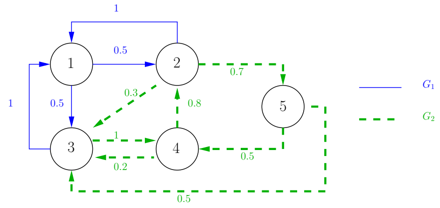
Perfect sampling for multiclass closed queueing networks

Anne Bouillard, **Ana Bušić**, **Christelle Rovetta**

MARMOTE, May 28, 2015

Network

- Multiclass closed queueing network
- K queues $./M/1$
- Z classes of customers
- Customers are not allowed to change class
- Preemptive service



State

- Queues have infinite capacity
- Population vector: $M = (M_1, M_2, \ldots, M_Z)$

Example

- $K = 5$, $Z = 2$
- $M = (2, 3)$
- A possible state: $\mathbf{x} = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 1 \end{pmatrix}$.

State space

- $\mathcal{S}$: set of all possible states

$$|\mathcal{S}| = \prod_{z=1}^{Z} \binom{M_z}{M_z + |K(z)| - 1}$$

- $|\mathcal{S}| = O(\prod_{z=1}^{Z} M_z^{|K(z)|})$
- $|\mathcal{S}| = O(M_\times^K)$, where $M_\times = \prod_{z=1}^{Z} M_z$

Example

- $K = 5,\ Z = 2,\ M = (2, 3)$

$$|\mathcal{S}| = \binom{2}{2+3-1} \times \binom{3}{3+4-1} = 120$$

Routing of class z

- Routing of a class z customer from queue i to queue j:

$$\widetilde{t}_{i,j,z}(\mathbf{x}) = \mathbf{x} - \mathbb{1}_{\{x_{z,i}>0\}}\mathbf{e}_{zi} + \mathbb{1}_{\{x_{z,i}>0\}}\mathbf{e}_{zj}$$

Example

$$\widetilde{t}_{2,1,1}(\begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 1 \end{pmatrix}) = \begin{pmatrix} 1+1 & 1-1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 1 \end{pmatrix}$$

Service discipline

- $\mathbf{x}_{*,k} = (x_{1,k}, \ldots, x_{Z,k})^t \in \mathbb{N}^Z$
- Function that describes the discipline in queue i:

$$f_i : \mathbb{N}^Z \to \{0, \ldots, Z\}$$

- We assume that:
  1. The service discipline is Markovian.
  2. $|\mathbf{x}_{*,i}| = 0$ if and only if $f_i(\mathbf{x}_{*,i}) = 0$
  3. If $|\mathbf{x}_{*,i}| > 0$ then $f_i(\mathbf{x}_{*,i}) \in \{z \text{ such that } x_{z,i} > 0\}$
- PRIORITY: $f_i(\mathbf{x}_{*,i}) = \min\{z \mid x_{z,i} > 0\}\mathbb{1}_{\{|\mathbf{x}_{*,i}|>0\}}$

Transition function

- $i \in \{1, \ldots, K\}$ a departure queue
- $J = (j_1, \ldots, j_Z) \in \{1, \ldots, K\}^Z$ destinations vector
- Transition function: $t_{i,J}(\mathbf{x}) = \widetilde{t}_{i,j_z,z}(\mathbf{x})$
  - $z = f_i(\mathbf{x}_{*,i})$
  - $\widetilde{t}_{i,j,z}(\mathbf{x}) = \mathbf{x} - \mathbb{1}_{\{x_{z,i}>0\}}\mathbf{e}_{zi} + \mathbb{1}_{\{x_{z,i}>0\}}\mathbf{e}_{zj}$

Example

- $\mathbf{x} = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 1 \end{pmatrix}$, queue 2 has the PRIORITY discipline
- $t_{2,(1,5)}(\mathbf{x}) =?$

Transition function

- $i \in \{1, \dots, K\}$ a departure queue
- $J = (j_1, \dots, j_Z) \in \{1, \dots, K\}^Z$ destinations vector
- Transition function: $t_{i,J}(\mathbf{x}) = \widetilde{t}_{i,j_z,z}(\mathbf{x})$
  - $z = f_i(\mathbf{x}_{*,i})$
  - $\widetilde{t}_{i,j,z}(\mathbf{x}) = \mathbf{x} - \mathbb{1}_{\{x_{z,i}>0\}}\mathbf{e}_{zi} + \mathbb{1}_{\{x_{z,i}>0\}}\mathbf{e}_{zj}$

