.

Ce travail ouvre deux perspectives: d'une part, l'approfondissement de ces notions théoriques, et d'autre part, leur mise en œuvre logicielle.

L'SST-équivalence est une première tentative de définition d'un cadre théorique général pour le remplacement de sous-réseaux ouverts. Mais le fait qu'elle ne soit pas définie par rapport à un environnement a quelques effets indésirables:

- D'un côté, cette équivalence est trop fine car elle sépare des réseaux qui ont un comportement "équivalent" dans certains environnements et pas dans

d'autres.

- D'un autre côté, certaines identifications de réseaux entraînent la non fermeture de l'ensemble des systèmes à interface ouverte par le remplacement de réseaux SST-équivalents, ce qui nous a amené à définir les réseaux robustes.

Cette notion d'équivalence doit être modifiée pour prendre en compte ces deux problèmes.

L'extension aux réseaux colorés est immédiate, si on se borne à dire que deux réseaux colorés sont SST-équivalents si leurs réseaux dépliés le sont. Il serait donc plus intéressant de définir l'équivalence avec des marquages et des séquences symboliques (au sens de Haddad [30,16]), et de vérifier qu'elle correspond à celle définie avec les marquages et les séquences "ordinaires."

Un autre point à étudier est la méthode de preuve de cette équivalence. Sa vérification est complexe et la conception d'un algorithme manipulant les graphes de marquages semble difficile, d'autant plus qu'il s'agit de comparer des réseaux privés de leurs places d'interface qui sont donc en général non bornés. Deux solutions sont possibles: la définition de transformations conservant cette équivalence, et la définition de classes particulières de réseaux pour lesquelles la vérification de l'équivalence est plus simple.

C'est là que les réseaux réentrants trouvent leur utilité: la vérification de l'équivalence de deux réseaux réentrants est facile; mais il reste à vérifier la réentrance. Les seules propriétés difficiles à prouver sont celle d'espace d'accueil et d'absence de synchronisation des services:

- La propriété d'espace d'accueil est vérifiable par l'algorithme de Johnen[35], même pour les réseaux non bornés. Cette partie de la thèse débouche donc sur une mise en œuvre logicielle. Néanmoins, pour la modélisation et l'analyse des systèmes distribués, une telle mise en œuvre est surtout intéressante pour les réseaux colorés paramétrés: il est donc souhaitable de développer maintenant des algorithmes de vérification (semi-)automatique de la propriété d'espace d'accueil pour ces réseaux.

- La propriété d'indépendance des services d'un réseau réentrant est assez contraignante dans certaines applications. Il faudrait définir une classe moins restrictive pour laquelle l'équivalence reste simple à vérifier et la mise en œuvre logicielle envisageable.

Nous avons défini des normes munies d'opérations de composition de notion de preuves et illustré une méthode de réutilisation de preuve dans une conception hiérarchique sur des exemples simples.

Il reste d'abord à étudier dans le cas général le lien entre ces opérateurs et la composition de réseaux. Par exemple, sous quelles conditions la composition de deux réseaux admet comme norme la composition des normes associées à chaque réseau, et pour quelle composition: fusion de places ou de transitions.

Ensuite notre méthode de réutilisation de preuve demande à être formalisée pour être automatisée: définition d'une conception hiérarchique et des liens entre les différents modèles la composant, ainsi que la définition de la transformation et de la réutilisation de preuve.

Enfin et à plus long terme, dans la perspective d'une mise en œuvre logicielle, comme une méthode de vérification hiérarchique d'espace d'accueil assitée par un système expert, il faudra introduire des représentations de preuves en machine, des transformations de preuves et des règles de vérification détaillée de la validité des pas de preuve. On peut envisager aussi une extension aux réseaux colorés.

Bibliographie

[1] C. André. *Systèmes à évolutions parallèles: Modélisation par réseaux de Petri à capacités et analyse par abstraction.* Thèse d'état de l'Université de Nice, 1981.

