Réseaux d'automates stochastiques à temps discret

Anne Benoit

Laboratoire Informatique et Distribution, Grenoble

http://www.id-imag.fr

Motivations

- Croissance rapide des systèmes et des réseaux $\rightarrow$ problèmes critiques de performances
  - analyse fine du comportement : identifier les problèmes et les résoudre (monitoring, ...)
  - anticiper les problèmes de performance : modélisation du système et étude préalable des performances

Plan de l'exposé

- Introduction : la modélisation de systèmes
- Modèles à temps discret
- Réseaux d'automates stochastiques (SANs) à temps discret
- Construction de la chaîne de Markov équivalente
- Définition de la chaîne de Markov (preuve de cohérence de la sémantique)
- Conclusions et Perspectives

Modélisation – généralités

- **Complexité** des nouvelles générations de systèmes à concevoir $\rightarrow$ plusieurs techniques de modélisation
- Chaque technique doit définir :
  - les **états** du système (nombre fini)
  - les **transitions** entre chaque état (dynamique)
  - la **temporisation** des transitions (temps que l'on passe dans chaque état)
- hypothèse : système **Markovien**, i.e. sans mémoire

Modélisation – chaînes de Markov

- Chaînes de Markov: analyse des performances de systèmes dynamiques
- Systèmes à grand espace d'états (million d'états) : modélisation "à plat" impossible
- Formalismes de haut niveau structurés : génération de l'espace d'états et du générateur infinitésimal de la chaîne de Markov
- Algorithmes sophistiqués pour calculer des indices de performances

Modélisation – formalismes

- Réseaux de files d'attente : approche orientée “ressources consommées par des clients”
- Réseaux de Petri : analyse fine des synchronisations
- Algèbres de processus : composition concurrente, exécution parallèle
- Réseaux d'automates : intégration des synchronisations au modèle états-transitions

Modélisation – réseaux d'automates stochastiques

Formalisme des réseaux d'automates stochastiques (SANs) :

- Introduit dans les années 1980 par B. Plateau
- Modèles à temps continu : le générateur de la chaîne de Markov sous-jacente est sous forme tensorielle

Il est plus difficile de modéliser les systèmes à temps discret

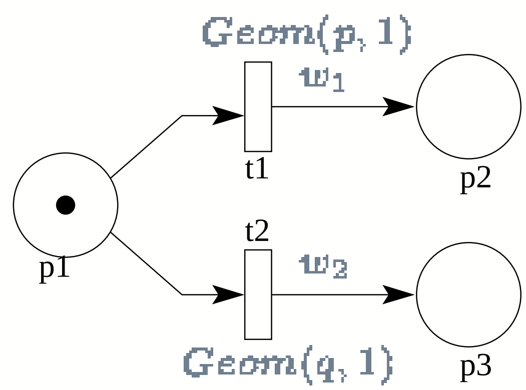
Modèles à temps discret

- Souci de simplification du modèle : le temps est regroupé en intervalles
  – agrégation temporelle
- Difficulté de modélisation : plusieurs événements peuvent avoir lieu pendant une même unité de temps
  – événements concurrents

$\rightarrow$ Peu de travaux sur ces modèles dans la littérature (en comparaison des modèles à temps continu)

Modèles à temps discret – réseaux de Petri

- Différentes **sémantiques** pour traiter les événements concurrents
- Transitions **concurrentes** : candidates au même instant

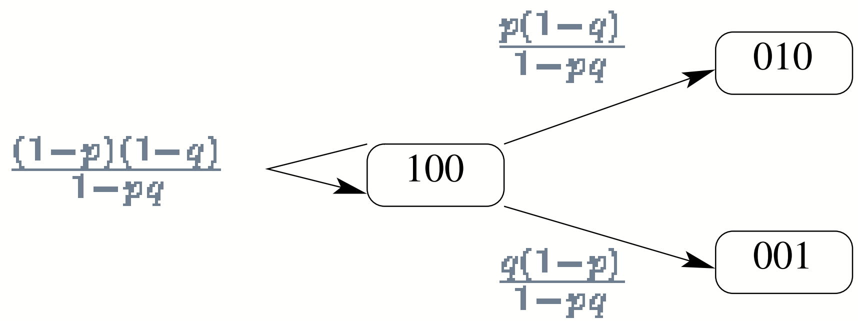


- Événements $e1$ et $e2$ associés à $t1$ et $t2$, probabilités d'occurrence respectives p et q

Modèles à temps discret – Molloy

- Travaux de Molloy

  Graphe de marquage du réseau de Petri :

