

Perfect simulation of queueing networks with blocking and rejection

Jean-Marc Vincent*

Laboratoire ID-IMAG,Projet INRIA APACHE,
51, avenue Jean Kuntzmann, F-38330 Montbonnot, France
Jean-Marc.Vincent@imag.fr

Abstract

Traditional product form property of Markovian queueing networks usually is vanished when capacity of queues are finite and clients are blocked or rejected. A new efficient simulation method, derived from Propp & Wilson [11] perfect simulation, is applied to the finite capacity queues context. We present an algorithm to sample directly states of the network according to stationary distribution. This method has been applied to large queueing networks and some examples are given : loss estimation on Erlang models, usage of the last queue in a line of queues with blocking, saturation estimation for a multi-stage interconnection switch.

1. Introduction

Queueing systems are of fundamental interest for modeling communication networks. Servers represent the access to links and queue capacity allows modeling of resource contention and storage before transmission. Two kinds of dimensioning are needed for network optimization. Time dimensioning have to fix servers speed and space dimensioning define memory capabilities of nodes. In all cases, the estimation of service quality are useful before the network deployment.

Under Markovian assumptions (Poisson arrivals, exponential service time, probabilistic routing etc.), it has been shown that the network of queues is modelled by a multi-dimensional Markov jump process. Then the quality of service is computed from the steady-state distribution of the process. Fortunately, when queues have an infinite capacity the steady-state distribution is product-form and could easily be computed in a reasonable time [4]. In some cases the hypothesis of infinite capacity could be released, preserving the product form [10, 2]. Unfortunately, in most cases,

the steady state distribution is not in a product form and adequate approximation techniques should be applied. Many works cover the domain of queueing networks with finite capacity, bibliographies of [9, 1] provide pointers to related works.

Simulation approaches are alternative methods to estimate quality of service of such networks. Based on discrete event simulation [3] or on Markov properties (MCMC methods) [6], simulations estimate the steady-state distribution on long run trajectories.

In this paper a specific simulation technique, “perfect simulation”, is used to sample the steady state distribution. Based on Propp & Wilson ideas [11], an algorithm is proposed. By proving the monotonicity of queueing networks with blocking and rejection, we improve drastically simulation time. This technique have been applied on classical models to validate statistically the approach and on huge models to demonstrate the efficiency of the method.

2. Multi-dimensional Markov jump process

Consider a queueing network with K queues. The state space of each queue Q_i is the set of integers $\mathcal{X}_i = \{0, \dots, C_i\}$, where C_i is the capacity of queue Q_i . The state space \mathcal{X} of the system is the Cartesian product of all \mathcal{X}_i ;

$$\mathcal{X} = \mathcal{X}_1 \times \dots \times \mathcal{X}_K.$$

The natural order on integer is extended to a partial order on \mathcal{X} using component-wise ordering.

Definition 1 (Event)

*An event e is an application defined on \mathcal{X} , that associates to each state $x \in \mathcal{X}$ a new state denoted by $\Phi(x, e)$. Φ is called the **transition function** of the system.*

For example, a packet arrival, an end of service and routing, a packet exit, are typical events in networks. The transition corresponding to an arrival in queue Q_i is an increment of x_i provided that $x_i < C_i$. In that case one should precise the routing policy (rejection, overflow on another queue,...).

* This work was partially supported by ACI SurePath project and DECORE-IMAG project

Denote by $\mathcal{E} = \{e_1, \dots, e_p\}$ the set of events. This set is supposed to be finite.

Definition 2 (Monotonous events)

An event $e \in \mathcal{E}$ is said to be **monotonous** if it preserves the partial ordering on \mathcal{X} . That is

$$\forall (x, y) \in \mathcal{X} \quad x \leq y \Rightarrow \Phi(x, e) \leq \Phi(y, e).$$

If all events are monotonous, the global system is said to be monotonous.

Now, to achieve the model construction, a Poisson process with intensity λ_j is associated to each event e_j . These Poisson processes are supposed independent. For example consider the following system described in figure 1.