[2] C. André. *Behaviour of a Place/Transition Net on a Subset of Transitions.* Applications and Theory of Petri Nets, IF 52, Springer Verlag, 1982, pp 131-135

[3] C. André. *Use of the Behavior Equivalence in Place-Transition Net Analysis.* Applications and Theory of Petri Nets, IF 52, Springer Verlag, 1982.

[4] C. André. *Structural Transformations Giving B-Equivalent P/T-Nets.* Applications and Theory of Petri Nets, IF 66, Springer Verlag, 1983, pp 241-250.

[5] E. Battiston, F, De Cindio, G. Mauri, L. Rapanotti. *Morphisms and Minimal Models for OBJSA Nets.* Twelfth International Conference on Application and Theory of Petri Nets, Gjern (Danemark), Juin 1991.

[6] B. Baumgarten. *On Internal and External Characterizations of PT-nets Building Block Behavior.* Advances in Petri Nets 88, LNCS 340, pp 44-61.

[7] G. Berthelot. *Transformation et analyse de réseaux de Petri, applications aux protocoles.* Thèse d'état de l'Université Paris 6, 1983.

[8] G. Berthelot. *Transformations and Decompositions of Nets.* Petri Nets: Central Models and their Properties, LNCS 254, Springer Verlag, 1986, pp359-376.

[9] A. Bourguet-Rouger. *External Behavior Equivalence between two Petri Nets.* Concurrency 88, LNCS 335, pp 237-256

[10] A. Bourguet-Rouger. *Etude de la concordance de comportement de deux réseaux de Petri. Application à la validation de protocoles. Détection automatique des erreurs de conception.* Thèse de doctorat de l'Université Paris 6, 27 Septembre 1990.

[11] G.W. BRAMS. *Réseaux de Petri: théorie et pratique.* Masson, volumes 1 et 2, Paris 1982 et 1983.

[12] G. Chehaibar. *Mise en Pratique de l'Intégration de Preuves dans la conception Progressive de Protocoles.* Journées FIRTECH Systèmes et Télématique "Méthodes et Outils pour la Spécification et la Validation des Protocoles," CNET, Issy les Moulineaux, Janvier 1990.

[13] G. Chehaibar. *Validation of Phase-Executed Protocols Modelled with Colored Petri Nets.* Eleventh International Conference on Application and Theory of Petri Nets, Paris, June 1990.

[14] G. Chehaibar. *Replacement of Open Interface Subnets and Stable State Transformation Equivalence.* Twelfth International Conference on Application and Theory of Petri Nets, Gjern (Danemark), Juin 1991.

[15] G. Chehaibar. *Use of Reentrant Nets in Modular Analysis of Colored Nets.* High-level Petri Nets: Theory and Application, Springer Verlag, 1991; et dans Advances in Petri Nets 91, Lecture Notes in Computer Science, Springer Verlag, à paraître en 1992.

[16] G. Chiola, C. Duteillet, G. Franceschinis, S. Haddad. *On Well-Formed Colored Nets and their Symbolic Reachability Graph.*Eleventh International Conference on Application and Theory of Petri Nets, Paris, June 1990.

[17] J.M. Colom and M. Silva. *Improving the Linearly Based Characterization of P/T Nets.* Tenth International Conference on Application and Theory of Petri Nets, Bonn, June 1989.

[18] F. De Cindio, G. De Michelis, L. Pomello, C. Simone. *Superposed Automata Nets.* Applications and Theory of Petri Nets, IF 52, Springer Verlag, 1982, pp 269-279

[19] F. De Cindio, G. De Michelis, L. Pomello, C. Simone. *A State Transformation Equivalence for Concurrent Systems: Exibited Functionality Equivalence.* Concurrency 88, LNCS 335, pp 222-236.

[20] F. De Cindio, G. De Michelis, C. Simone. *GAMERU: A Language for the Analysis and Design of Human Communication Pragmatics within Organizational Systems.* Advances in Petri Nets 87, LNCS 266, pp 21-44.

[21] E.W. Dijkstra. *Co-operating sequential processes.* Programming Languages, F. Genuys (ed), Academic Press, London 1968, pp 43-112.