Example

- $\mathbf{x} = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 1 \end{pmatrix}$, queue 2 has the PRIORITY discipline
- $t_{2,(1,5)}(\mathbf{x}) = ?$
  - $f_2(\mathbf{x}_{*,2}) = 1$

Transition function

- $i \in \{1, \ldots, K\}$ a departure queue
- $J = (j_1, \ldots, j_Z) \in \{1, \ldots, K\}^Z$ destinations vector
- Transition function: $t_{i,J}(\mathbf{x}) = \widetilde{t}_{i,j_z,z}(\mathbf{x})$
  - $z = f_i(\mathbf{x}_{*,i})$
  - $\widetilde{t}_{i,j,z}(\mathbf{x}) = \mathbf{x} - \mathbb{1}_{\{x_{z,i}>0\}}\mathbf{e}_{zi} + \mathbb{1}_{\{x_{z,i}>0\}}\mathbf{e}_{zj}$

Example

- $\mathbf{x} = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 1 \end{pmatrix}$, queue 2 has the PRIORITY discipline
- $t_{2,(1,5)}(\mathbf{x}) = ?$
  - $f_2(\mathbf{x}_{*,2}) = 1$
  - $t_{2,(1,5)}(\mathbf{x}) = \widetilde{t}_{2,1,1}(\mathbf{x}) = \begin{pmatrix} 2 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 1 \end{pmatrix}$

Markov chain

- $(U_n)_{n\in N} = (i_n, J_n)_{n\in N}$ an i.i.d sequence of random variables

$$\mathbb{P}(U_n = (i, J)) = \frac{\mu_i}{\sum_{k=1}^{K} \mu_k} \prod_{z=1}^{Z} P^z_{i,j_z}$$

- The evolution of the system can be described by an ergodic Markov chain:

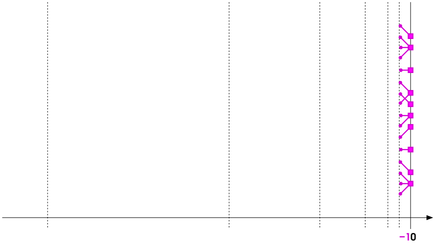
$$\begin{cases} X_0 \in \mathcal{S} \\ X_{n+1} = t_{U_n}(X_n) \end{cases}$$

- GOAL: sample the stationary distribution with the perfect sampling algorithm

Perfect sampling algorithm

Algorithm

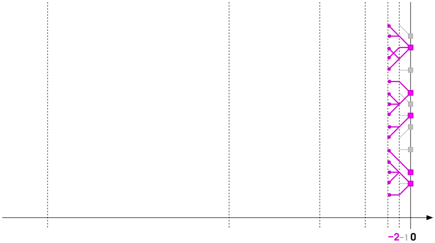
1. $n \leftarrow 1$
2. $t \leftarrow t_{U_{-1}}$
3. While $|t(\mathcal{S})| \neq 1$
4. $\quad n \leftarrow 2n$
5. $\quad t \leftarrow t_{U_{-n}}$
6. Return $t(\mathcal{S})$



Perfect sampling algorithm

Algorithm

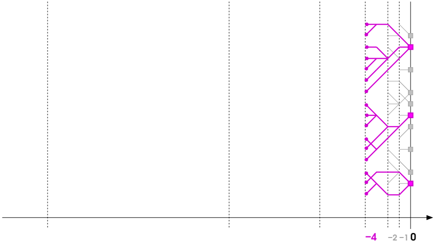
1. $n \leftarrow 1$
2. $t \leftarrow t_{U_{-1}}$
3. While $|t(\mathcal{S})| \neq 1$
4. $\quad n \leftarrow 2n$
5. $\quad t \leftarrow t_{U_{-n}}$
6. Return $t(\mathcal{S})$



Perfect sampling algorithm

Algorithm

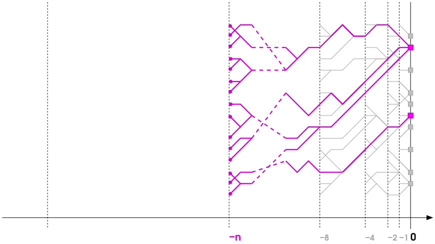
1. $n \leftarrow 1$
2. $t \leftarrow t_{U_{-1}}$
3. While $|t(\mathcal{S})| \neq 1$
4. $\quad n \leftarrow 2n$
5. $\quad t \leftarrow t_{U_{-n}}$
6. Return $t(\mathcal{S})$



