

Réseaux d'Automates Stochastiques : Génération de l'espace d'états atteignables et Multiplication vecteur-descripteur pour une sémantique en temps discret

Afonso Henrique CORRÊA DE SALES

Directeur de thèse : Mme Brigitte PLATEAU

Laboratoire d'Informatique de Grenoble
Équipe Projet INRIA MESCAL
CAPES - Brésil

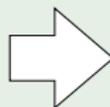


Évaluation de performance de systèmes et réseaux informatiques

D'autres domaines : système de production, physique, biologie, chimie, *etc.*

Démarche

Modélisation



Calcul de
prédiction de
performance

Intérêt

Méthodes qui utilisent des **modèles** basés sur la **théorie de chaînes de Markov**.

Chaînes de Markov

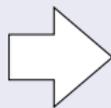
- La représentation du comportement du système en terme *états/transitions*
- Échelle de temps
 - Temps continu : distribution exponentielle
 - Temps discret : distribution géométrique
- Étude nécessite deux étapes importantes
 - Calcul des composants irréductibles
 - Calcul en régime transitoire/stationnaire des probabilités d'états

Chaînes de Markov

- **Avantage** : modélisation simple et facile pour des “petits” systèmes
- **Limitation** : modélisation directe sous la forme états/transitions de *systèmes à grand espace d'états* devient quasiment *impossible*

Formalismes structurés

Description
du modèle par
des composants



Description
mathématique
structurée

Outils/Mathématiques

- Algèbres tensorielles
- Arbres : diagrammes de décision

Exemples de formalismes structurés

- **Réseaux de Files d'Attente** [Little61, Basket *et al.* 75, Reiser *et al.* 80]
 - approche orienté : “ressources consommés par des clients”
- **Réseaux de Petri Stochastiques (SPN)** [Florin *et al.* 85]
 - analyse fine des synchronisations (places/transitions)
- **Algèbre de Processus pour l'Évaluation de Performance (PEPA)** [Hillston95]
 - composition concurrente et exécution parallèle
- **Réseaux d'Automates Stochastiques (SAN)** [Plateau84]
 - parallélisme et synchronisme + vitesse de transition dans un composant dépendant de tout le système

Exemples de formalismes structurés

Réseaux de Files d'Attente [Little61, Basket *et al.* 75, Reiser *et al.* 80]
approche orienté : "ressources consommés par des clients"

Réseaux de Petri Stochastiques (SPN) [Florin *et al.* 85]
analyse fine des synchronisations (places/transitions)

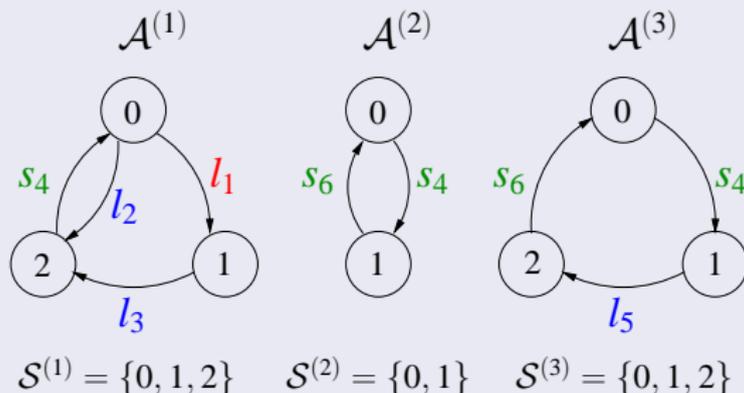
Algèbre de Processus pour l'Évaluation de Performance
(PEPA) [Hillston95]
composition concurrente et exécution parallèle

- **Réseaux d'Automates Stochastiques (SAN)** [Plateau84]
 - parallélisme et synchronisme + vitesse de transition dans un composant dépendant de tout le système

Réseaux d'Automates Stochastiques (SAN)

- Système modélisé par des (petits) sous-systèmes (automates) “presque” indépendants
- Automates sont représentés par des **états**, **événements** et **transitions** entre les états
- Événements :
 - locaux
 - synchronisants
- Vitesse de transition : taux/probabilités fonctionnels (*fonctions*)
- Échelle de temps : continue ou discrète
- Représentation modulaire de la matrice de transition de la chaîne de Markov sous-jacente (algèbres tensorielles)

Exemple : Modèle SAN à temps continu



Type	Even.	Taux	Type	Even.	Taux	Type	Even.	Taux
loc	l_1	f_1	loc	l_2	γ	loc	l_3	δ
syn	s_4	λ	loc	l_5	μ	syn	s_6	σ

$$f_1 = [(\mathcal{A}^{(2)} == 0) \&\& (\mathcal{A}^{(3)} == 0)] \times \alpha \parallel$$

$$[(\mathcal{A}^{(2)} == 1) \&\& (\mathcal{A}^{(3)} == 2)] \times \beta$$

Représentation tensorielle

Descripteur Markovien Q :

$$Q = \sum_{j=1}^N \bigotimes_{i=1}^N \mathcal{Q}_j^{(i)} + \sum_{e \in \mathcal{E}_s} \left(\bigotimes_{i=1}^N \mathcal{Q}_{e^+}^{(i)} + \bigotimes_{i=1}^N \mathcal{Q}_{e^-}^{(i)} \right), \quad \text{où } \mathcal{Q}_j^{(i)} = \begin{cases} \mathcal{Q}_l^{(i)} & \text{si } j = i \\ I_{|\mathcal{S}^{(i)}|} & \text{si } j \neq i \end{cases}$$