[22] R. Di Giovanni. *Petri Nets and Software Engineering: HOOD Nets.* Eleventh International Conference on Application and Theory of Petri Nets, Paris, June 1990.

[23] C. Girault and G. Chehaibar. *Proof Reusability in Stepwise Refinement with Petri Net Modelling: Application to a Client-Server Model.* Fourth International Symposium on Computer and Information Sciences, Turquie, Novembre 1989.

[24] C. Girault. *Modélisation Hiérarchique d'un Système Distribué.* Journées FIRTECH Systèmes et Télématique "Méthodes et Outils pour la Spécification et la Validation des Protocoles," CNET, Issy les Moulineaux, Janvier 1990.

[25] C. Girault. *Petri Net Methods for Design and Analysis of Distributed Systems.* Invited Talk, Eleventh International Conference on Application and Theory of Petri Nets, Paris, June 1990.

[26] H.J. Genrich, E. Stankiewicz-Wiechno. *A Dictionary of Some Basic Notions of Net Theory.* Net Theory and Applications, Proceedings of Advanced Course on General Net Theory of Processes and Systems, Hamburg, October 1979, Springer Verlag, 1980.

[27] R. van Glabbeek, U. Goltz. *Equivalence Notions for Concurrent Systems and Refinement of Actions.* MFCS 89, LNCS 379, pp 237-248.

[28] A.N. Habermann. *Prevention of System Deadlock.* Communication of the ACM, Volume 12, Number 7, July 1969, pp 373-377.

[29] M. Hack. *Extended State-Machine Allocatable Nets, an Extension of Free Choice Nets.* M.I.T. Cambridge Mass. Project Mac, CSG-Memo 78-1, 1973.

[30] S. Haddad. *Une catégorie régulière de réseau de Petri de haut niveau: définition, propriétés et réductions. Applications à la validation de systèmes distribués.* Thèse de doctorat de l'Université Paris 6, Juin 1987.

[31] S. Haddad. *A Reduction Theory for Colored Nets.* Advances in Petri Nets 89, LNCS 424, Springer Verlag.

[32] D. Hauschildt and R. Valk. *Safe States in Banker like Resource Allocation Problems.* Advances in Petri Nets 85, LNCS 222, pp 253-277.

[33] R.C. Holt. *Some Deadlock Properties of Computer Systems.* Computing Surveys, volume 4, number 3, September 1972, pp 179-196.

[34] K. Jensen. *Coloured Petri Nets.* Advances in Petri Nets 86, LNCS 254, Springer Verlag 1987, pp 248-299.

[35] C. Johnen. *Analyse Algorithmique des Réseaux de Petri: Vérification d'Espace d'Accueil, Systèmes de Réécriture.* Thèse de doctorat de l'Université Paris-Sud, 4 Décembre 1987.

[36] R.M. Keller. *Formal Verification of Parallel Programs.* Communication of the ACM, Volume 19, Number 7, July 1969, pp 371-384.

[37] V.E. Kotov. *An Algebra for Parallelism Based on Petri Nets.* MFCS 78, LNCS 64, Springer Verlag, pp39-55.

[38] G. Lausen. *Modlling and Analysis of the Behaviour of Infomation Systems.* IEEE Transactions on Software Engineering, volume 14, number 11, November 1988, pp 1610-1620.

[39] K. Lautenbach and P.S. Thiagarajan. *Analysis of a Resource Allocation Problem Using Petri Nets.* First European Conference on Parallel and Distributed Procesing, Toulouse, France, February 1979.

[40] J.C. Lloret. *Réseaux prédicat/transition étiquetés pour la modélisation et la vérification de systèmes informatiques répartis.* Thèse de doctorat de l'Université Paul Sabatier, Juillet 1990.

[41] A. Mazurkiewicz. *Semantics of Concurrent Systems: a Modular Fixed-Point Trace Approach.* Advances in Petri Nets 84, LNCS 188, Springer Verlag, pp 353-371.

[42] G. Memmi. *Méthodes d'analyse des réseaux de Petri, Réseaux à files, Applications au temps réel.* Thèse d'Etat de l'Université Paris 6, 7 Juin 1983.