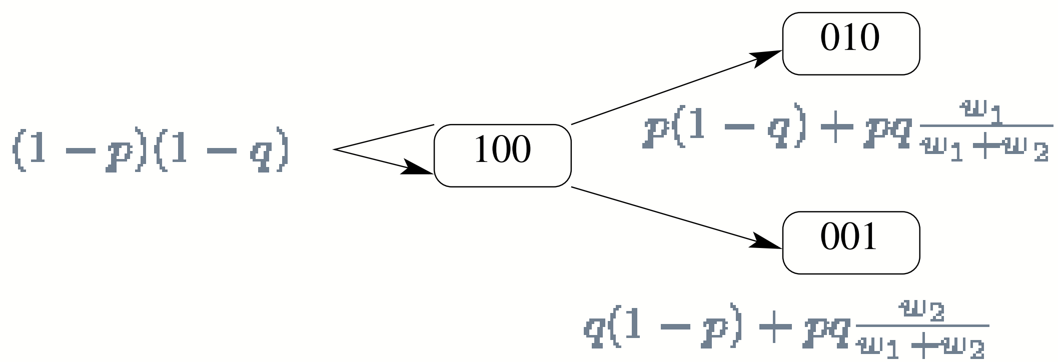


- Les événements concurrents $e1$ et $e2$ ne peuvent pas avoir lieu en même temps
- Division des probabilités par $(1 - pq)$ pour avoir une somme égale à 1

Modèles à temps discret – Ciardo

- Travaux de Ciardo

  Graphe de marquage du réseau de Petri :



- Poids $w1$ et $w2$ associés aux événements $e1$ et $e2$
- Les deux événements peuvent se déclencher en même temps (probabilité pq), la place du jeton dépend alors des poids

Modèles à temps discret – SANs

- Travaux de Atif
- Notion d'*événements compatibles* : identifier les événements pouvant se réaliser pendant une même unité de temps
- Il faut fournir l'ensemble des événements compatibles avec le modèle SAN
- Contrainte globale sur la somme des probabilités à vérifier
- sémantique peu précise en cas de conflit (quel événement a lieu en premier?)

SANs à temps discret – introduction

Nous proposons une refonte du formalisme de SANs à temps discret :

- Simplicité du modèle
- Détailler la sémantique en cas de conflit,
  notion de priorité sur les événements
- Algorithme de génération de la chaîne de Markov sous-jacente,
  identification automatique des événements concurrents

SANs à temps discret – généralités

- Intérêt des SANs : modélisation du système à l'aide de plusieurs composants indépendants qui tournent en parallèle et qui interagissent
- Composants : automates = états locaux et transitions, et un ensemble d'événements qui déclenchent les transitions
- Interactions : événements synchronisant (à l'opposé des événements locaux) et probabilités de transition fonctionnelles

SANs à temps discret – quelques définitions

- Réseau d'automates stochastiques :
  - N automates $\mathcal{A}^{(i)}$, $i = 1..N$
  - ensemble d'événements $\mathcal{E}$
- Automate stochastique $\mathcal{A}^{(i)}$:
  - ensemble d'états locaux $S^{(i)}$
  - ensemble de transitions $T^{(i)}$
- Espace d'états produit $\hat{S} = S^{(1)} \times ... \times S^{(N)}$
- État global $x = (x^{(1)}, ..., x^{(N)}) \in \hat{S}$

SANs à temps discret – quelques définitions

Un événement $e \in \mathcal{E}$ est défini par :

- une probabilité de transition pt_e *(fonction de* $\hat{S}$ *dans* $[0, 1]$*)*
- une priorité $prio_e$ *(détermine comment agir en cas de conflit,* $1 \leq prio_e < +\infty$*)*
- un ensemble d'automates impliqués par cet événement O_e *(l'occurrence de* e *entraîne une modification de l'état local de chacun de ces automates)*

SANs à temps discret – quelques définitions

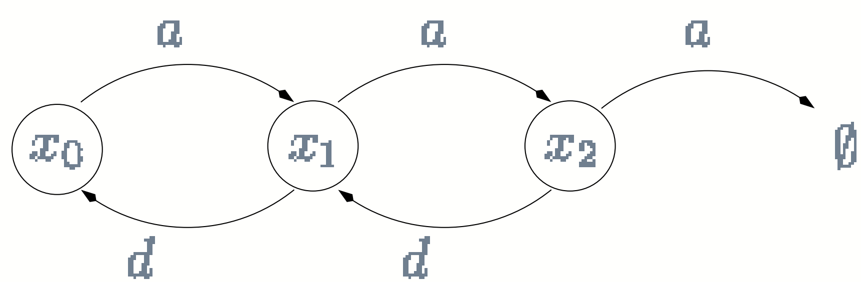
Une transition $t \in T^{(i)}$ est défini par :