- $\mathcal{Q}_l^{(i)}$ matrices de transitions des événements locaux
- $\mathcal{Q}_{e^+}^{(i)}$ matrices de transitions de l'événement synchronisant e

Problème des modèles compositionnels

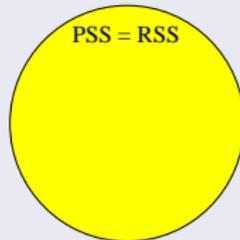
Modèles
compositionnels



Explosion
combinatoire
de l'espace d'états

Espace d'états du modèle

PSS = RSS



Espace d'états produit (PSS)

$$\text{PSS} = \mathcal{S}^{(1)} \times \dots \times \mathcal{S}^{(N)}$$

Espace d'états atteignables (RSS)

$$\text{RSS} \subseteq \text{PSS}$$

Travaux précédents

Méthode performante pour la génération du RSS de **modèles SPN** (Réseaux de Petri Stochastiques) [Ciardo *et al.* 01] basée sur l'utilisation d'arbres (MDD).

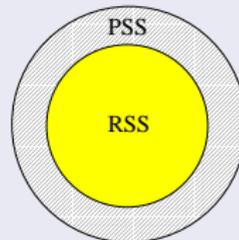
Problème des modèles compositionnels

Modèles
compositionnels



Explosion
combinatoire
de l'espace d'états

Espace d'états du modèle



Espace d'états produit (PSS)

$$\text{PSS} = \mathcal{S}^{(1)} \times \dots \times \mathcal{S}^{(N)}$$

Espace d'états atteignables (RSS)

$$\text{RSS} \subseteq \text{PSS}$$

Travaux précédents

Méthode performante pour la génération du RSS de **modèles SPN** (Réseaux de Petri Stochastiques) [Ciardo *et al.* 01] basée sur l'utilisation d'arbres (MDD).

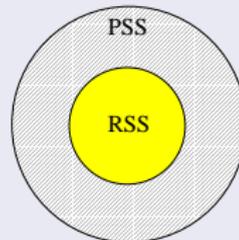
Problème des modèles compositionnels

Modèles
compositionnels



Explosion
combinatoire
de l'espace d'états

Espace d'états du modèle



Espace d'états produit (PSS)

$$PSS = \mathcal{S}^{(1)} \times \dots \times \mathcal{S}^{(N)}$$

Espace d'états atteignables (RSS)

$$RSS \subseteq PSS$$

Travaux précédents

Méthode performante pour la génération du RSS de **modèles SPN** (Réseaux de Petri Stochastiques) [Ciardo *et al.* 01] basée sur l'utilisation d'arbres (MDD).

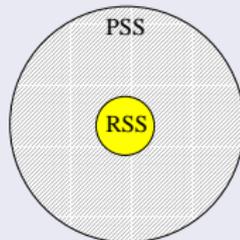
Problème des modèles compositionnels

Modèles
compositionnels



Explosion
combinatoire
de l'espace d'états

Espace d'états du modèle



Espace d'états produit (PSS)

$$PSS = \mathcal{S}^{(1)} \times \dots \times \mathcal{S}^{(N)}$$

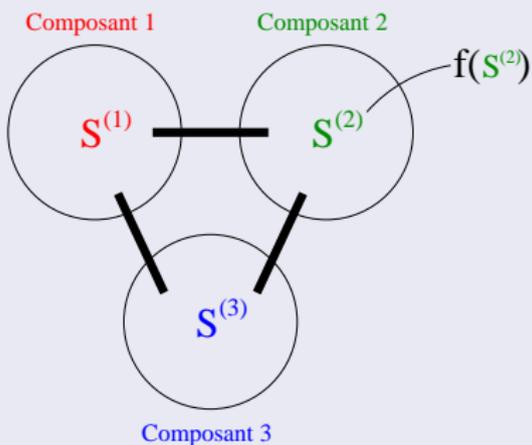
Espace d'états atteignables (RSS)

$$RSS \subseteq PSS$$

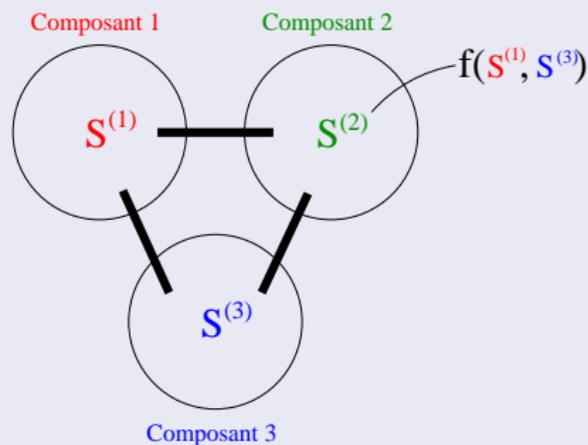
Travaux précédents

Méthode performante pour la génération du RSS de **modèles SPN** (Réseaux de Petri Stochastiques) [Ciardo *et al.* 01] basée sur l'utilisation d'arbres (MDD).