Perfect sampling algorithm

Algorithm

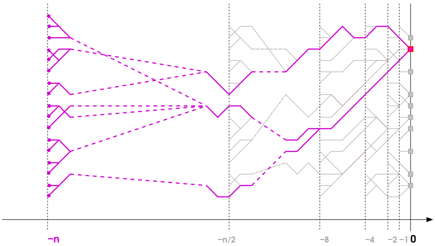
1. $n \leftarrow 1$
2. $t \leftarrow t_{U_{-1}}$
3. While $|t(\mathcal{S})| \neq 1$
4. $\quad n \leftarrow 2n$
5. $\quad t \leftarrow t_{U_{-n}}$
6. Return $t(\mathcal{S})$



Perfect sampling algorithm

Algorithm

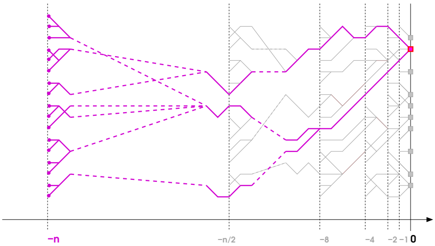
1. $n \leftarrow 1$
2. $t \leftarrow t_{U_{-1}}$
3. While $|t(\mathcal{S})| \neq 1$
4. $\quad n \leftarrow 2n$
5. $\quad t \leftarrow t_{U_{-n}}$
6. Return $t(\mathcal{S})$



Perfect sampling algorithm

Algorithm

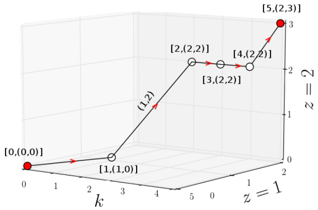
1. $n \leftarrow 1$
2. $t \leftarrow t_{U_{-1}}$
3. While $|t(\mathcal{S})| \neq 1$
4. $\quad n \leftarrow 2n$
5. $\quad t \leftarrow t_{U_{-n}}$
6. Return $t(\mathcal{S})$



- PROBLEM: $|\mathcal{S}|$

State as path

- State: $\mathbf{x} = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 1 \end{pmatrix}$
- Path:



- g: function which associates a set of arcs to a state

Diagram

- $N \subseteq \{0, \ldots, K\} \times \mathbb{N}^Z$: set of nodes
- $g : \mathcal{S} \to \mathcal{P}(N^2)$

Definition

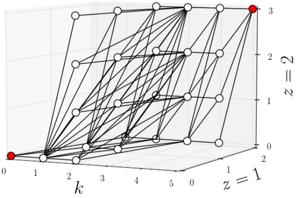
A directed graph $D = (N, A)$ is called a **diagram** if $\exists S \subseteq \mathcal{S}$:

$$A = g(S) := \bigcup_{\mathbf{x} \in S} g(\mathbf{x}).$$

Complete diagram

Definition

A diagram $D = (N, A)$ is said to be **complete** if $A = g(\mathcal{S})$.



Number of arcs

Lemma

Let $D = (N, A)$ be a diagram. If $K \geq 2$, then

$$|A| \leq 2 \prod_{z=1}^{Z} (M_z + 1) + (K - 2) \prod_{z=1}^{Z} \frac{(M_z + 1)(M_z + 2)}{2}.$$

- Complexity: $|A| = O(KM_\times^2)$

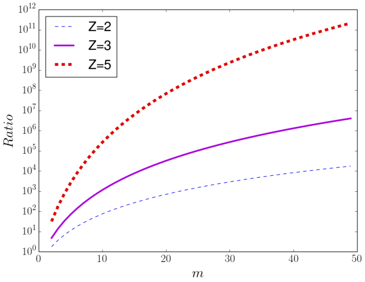
Example

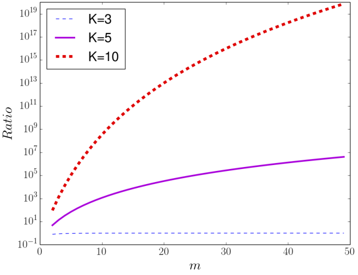
- Number of arcs in a complete diagram: $|A| = 71$
- Size of the state space: $|\mathcal{S}| = 120$

