Simulation parfaite de réseaux de Jackson à capacité bornée et à forte charge par un processus bornant

Ana Bušić Bruno Gaujal Florence Perronnin*

INRIA-ENS Paris INRIA Rhône-Alpes Université Joseph Fourier

Réunion plénière ANR Marmote

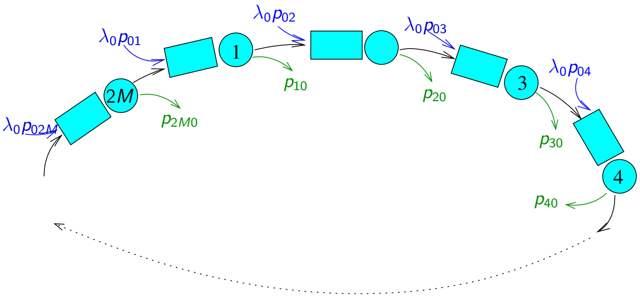
15 janvier 2015

Plan

Réseaux de Jackson

- Routage probabiliste
- Un client arrivant dans une file pleine est perdu.
- FIFO
- Capacité finie ou non

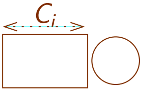
Exemple démo



Simulation parfaite à l'aide d'un processus bornant

Principe (rappel)

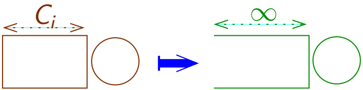
Si une file a une capacité C finie, on la remplace par une file infinie.



Simulation parfaite à l'aide d'un processus bornant

Principe (rappel)

Si une file a une capacité C finie, on la remplace par une file infinie.

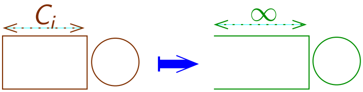


- La trajectoire du réseau infini est une borne supérieure de celle du réseau original.

Simulation parfaite à l'aide d'un processus bornant

Principe (rappel)

Si une file a une capacité C finie, on la remplace par une file infinie.

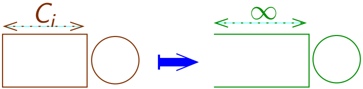


- La trajectoire du réseau infini est une borne supérieure de celle du réseau original.
- Le réseau de Jackson infini est à forme produit s'il est stable.

Simulation parfaite à l'aide d'un processus bornant

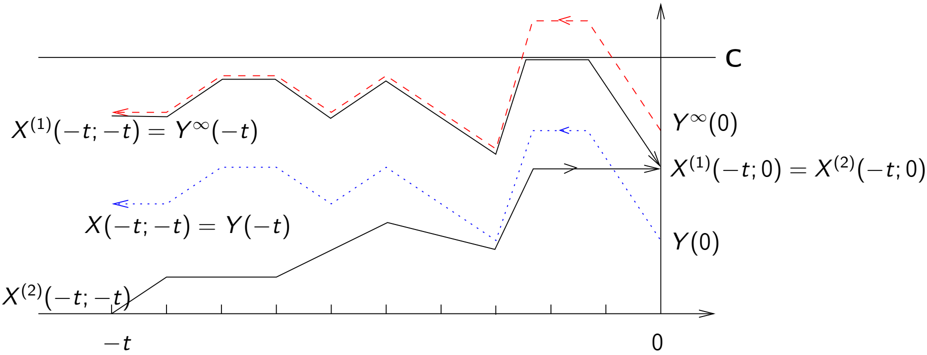
Principe (rappel)

Si une file a une capacité C finie, on la remplace par une file infinie.



- La trajectoire du réseau infini est une borne supérieure de celle du réseau original.
- Le réseau de Jackson infini est à forme produit s'il est stable.
- On sait renverser en temps le processus bornant stationnaire.