Modèle SPN



Modèle SAN



Objectif

- Trouver une *méthode efficace* pour la génération de l'espace d'états atteignables d'un modèle SAN à temps continu qui permet l'utilisation de *fonctions qui dépendent d'états globaux* du modèle.

Modèles à temps continu

Un **seul événement** peut avoir lieu à chaque instant (variables aléatoires indépendantes suivant une distribution exponentielle).

Modèles à temps discret

À chaque intervalle de temps **plusieurs événements** peuvent avoir lieu dans le même instant (variables aléatoires indépendantes suivant une distribution géométrique).

La possibilité d'avoir des **événements dans la même unité de temps** rend la modélisation de *systemes en temps discret plus complexe* que la modélisation de *systemes en temps continu*.

Étapes de calcul

- Espace d'états atteignables (RSS)
- Vecteur de probabilité

Opération la plus coûteuse

Multiplication d'un vecteur de probabilité π par la matrice de transition \mathcal{M} de la chaîne de Markov du modèle :

$$\pi \times \mathcal{M} = \pi'$$

Objectifs

- Trouver une *représentation compacte* d'un modèle SAN à temps discret
- Exploiter cette représentation pour avoir *un calcul* $\pi \times \mathcal{M}$ *efficace*

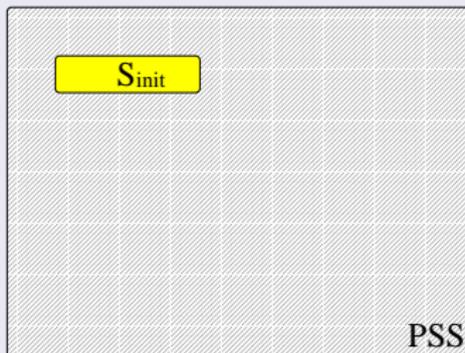
Plan

- Contexte
- **Génération du RSS**
 - Principe de la génération
 - Outils/Mathématiques
 - Descripteur d'atteignabilité
 - Études de cas
 - Synthèse
- Calcul du vecteur de probabilité
- Conclusions et Perspectives

Modèle états/transitions

- \mathcal{S}_{init} : l'ensemble d'états initiaux ($\mathcal{S}_{init} \subseteq \text{PSS}$)
- \mathcal{N} : la fonction “*next-state*” ($\text{PSS} \rightarrow 2^{\text{PSS}}$), où $\mathcal{N}(\tilde{x})$ détermine l'ensemble des états qui peuvent être **atteignables** à partir de l'état global \tilde{x}

Principe de la méthode de génération du RSS



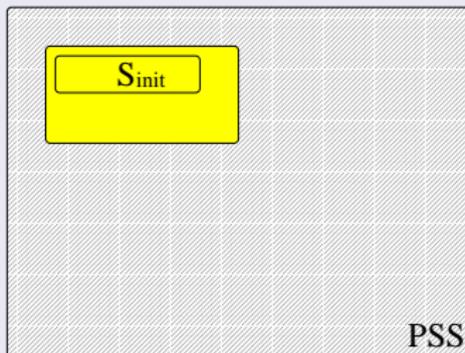
Clôture réflexive et transitive de la fonction \mathcal{N}

$$\text{RSS} = \mathcal{S}_{init} \cup \mathcal{N}(\mathcal{S}_{init}) \cup \mathcal{N}^2(\mathcal{S}_{init}) \cup \dots = \mathcal{N}^*(\mathcal{S}_{init})$$

Modèle états/transitions

- \mathcal{S}_{init} : l'ensemble d'états initiaux ($\mathcal{S}_{init} \subseteq \text{PSS}$)
- \mathcal{N} : la fonction “*next-state*” ($\text{PSS} \rightarrow 2^{\text{PSS}}$), où $\mathcal{N}(\tilde{x})$ détermine l'ensemble des états qui peuvent être **atteignables** à partir de l'état global \tilde{x}

Principe de la méthode de génération du RSS



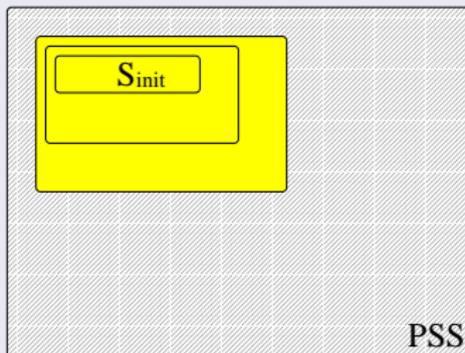
Clôture réflexive et transitive de la fonction \mathcal{N}

$$\text{RSS} = \mathcal{S}_{init} \cup \mathcal{N}(\mathcal{S}_{init}) \cup \mathcal{N}^2(\mathcal{S}_{init}) \cup \dots = \mathcal{N}^*(\mathcal{S}_{init})$$