Complexity

- $M_\times = \prod_{z=1}^{Z} M_z$
- Complexity:

Diagram $|A| = O(KM_\times^2)$
State $|\mathcal{S}| = O(M_\times^K)$





- Ratio $\frac{|\mathcal{S}|}{|A|}$, $K = 5$
- Ratio $\frac{|\mathcal{S}|}{|A|}$, $Z = 3$

Set of states and diagram

- Function ϕ associates to a set of states $S \in \mathcal{S}$ the diagram:

$$\phi(S) = (N, g(S))$$

- Function ψ transforms diagram $D = (N, A)$ into the largest set of states $S \subseteq \mathcal{S}$ such that $g(S) = A$:

$$\psi(D) = \bigcup_{S \subseteq \mathcal{S},\ A = g(S)} S$$

Transition function

- (i, J) a couple departure-destinations
- Transition function:

$$T_{i,J}(D) = \phi \circ t_{i,J} \circ \psi(D)$$

Lemma

Let $S \subseteq \mathcal{S}$ be a set of states and D be a diagram, $\forall (i, J) \in \{1, \ldots, K\}^{Z+1}$:

1. *if $S \subseteq \psi(D)$ then $t_{i,J,}(S) \subseteq \psi(T_{i,J}(D))$;*
2. *if $|\psi(D)| = 1$ then $|\psi(T_{i,J,}(D))| = 1$.*

Transition function

- (i, J) a couple departure-destinations
- Transition function:

$$T_{i,J}(D) = \phi \circ t_{i,J} \circ \psi(D)$$

Lemma

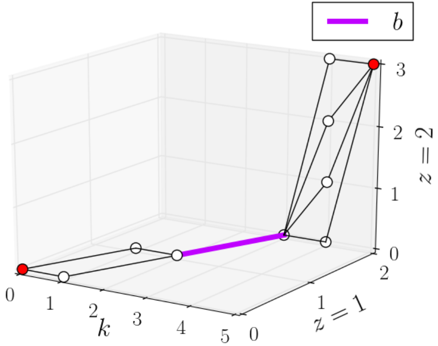
Let $S \subseteq \mathcal{S}$ be a set of states and D be a diagram, $\forall (i, J) \in \{1, \ldots, K\}^{Z+1}$:

1. *if $S \subseteq \psi(D)$ then $t_{i,J,}(S) \subseteq \psi(T_{i,J}(D))$;*
2. *if $|\psi(D)| = 1$ then $|\psi(T_{i,J,}(D))| = 1$.*

- Transition algorithm does not use the formula

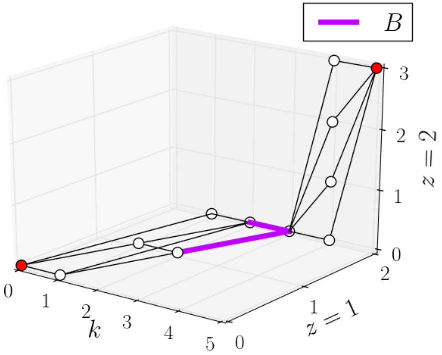
Paths

- $\mathcal{P}\text{aths}(b, A) = \{a \in A \mid \exists \mathbf{x} \in \mathcal{S} \text{ s.t. } a \in g(\mathbf{x}) \text{ AND } b \in g(\mathbf{x})\}$



Paths

- $\mathcal{P}\mathrm{aths}(B, A) = \bigcup_{b \in B} \mathcal{P}\mathrm{aths}(b, A)$



Discipline service

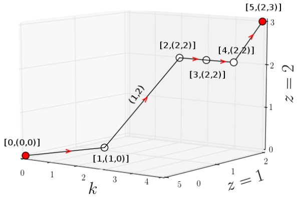
- For $a = \big([k-1, \mathbf{s}], [k, \mathbf{d}]\big) \in A$, we define:

$$v(a) = \mathbf{d} - \mathbf{s} \in \mathbb{N}^Z$$

- $\mathbf{x}_{*,k} = (x_{1,k}, \ldots, x_{Z,k})^t \in \mathbb{N}^Z$
- Function that describes the discipline in queue i:

State $f_i : \mathbb{N}^Z \to \{0, \ldots, Z\}$

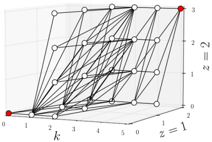
Diagram $F_i : \mathbb{N}^Z \to \{0, \ldots, Z\}$, $F_i(a) = f_i(v(a))$