Simulation parfaite avec le processus bornant



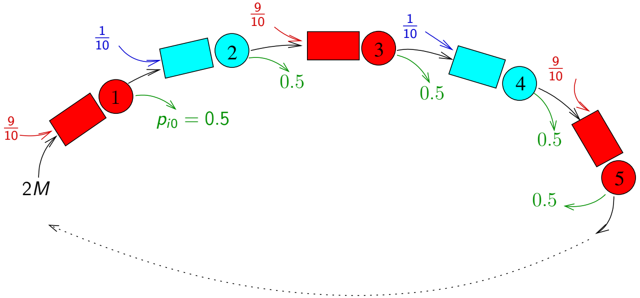
- État de départ << état maximum
- Durée de simulation indépendante de la capacité des files

Plan

Et si le réseau bornant n'est pas stable ?

Et si le réseau bornant n'est pas stable ?

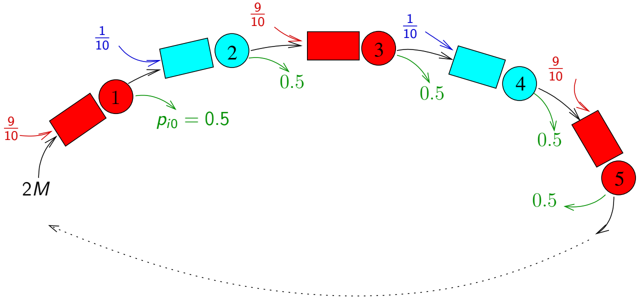
Exemple démo



- File i : capacité C_i
- Files impaires surchargées $\lambda_i \geqslant \mu_i$

Et si le réseau bornant n'est pas stable ?

Exemple démo

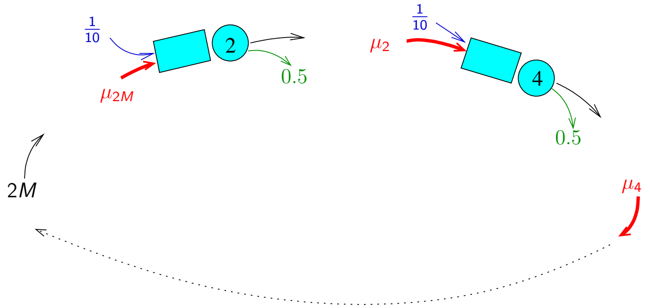


- File i : capacité C_i
- Files impaires surchargées $\lambda_i \geqslant \mu_i$

Équations de trafic :

$$\begin{cases} \forall i \neq 0,\ \lambda_i & = & \sum_{j=1}^{2M} p_{ji}(\lambda_j \wedge \mu_j) \\ \lambda_0 & = & \mu_0 \end{cases}$$

Astuce : remplacer les files instables par des sources



Le réseau de Jackson obtenu est stable.
Méthode :

1. Déterminer l'ensemble des files instables...
2. Processus bornant : files instables à l'état $+\infty$.

Plan

Équations de trafic pour les réseaux de Jackson instables

$$\begin{cases} \forall i \neq 0,\ \lambda_i & = & \sum_{j=1}^{2M} p_{ji}(\lambda_j \wedge \mu_j) \\ \lambda_0 & = & \mu_0 \end{cases}$$

- 1984 : Algorithme en $\mathcal{O}(M^4)$ [Goodman and Massey]
- 2014 : Nouvel algo en $\mathcal{O}(M^3)$

Algorithme en $\mathcal{O}(M^3)$

On définit les variables :

- $\mathcal{L}$ ensemble de files (initialement toutes les files)
- x solution du système suivant :

$$\left\{ \begin{array}{lll} x_0 & = & \mu_0 \\ x_i & = & \sum_{j \in \mathcal{L}} \mu_j p_{ji} + \sum_{j \notin \mathcal{L}} (x_j \wedge \mu_j) p_{ji} \quad \textit{for } i \neq 0 \end{array} \right.$$

La méthode consiste à :