Modèle états/transitions

- \mathcal{S}_{init} : l'ensemble d'états initiaux ($\mathcal{S}_{init} \subseteq \text{PSS}$)
- \mathcal{N} : la fonction "next-state" ($\text{PSS} \rightarrow 2^{\text{PSS}}$), où $\mathcal{N}(\tilde{x})$ détermine l'ensemble des états qui peuvent être **atteignables** à partir de l'état global \tilde{x}

Principe de la méthode de génération du RSS



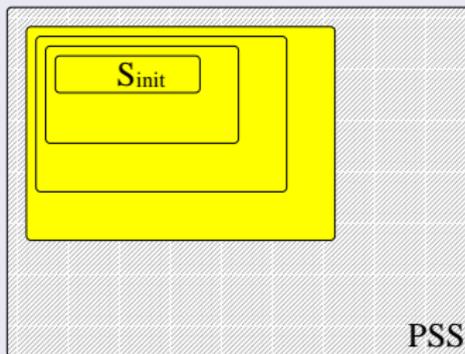
Clôture réflexive et transitive de la fonction \mathcal{N}

$$\text{RSS} = \mathcal{S}_{init} \cup \mathcal{N}(\mathcal{S}_{init}) \cup \mathcal{N}^2(\mathcal{S}_{init}) \cup \dots = \mathcal{N}^*(\mathcal{S}_{init})$$

Modèle états/transitions

- \mathcal{S}_{init} : l'ensemble d'états initiaux ($\mathcal{S}_{init} \subseteq \text{PSS}$)
- \mathcal{N} : la fonction “*next-state*” ($\text{PSS} \rightarrow 2^{\text{PSS}}$), où $\mathcal{N}(\tilde{x})$ détermine l'ensemble des états qui peuvent être **atteignables** à partir de l'état global \tilde{x}

Principe de la méthode de génération du RSS



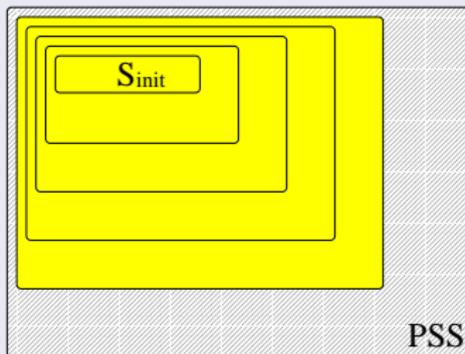
Clôture réflexive et transitive de la fonction \mathcal{N}

$$\text{RSS} = \mathcal{S}_{init} \cup \mathcal{N}(\mathcal{S}_{init}) \cup \mathcal{N}^2(\mathcal{S}_{init}) \cup \dots = \mathcal{N}^*(\mathcal{S}_{init})$$

Modèle états/transitions

- \mathcal{S}_{init} : l'ensemble d'états initiaux ($\mathcal{S}_{init} \subseteq \text{PSS}$)
- \mathcal{N} : la fonction "next-state" ($\text{PSS} \rightarrow 2^{\text{PSS}}$), où $\mathcal{N}(\tilde{x})$ détermine l'ensemble des états qui peuvent être **atteignables** à partir de l'état global \tilde{x}

Principe de la méthode de génération du RSS



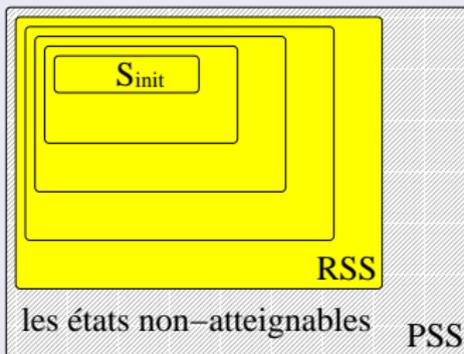
Clôture réflexive et transitive de la fonction \mathcal{N}

$$\text{RSS} = \mathcal{S}_{init} \cup \mathcal{N}(\mathcal{S}_{init}) \cup \mathcal{N}^2(\mathcal{S}_{init}) \cup \dots = \mathcal{N}^*(\mathcal{S}_{init})$$

Modèle états/transitions

- \mathcal{S}_{init} : l'ensemble d'états initiaux ($\mathcal{S}_{init} \subseteq \text{PSS}$)
- \mathcal{N} : la fonction “*next-state*” ($\text{PSS} \rightarrow 2^{\text{PSS}}$), où $\mathcal{N}(\tilde{x})$ détermine l'ensemble des états qui peuvent être **atteignables** à partir de l'état global \tilde{x}

Principe de la méthode de génération du RSS



Clôture réflexive et transitive de la fonction \mathcal{N}

$$\text{RSS} = \mathcal{S}_{init} \cup \mathcal{N}(\mathcal{S}_{init}) \cup \mathcal{N}^2(\mathcal{S}_{init}) \cup \dots = \mathcal{N}^*(\mathcal{S}_{init})$$

Démarche

Idée : utiliser les travaux de Ciardo basés sur l'utilisation de MDD [Ciardo *et al.* 01].