- un état de départ $x_{t^{(i)},dep} \in S^{(i)}$ et un état d'arrivée $x_{t^{(i)},arr} \in S^{(i)} \cup \emptyset$
- un événement associé à la transition $evt_{t^{(i)}} = e \in \mathcal{E}$
- une probabilité de routage $\pi_{t^{(i)}}$ (fonction de $\hat{S}$ dans $[0, 1]$)

$\emptyset$ est un état particulier de chaque automate, appelé l'état vide

SANs à temps discret – exemples

Automate = graphe dont les noeuds sont les états et les arcs un ensemble de transitions, étiquetés par les événements associés à ces transitions.

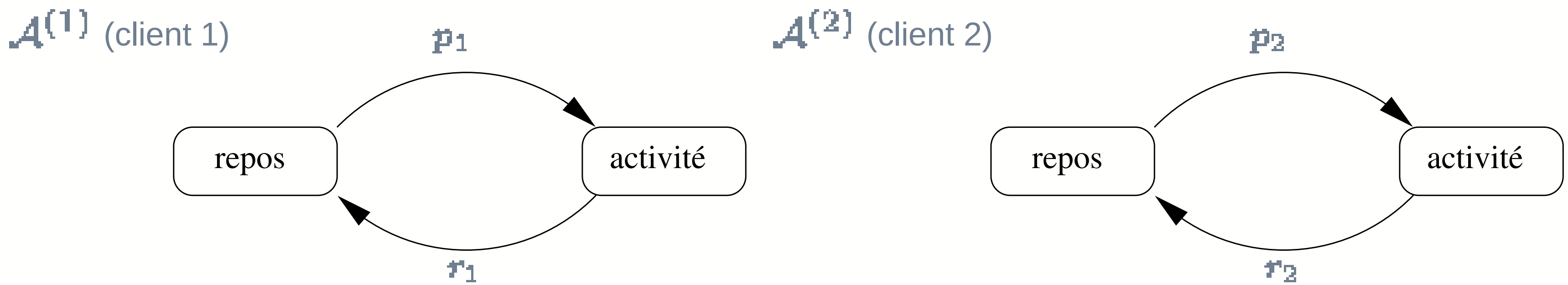


File d'attente de capacité 2, 3 états locaux et 2 événements

- a : événement d'arrivée d'un client
- d : événement de départ d'un client

SANs à temps discret – exemples

Exclusion mutuelle, 2 clients se partagent une unique ressource.



- p_i : événement de prise de ressource par le client i
- r_i : événement de relâche de ressource par le client i

Prise de ressource : probabilité de transition fonctionnelle f

f(état global) = δ (état($\mathcal{A}^{(1)}$) = *repos* et état($\mathcal{A}^{(2)}$) = *repos*)

Priorités : $prio_{r_1} = prio_{r_2} < prio_{p_1} < prio_{p_2}$

Construction de la chaîne de Markov équivalente

- états de la chaîne de Markov = états accessibles du SAN (exemple précédent : état *(activité, activité)* non accessible)
- génération de l'ensemble des états accessibles, et des transitions pour aller d'un état à un autre
- pour chaque état initial, on regarde vers où on peut aller et on calcule la probabilité de transition associée

Construction de la chaîne de Markov équivalente

État initial x, état courant $y = x$. On examine tous les événements possibles depuis x, $e \in EP(x)$ *(il existe un arc partant de x étiqueté par e)*, du plus au moins prioritaire.

- Si e est réalisable depuis y (*probabilité non nulle*), il peut avoir lieu ou non, probabilités $pt_e(y)$ et $1 - pt_e(y)$

$\rightarrow$ liste d'états intermédiaires obtenus par différents routages.

- Sinon, l'état ne change pas (état intermédiaire y).

États intermédiaires : nouveaux états courants. On traite alors les événements de EP(x) qui suivent par ordre de priorité.