1. Calculer le vecteur $\mathbf{x}$ solution du système
2. Trouver une file i^* dans $\mathcal{L}$ t.q. $i^* \in \mathcal{L}, x_{i^*} \leq \mu_{i^*}$.
3. Mettre à jour l'ensemble $\mathcal{L} = \mathcal{L} \backslash \{i^*\}$

Plan

Taux des événements en sens renversé

Pour renverser la trajectoire bornante (et obtenir un état de départ stationnaire) il faut déterminer les taux du réseau de Jackson en temps renversé :

File i stable

- Arrivées : $p_{i0}\lambda_i$
- Routage vers j instable : $\mu_i \mu_j \frac{p_{ji}}{\lambda_i}$
- Routage vers j stable : $\mu_i \frac{\lambda_j}{\lambda_i} p_{ji}$

File i instable

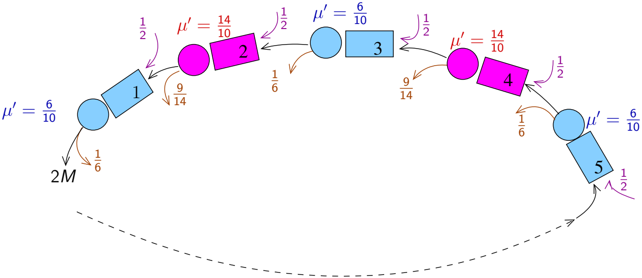
- Arrivées $p_{i0}\lambda_i$
- Routage vers j instable : $\mu_j p_{ji}$
- Routage vers j stable : $p_{ji}\lambda_j$

Plan

Si toutes les files (ou presque) sont instables ?

- Les remplacer par des sources puis simuler depuis C_i ne représente aucun gain
- Mais le réseau complémentaire, lui, est stable : $X \longmapsto C - X$

On simule les places vides au lieu des clients.

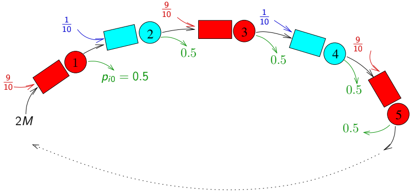


Plan

Quel réseau simuler ?

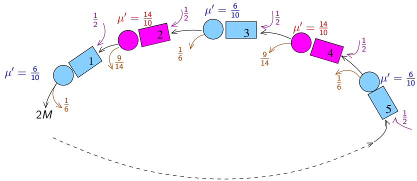
On ne sait pas *a priori* quel réseau sera plus rapide à simuler.

Complexité théorique



Taille des files stables : $\mathbb{E}[X_i] = \frac{3}{2}$

Borne sur le temps de coalescence $156M^2$



Taille des files stables : $\mathbb{E}[X_i] = \frac{5}{2}$

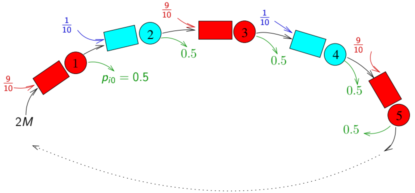
Borne sur le temps de coalescence $259M^2$

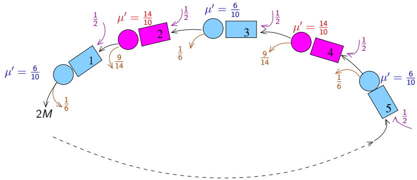
En pratique

Course entre la simulation du réseau et de son complémentaire

$M = 10$ (20 files), $C_i = 10$, 1000 réplications

Temps de coalescence





Files stables : $\mathbb{E}[X_i] = 0.46$

Files instables : $\mathbb{E}[X_i] = 4.91$

Temps de coalescence : $2629(\pm 100)$ | Temps de coalescence : $3842(\pm 100)$

Temps CPU : $2.2ms(\pm 36\mu s)$ | Temps CPU : $3.2ms(\pm 55\mu s)$