Diagrammes de Décision Multi-valués (MDD)

- Fonction : $\{0, 1, \dots, n_N\} \times \dots \times \{0, 1, \dots, n_1\} \rightarrow \{\mathbf{0}, \mathbf{1}\}$
- Structure arborescente (stockage compact)
- Fonction caractéristique d'ensemble dans un PSS
 - $\mathcal{S}^{(N)} \times \dots \times \mathcal{S}^{(1)} \rightarrow \{\mathbf{0}, \mathbf{1}\}$
- Manipulation ensembliste efficace (union et intersection)

Caractéristiques [Ciardo *et al.* 01]

- Fonction \mathcal{N} structurée : $\mathcal{N}(\tilde{x}) = \mathcal{N}_N(\tilde{x}) \times \dots \times \mathcal{N}_1(\tilde{x})$
 - $\mathcal{N}_i(\tilde{x})$ décrit le changement dans l'état local $x^{(i)}$ pour \tilde{x}
- Saturation : $(*, *, x^{(3)}) \xrightarrow{\mathcal{N}_3} (*, *, y^{(3)})$

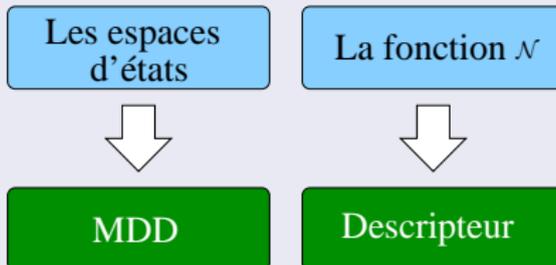
Problème

$$(*, *, x^{(3)}) \xrightarrow{\mathcal{N}_3} (*, *, y^{(3)}) \text{ dépendant de } f$$

Idée

Trouver une représentation structurée de la fonction \mathcal{N} qui prend en compte l'utilisation des **fonctions**, où on puisse appliquer le principe de la **saturation**.

Ce qu'on va utiliser



Obtention du descripteur d'atteignabilité R

- Formulation tensorielle obtenue à partir du descripteur Markovien Q

Descripteur d'atteignabilité R :

$$R = \sum_{j=1}^N \bigotimes_{i=1}^N \check{R}_j^{(i)} + \sum_{e \in \mathcal{E}_s} \bigotimes_{i=1}^N \check{R}_e^{(i)}, \quad \text{où } \check{R}_j^{(i)} = \begin{cases} \check{R}_l^{(i)} & \text{si } j = i \\ I_{|S^{(i)}|} & \text{si } j \neq i \end{cases}$$

- $\check{R}_l^{(i)}$ matrices d'atteignabilité locales
- $\check{R}_e^{(i)}$ matrices d'atteignabilité de l'événement synchronisant e
- Représentation par un MDD de l'espace d'états où la fonction ne s'annule pas (partitions)
- Décomposition du produit tensoriel de matrices selon ces partitions

Décomposition des matrices d'atteignabilité

On sépare les éléments d'une matrice par classes d'équivalence suivant l'espace d'états où ils sont identiquement nuls :

$$\check{R}_l^{(i)} = \sum_{c=1}^{C_l^{(i)}} \check{R}_{l,c}^{(i)} \quad \check{R}_e^{(i)} = \sum_{c=1}^{C_e^{(i)}} \check{R}_{e,c}^{(i)}$$

- $C_l^{(i)}$: le nombre de classes d'équivalence de $\check{R}_l^{(i)}$
- $C_e^{(i)}$: le nombre de classes d'équivalence de $\check{R}_e^{(i)}$

Théorème

Étant donné un modèle SAN à temps continu, le descripteur d'atteignabilité R de ce modèle est donné par :

$$R = \sum_{j=1}^N \sum_{c=1}^{C_l^{(j)}} \bigotimes_{i=1}^N \check{R}_{j,c}^{(i)} + \sum_{e \in \mathcal{E}_s} \sum_{c^{(1)}=1}^{C_e^{(1)}} \cdots \sum_{c^{(N)}=1}^{C_e^{(N)}} \bigotimes_{i=1}^N \check{R}_{e,c^{(i)}}^{(i)}$$

$$\text{où } \check{R}_{j,c}^{(i)} = \begin{cases} \check{R}_{l,c}^{(i)} & \text{si } j = i \\ I_{|S^{(i)}|} & \text{si } j \neq i \end{cases}$$

Dîner des philosophes (résultats)

Taux constants							
N	Mémoire (KiB)		Espace d'états		Temps (s)		
	Finale	Pic	RSS	PSS	Fonctions	RSS	Total
10	7	34	$5,7 \times 10^3$	$5,9 \times 10^4$	-	$1,54 \times 10^{-3}$	$1,54 \times 10^{-3}$
20	14	91	$3,9 \times 10^7$	$3,5 \times 10^9$	-	$3,36 \times 10^{-3}$	$3,36 \times 10^{-3}$
100	72	1 411	$1,6 \times 10^{38}$	$5,2 \times 10^{47}$	-	$1,84 \times 10^{-2}$	$1,84 \times 10^{-2}$
1 000	719	121 344	$5,0 \times 10^{382}$	$1,3 \times 10^{477}$	-	$2,44 \times 10^{-1}$	$2,44 \times 10^{-1}$