[43] G. Memmi and J. Vautherin. *Analysing Nets by the Invariant Method.* Petri Nets: Central Models and their Properties, LNCS 254, Springer Verlag, 1986, pp 300-337.

[44] C. Dimitrovici, U. Hummert, L. Petrucci. *The Properties of Algebraic Net Schemes in Some Semantics.* Eleventh International Conference on Application and Theory of Petri Nets, Paris, June 1990.

[45] L. Petrucci. *Techniques d'analyse des réseaux de Petri algébriques.* Thèse de doctorat de l'Université Paris 6, 18 Janvier 1991.

[46] L. Pomello. *Some Equivalence Notions for Concurrent Systems: An Overview.* Advances in Petri Nets 85, LNCS 222, pp 381-400

[47] L. Pomello, C. Simone. *A State Transformation Preorder over a Class of EN-Systems.* Tenth International Conference on Application and Theory of Petri nets, Bonn, June 1989.

[48] L. Pomello. *Refinement of Concurrent Systems Based on Local State Transformations.* Stepwise Refinement of Distributed Systems, LNCS 430, Springer Verlag 1990, pp 641-668.

[49] W. Reisig. *On a Class of Cooperating Sequential Processors.* First European Conference on Parallel and Distributed Processing, Toulouse, February 1979.

[50] W. Reisig. *Petri Nets: An Introduction.* Springer Verlag, 1985.

[51] Y. Souissi and G. Memmi. *Compositions of Nets via a Communication Medium.* Tenth International Conference on Application and Theory of Petri Nets, Bonn, June 1989.

[52] Y. Souissi. *Une Etude de la Préservation de Propriétés par Composition de Réseaux de Petri. Quelques Extensions aux Réseaux à Files. Application à la Validation de Protocoles de Communication.* Thèse de doctorat de l'Université Paris 6, 20 Février 1990.

[53] Y. Souissi. *On Liveness Preservation by Composition of Nets via a Set of Places.* Eleventh International Conference on Application and Theory of Petri Nets, Paris, June 1990.

[54] Y. Souissi. *Deterministic Systems of Sequential Processes: A Class of Structured Petri Nets.* Twelfth International Conference on Application and Theory of Petri Nets, Gjern (Danemark), Juin 1991.

[55] I. Suzuki, T. Murata. *A Method for Stepwise Refinement and Abstraction of Petri Nets.* JCSS 27, 1983, pp 51-76.

[56] M. Tazza. *Quantative Analysis of a Resource Allocation Problem: a Net Theory Based Proposal.* Concurrency and Nets, Springer Verlag 1987, pp 511-532

[57] R. Valette. *Analysis of Petri Nets by Stepwise Refinements.* JCSS 18, 1979, pp 35-46.

[58] R. Valk and M. Jantzen. *The residue of vector sets with application to decidability problems in Petri Nets.* Acta Informatica 21, pp 643-674, 1985.

[59] A. Valmari. *Compositional State Space Generation.* Eleventh International Conference on Application and Theory of Petri Nets, Paris, June 1990.

[60] W. Vogler. *Behavior Preserving Refinements of Petri Nets.* Graph-Theoretic Concepts in Computer Science 86, LNCS 246, pp 82-93.

[61] W. Vogler. *Failures Semantics and Deadlocking of Modular Petri Nets.* Acta Informatica 26, pp 333-348, 1989.

[62] W. Vogler. *Failures Semantics Based on Interval Semiwords is a Congruence for Refinement.* Distributed Computing 4, pp 139-162, 1991.

[63] W. Vogler. *Failures Semantics of Petri Nets and the Refinement of Places and Transitions.* TUM 350, Janvier 1990.

[64] W. Vogler. *Asynchronous Communication of Petri Nets and the Refinement of Transitions.* TUM 342/7/91 A, 1991.

[65] K. Voss. *Interface as a Basic Concept for System Specification and Verification.* Concurrency and Nets, Springer Verlag 1987, pp 585-604

[66] G. Winskel. *Petri Nets, Morphisms and Compositionality.* Advances in Petri Nets 85, LNCS 222, pp 453-477.