Transition algorithm

Algorithm T_{iJ} on $D = (N, A)$

- Transition $T_{2,(1,5)}$

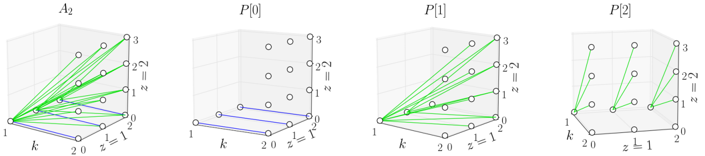


Transition algorithm

Algorithm T_{iJ} on $D = (N, A)$

1. Split A_i into $Z + 1$ subsets: $P[z] \leftarrow \{a \in A_i \mid F_i(a) = z\}$

- Transition $T_{2,(1,5)}$

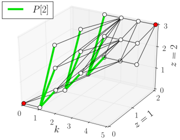


Transition algorithm

Algorithm T_{iJ} on $D = (N, A)$

1. Split A_i into $Z + 1$ subsets: $P[z] \leftarrow \{a \in A_i \mid F_i(a) = z\}$
2. For $z \in \{0, \ldots, Z\}$ compute $\mathcal{S}\mathrm{erve}[z] \leftarrow \mathcal{P}\mathrm{aths}(P[z], A)$

- Transition $T_{2,(1,5)}$

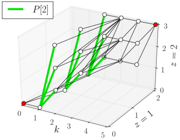


Transition algorithm

Algorithm T_{iJ} on $D = (N, A)$

1. Split A_i into $Z + 1$ subsets: $P[z] \leftarrow \{a \in A_i \mid F_i(a) = z\}$
2. For $z \in \{0, \dots, Z\}$ compute $\mathcal{S}\mathrm{erve}[z] \leftarrow \mathcal{P}\mathrm{aths}(P[z], A)$
3. Compute $\mathcal{S}\mathrm{erve}'[z]$

- Transition $T_{2,(1,5)}$

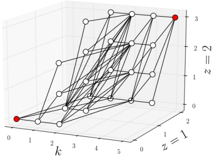


Transition algorithm

Algorithm T_{iJ} on $D = (N, A)$

1. Split A_i into $Z + 1$ subsets: $P[z] \leftarrow \{a \in A_i \mid F_i(a) = z\}$
2. For $z \in \{0, \ldots, Z\}$ compute $\mathcal{S}\mathrm{erve}[z] \leftarrow \mathcal{P}\mathrm{aths}(P[z], A)$
3. Compute $\mathcal{S}\mathrm{erve}'[z]$
4. Compute $A' \leftarrow \bigcup_{z=0}^{Z} \mathcal{S}\mathrm{erve}'[z]$
5. Return $D' = (N, A')$

- Transition $T_{2,(1,5)}$



Perfect sampling algorithm

States

1. $n \leftarrow 1$
2. $t \leftarrow t_{U_{-1}}$
3. While $|t(\mathcal{S})| \neq 1$
4. $\quad n \leftarrow 2n$
5. $\quad t \leftarrow t_{U_{-n}}$
6. Return $t(\mathcal{S})$

Diagram

1. $n \leftarrow 1$
2. $T \leftarrow T_{U_{-1}}$
3. While $|\psi(T(\mathcal{S}))| \neq 1$
4. $\quad n \leftarrow 2n$
5. $\quad T \leftarrow T_{U_{-n}}$
6. Return $\psi(T(\mathcal{S}))$

Clones

- Clones is a Matlab toolbox to make CLOsed queueing Networks Exact Sampling available:

  http://www.di.ens.fr/~rovetta/Clones/

- Monoclass case, infinite/finite capacity, multiserver (V2.0)
- Represent sets of states and diagrams as matrix

PyClones

- PyClones is a Python package to make CLOsed queueing Networks Exact Sampling (Beta)
- Multiclass, infinite capacity
- Represent sets of states and diagrams as objects
  - Compute transitions
  - Plot diagrams
  - Compute ψ function and $Card(\psi)$
  - Carry out exact sampling

Conclusion

- Definition of the model
- Multiclass diagram
- PyClones

Conclusion

- Definition of the model
- Multiclass diagram
- PyClones
- TP DEMAIN !!!!!

Merci pour votre attention !