Taux fonctionnels							
N	Mémoire (KiB)		Espace d'états		Temps (s)		
	Finale	Pic	RSS	PSS	Fonctions	RSS	Total
10	7	34	$5,7 \times 10^3$	$5,9 \times 10^4$	$1,70 \times 10^{-4}$	$1,25 \times 10^{-3}$	$1,42 \times 10^{-3}$
20	14	86	$3,9 \times 10^7$	$3,5 \times 10^9$	$2,79 \times 10^{-4}$	$2,76 \times 10^{-3}$	$3,04 \times 10^{-3}$
100	72	1 155	$1,6 \times 10^{38}$	$5,2 \times 10^{47}$	$1,86 \times 10^{-3}$	$1,51 \times 10^{-2}$	$1,70 \times 10^{-2}$
1 000	719	94 425	$5,0 \times 10^{382}$	$1,3 \times 10^{477}$	$5,08 \times 10^{-2}$	$1,81 \times 10^{-1}$	$2,32 \times 10^{-1}$

Synthèse

- Descripteur d'atteignabilité (fonction \mathcal{N} structurée) : fonctions + saturation
- Méthode de génération efficace du RSS de modèles avec des taux fonctionnels
- Implantée sur le logiciel PEPS

<http://www-id.imag.fr/Logiciels/peps/>

- Validation : modèles SPN équivalents aux modèles SAN avec des taux constants (logiciel SMART)

<http://www.cs.ucr.edu/~ciardo/SMART/>

[Sales et Plateau 09]

Plan

- Contexte
- Génération du RSS
- **Calcul du vecteur de probabilité**
 - Situation entre événements en temps discret
 - Travaux précédents
 - Algèbre Tensorielle complexe (ATX)
 - Décomposition du produit tensoriel complexe en facteurs normaux
 - Multiplication vecteur-descripteur
 - Synthèse
- Conclusions et Perspectives

Rappel : Modèles à temps discret

Plusieurs événements peuvent avoir lieu dans le même instant

- Combinatoire d'événements

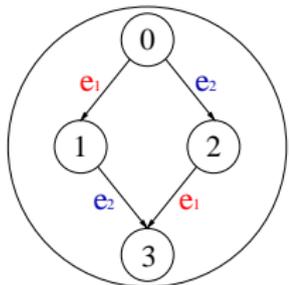
Différentes situations

- Simultanéité
- Conflit
- Concurrence

Situations entre événements dans la même unité de temps

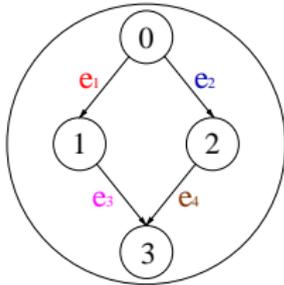
Simultanéité

Composant



Conflit

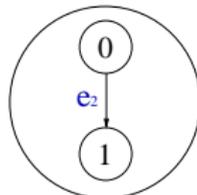
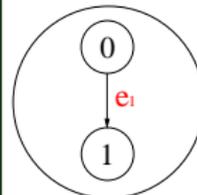
Composant



Concurrence

Composant 1

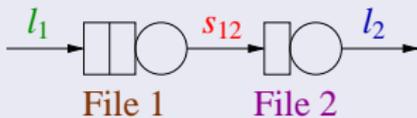
Composant 2



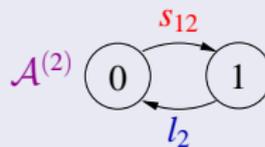
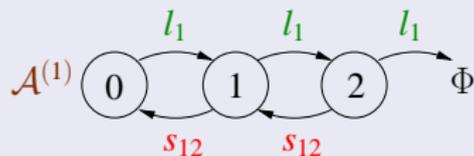
SAN à temps discret (travaux antérieurs)

- Restriction à des ensembles d'événements compatibles [Atif *et al.* 92]
 - Événements compatibles (= événements concurrents)
 - Utilisateur doit fournir les ensembles d'événements
 - Cas de conflit : choix aléatoire
 - Simultanéité est écartée
- Notion de priorité sur les événements [Benoit *et al.* 03]
 - Établit un ordre d'occurrence
 - Précise les cas de conflit
 - Introduit le déterminisme dans le modèle
 - Algorithme de génération extensive de la chaîne de Markov sous-jacente

Exemple : Modèle SAN à temps discret (approche priorité)