Construction de la chaîne de Markov équivalente

Exemple de l'exclusion mutuelle

$L = [p_1, r_2]$
$E = [(ra, 1)]$

r_2 a lieu (μ) — (rr, μ)

$L = [p_1]$
$E = [(rr, \mu), (ra, 1-\mu)]$

p_1 a lieu (λ) — $(ar, \mu\lambda)$

p_1 n'a pas lieu ($1-\lambda$) — $(rr, \mu(1-\lambda))$

r_2 n'a pas lieu ($1-\mu$) — $(ra, 1-\mu)$ — p_1 n'est pas réalisable (proba. de transition $\lambda f(ra) = 0$) — $(ra, 1-\mu)$

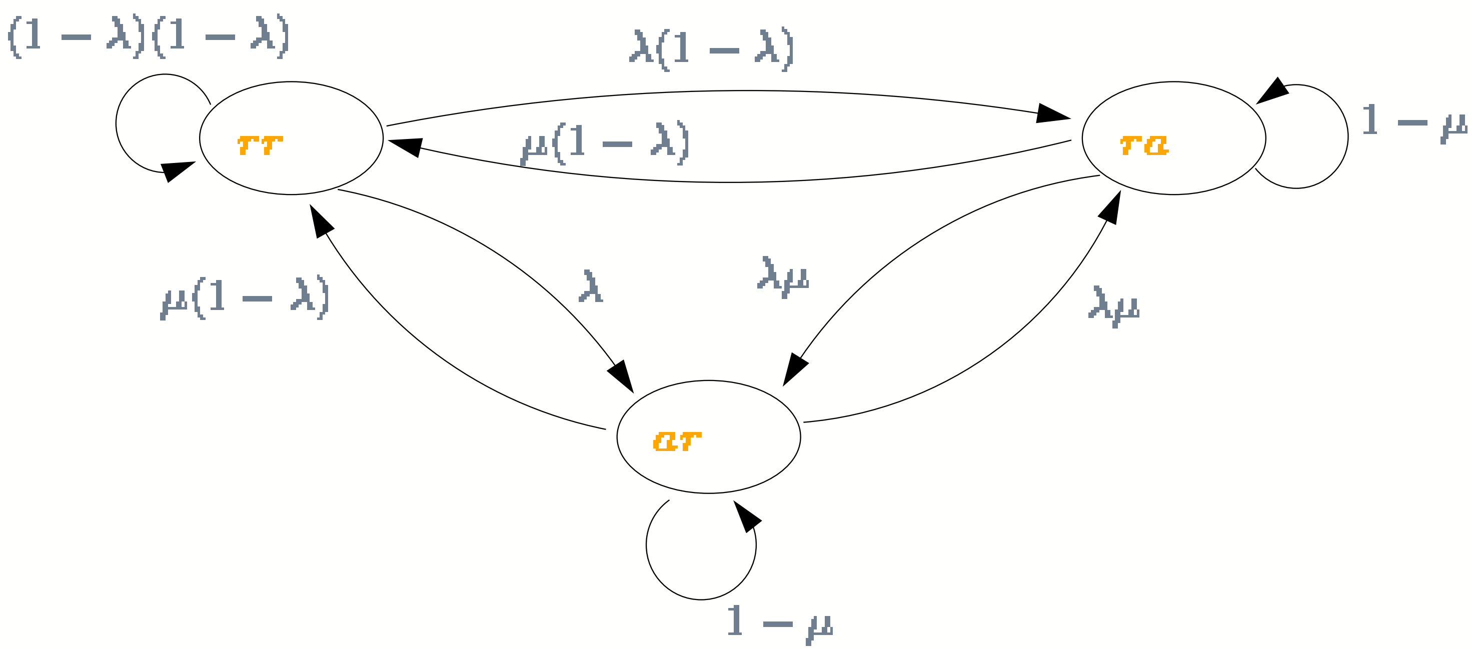
$L = [\,]$
$E = [(ar, \mu\lambda), (rr, \mu(1-\lambda)), (ra, 1-\mu)]$

A chaque niveau de l'arbre, on dispose de la liste E des états accessibles, pondérés par les probabilités.

On descend d'un niveau lorsqu'on traite un événement.

Construction de la chaîne de Markov équivalente

Exemple de l'exclusion mutuelle



Définition de la chaîne de Markov

- états = états accessibles du SAN (générés par l'algorithme)
- probabilités de transition : obtenues par l'algorithme
  - processus sans mémoire (les probabilités de transition ne dépendent que de l'état de départ)
  - somme des probabilités partant de chaque état égale à 1 (prouvé avec l'algorithme)

Conclusions

- Formalisme de SANs à temps discret
- Algorithme de génération de la chaîne de Markov :
  - génération des états accessibles
  - génération des événements concurrents
- Notion de priorités, détermine comment agir en cas de conflit :
  - attribution des priorités par l'utilisateur
  - attribution automatique pour les modèles plus complexes?

Perspectives

- Expression tensorielle pour représenter le descripteur d'un SAN à temps discret
  - nouveaux opérateurs tensoriels tenant compte des priorités
- Méthodes de résolution basées sur cette représentation tensorielle
  - méthodes similaires à l'algorithme du shuffle classique employé sur les modèles à temps continu

Des questions ?