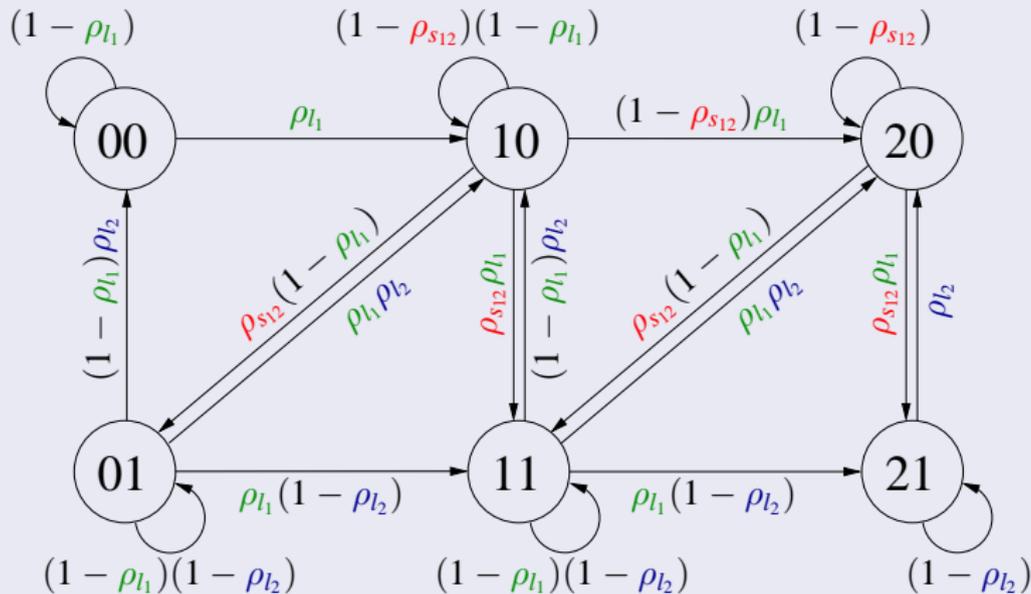
Type	Even.	Prob.	Prio.
syn	s_{12}	$\rho_{s_{12}}$	1
loc	l_1	ρ_{l_1}	2
loc	l_2	ρ_{l_2}	3



État fantôme Φ

L'état fantôme est un état intermédiaire utilisé pour rendre l'occurrence d'un événement possible après l'occurrence d'un événement plus prioritaire.

Exemple : Chaîne de Markov à temps discret



Jusqu'au travail de Benoit

Pas de représentation tensorielle (compacte) du modèle.

Limitation

- Algèbres classiques
 - Travaillent sur les *matrices numériques*
 - Ne peuvent pas capturer les aspects de *conflit, simultanéité et concurrence d'événements*

Solution proposée

Définir une nouvelle algèbre qui *travaille d'abord sur les événements*.

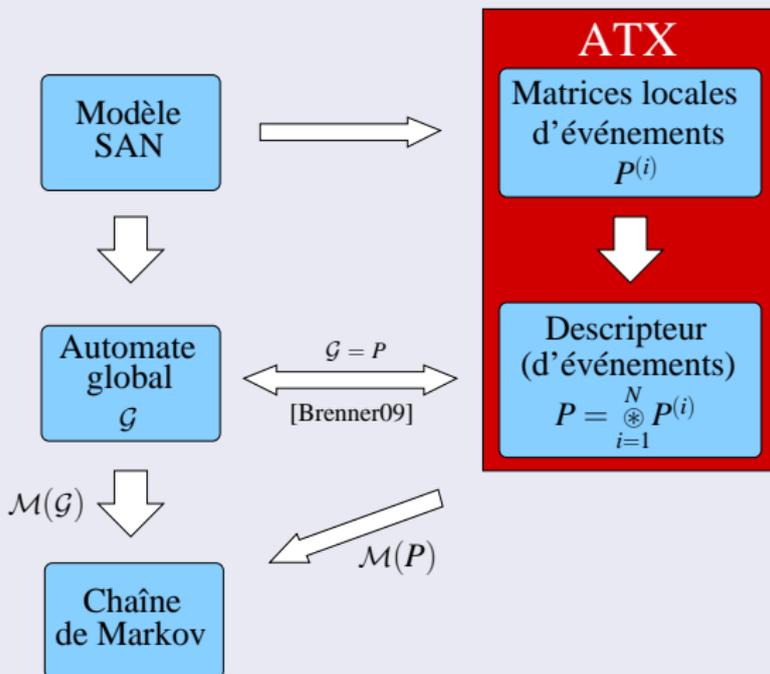
Algèbre sur les événements

- Opérateurs
 - Produit de simultanéité “ \cdot ”
 - Somme de choix “ $+$ ”
 - Produit de concurrence “ $*$ ”

Algèbre Tensorielle complexe (ATX)

- Opérateur matriciel
 - Produit tensoriel complexe “ \otimes ” (permet aussi l'utilisation de *probabilités fonctionnelles*)

Principe général



Retour au problème

$$\pi \times \mathcal{M}(P) = \pi' \text{ efficace}$$

- $\mathcal{M}(P)$ est la matrice de transition de la chaîne de Markov obtenue à partir du descripteur d'événements P .

Idée

Obtenir une méthode similaire à l'algorithme du *Shuffle* [Fernandes *et al.* 98] (modèles à temps continu) **décomposant le produit tensoriel complexe en facteurs normaux.**

Théorème

Étant donné des matrices d'événements $P^{(i)}$ et le produit tensoriel complexe, alors :

$$\begin{aligned}
 \bigotimes_{i=1}^N P^{(i)} &= P^{(1)} \otimes P^{(2)} \otimes \dots \otimes P^{(N)} \\
 &= (P^{(1)} \otimes I_e) * (I_e \otimes P^{(2)} \otimes I_e) * \dots * (I_e \otimes P^{(N)}) \\
 &= \text{FN}(P^{(1)}) * \text{FN}(P^{(2)}) * \dots * \text{FN}(P^{(N)})
 \end{aligned}$$

Matrice identité d'événements

$$I_e = \begin{vmatrix}
 \check{I} & \check{O} & \dots & \check{O} \\
 \check{O} & \check{I} & \dots & \check{O} \\
 \vdots & \vdots & \ddots & \vdots \\
 \check{O} & \check{O} & \dots & \check{I}
 \end{vmatrix}$$

- \check{I} : événement neutre
- \check{O} : événement nul

On s'intéresse à

$$\pi \times \mathcal{M}(P) = \pi'$$

On a effectivement

$$P = \bigotimes_{i=1}^N P^{(i)} = \bigotimes_{i=1}^N \text{FN}(P^{(i)})$$

Multiplication vecteur-descripteur

$$v^{(0)} = \pi_{\text{ev}}(\pi)$$

$$v^{(1)} = v^{(0)} * \text{FN}(P^{(1)})$$

$$v^{(2)} = v^{(1)} * \text{FN}(P^{(2)})$$

$$\vdots$$

$$v^{(i)} = v^{(i-1)} * \text{FN}(P^{(i)})$$

$$\vdots$$

$$v^{(N)} = v^{(N-1)} * \text{FN}(P^{(N)})$$

$$\pi' = \text{ev}_{\pi}(v^{(N)})$$

Les éléments intermédiaires du vecteur d'événements v

Expressions non-réduites de l'algèbre sous la forme :

$$\bigotimes_{i=1}^c \left[\sum_{j=1}^{s_c} \left(\sum_{k=1}^{p_{s_c}} e_{ijk} \right) \right]$$

Shuffle classique (temps continu)

Nombre de multiplications :

$$\prod_{i=1}^N n_i \times \sum_{i=1}^N \left(\frac{nz_i}{n_i} \right)$$

- n_i : la dimension de $Q^{(i)}$
- nz_i : le nombre d'éléments non-nuls de $Q^{(i)}$

Notre méthode

Nombre de *produits de concurrence* :

$$\prod_{i=1}^N n_i \times \sum_{i=1}^N \left(\frac{nz_i}{n_i} \times nze_i \right)$$

- n_i : la dimension de $P^{(i)}$
- nz_i : le nombre d'éléments non-nuls de $P^{(i)}$
- nze_i : le nombre d'expressions non-réduites de $v^{(i-1)}$,

$$nze_i = \prod_{k=1}^{i-1} \frac{nz_k}{n_k} \quad (\text{où } nze_1 = 1)$$

Synthèse

- Arithmétique sur les événements et matrices d'événements
- Formulation tensorielle du modèle SAN à temps discret
- Calcul direct sur les matrices d'événements

- Contexte
- Génération du RSS
- Calcul du vecteur de probabilité
- **Conclusions et Perspectives**
 - Génération du RSS
 - Calcul du vecteur de probabilité
 - Perspectives

Conclusion : Génération du RSS

- Descripteur d'atteignabilité (fonction \mathcal{N} structurée) : fonctions + saturation
- Méthode de génération efficace du RSS d'un modèle SAN avec des taux fonctionnels
- RSS représenté par un MDD : amélioration importante pour le logiciel PEPS

Notre méthode permet de calculer le RSS de modèles *plus grands* d'une façon *performante*, en conservant le *pic de mémoire* suffisamment *bas*.

Conclusion : Calcul du vecteur de probabilité

- Modélisation simples
- Algèbre Tensorielle complexe (ATX)
 - Puissante : opérations sur les événements et matrices d'événements
- Descripteur d'événements
 - Compact : un seul terme tensoriel
 - Combinatoire d'événements : transparente à l'utilisateur
- Multiplication vecteur-descripteur
 - L'idée de base du *Shuffle* (modèles à temps continu)
 - Suite d'opérations (de la taille d'un automate et pour tous les automates) au lieu des opérations sur l'espace d'états produit du modèle

Perspectives

Méthode de stockage pour des modèles structurés

- Diagrammes de matrices pour la fonction \mathcal{N} [Miner01]

Utilisation et enrichissement de l'ATX

- Implantation de la multiplication vecteur-descripteur
- Heuristique de simplification : pré-évaluation des fonctions
- Méthode de multiplication
 - Génération à la volée [Buchholz *et al.* 00]
- Comparaison avec d'autres formalismes
 - Expressivité de la nouvelle algèbre sur les événements
- Génération du RSS de modèles à temps discret

Merci pour votre attention