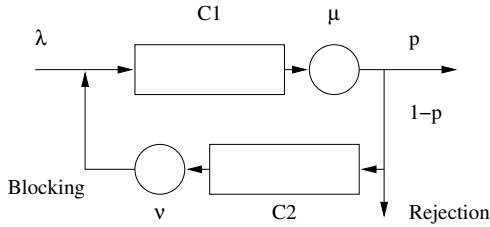


Figure 1. Queues with 2 types of contention

There are 4 events:

- e_1 packet external arrival with rate $\lambda_0 = \lambda$.
- e_2 end of service of the packet on the first queue and routing outside the network with rate $\lambda_1 = p\mu$.
- e_3 end of service on the first queue and routing to the second queue, if the second queue is full the packet is lost. The rate associated to e_2 is $\lambda_2 = (1-p)\mu$.
- e_4 end of service on the second queue, if the first queue is full the packet is blocked in the second queue and start again a service. The rate associated to e_3 is ν .

End of service events, e_1, e_2 and e_3 make a transition if the corresponding queue is not empty. When the queue is empty the event is just a skip operation.

Theorem 1 (Uniformized process)

The uniformized process driven by the Poisson process with rate $\Lambda = \sum_{j=1}^p \lambda_j$ and generating at each time of the process an event $e \in \mathcal{E}$ according to the probability distribution $(\frac{\lambda_1}{\Lambda}, \dots, \frac{\lambda_p}{\Lambda})$ is equivalent to the queueing network Markov process.

The proof of this result is obtained by writing down the infinitesimal transition equations. One should notice that the idea is to introduce the independence between events, when an event could not be applied to one state x the state is not changed, the method is analogous with the rejection method in stochastic simulation.

Proposition 1 (Uniformized monotonous system)

Provided that the system is monotonous, the discrete time Markov chain embedded in the uniformized process is also monotonous.

Consider now a typical routing event e , occurring in a queueing network. This event is defined by the origin queue Q_i and a list of destinations $Q_{j_1}, Q_{j_2}, \dots, Q_{j_i}$ with the semantic :

- if Q_i is empty do nothing;
- if Q_i is not empty find the first non empty queue Q_{j_k} in the list, decrements the number of packets in Q_i and increments the number of packets in Q_{j_k}
- if all queues are full then reject the packet out of the network.

Proposition 2 (Monotonicity of routing and rejection)

A routing event with rejection if all destination queues are full is a monotone event.

The proof is done by exploring all possibilities. Let $(x, y) \in \mathcal{X}^2$, such that $x \leq y$. Let e be a routing event defined by its origin Q_i and its list of destinations $Q_{j_1}, Q_{j_2}, \dots, Q_{j_i}$.

- if $y_i = 0$ then $x_i = 0$ and the event does not modify state x and y ; $\Phi(x, e) = x \leq \Phi(y, e) = y$;
- if $y_i > 0$ and $x_i \geq 0$, then $\Phi(y, e)|_i = y_i - 1 \geq x_i - 1 = Phi(x, e)|_i$. Let Q_{j_k} the first non saturated queue in state y . Because $x \leq y$ the first non saturated queue for state x is either strictly before Q_{j_k} in the list and the corresponding queue is saturated in state y or Q_{j_k} is the first non saturated queue in state x and y . In both cases, the order is preserved.

Proposition 3 (Monotonicity of routing and blocking)

A routing event with blocking in the original queue if all destination queues are full is a monotone event.

The proof is similar to the proof of the previous proposition.

Clearly arrivals from outside or routing directly outside are also monotone events.

Theorem 2 (Monotonous Markovian networks)

A Markovian network of queues with rejection routing or blocking is monotonous.

This result is clear according to the previous propositions. This result is an extension of a work from D. Mattson [8]

3. Perfect sampling method

Formally, when all the knowledge of the dynamics is included in the state description, the system can be described by a transition function Φ , typically

$$X_{n+1} = \Phi(X_n, U_{n+1}); \tag{1}$$

where X_n is n^{th} observed state of the system, and $\{U_n\}_{n \in \mathbb{Z}}$ the sequence of inputs of the system, typically a sequence of

calls to a Random function. This type of stochastic recursive sequence has been widely studied in a general framework [5] or [7] and some results related with perfect simulation may be found in [12, 13].

It is clear that, if the $\{U_n\}$ are independent and identically distributed, the process $\{X_n\}_{n \in \mathbb{Z}}$ defined by an initial value X_0 and the recursive equations (1) is a Markov chain. Conversely, given a transition matrix P , it is possible to find a transition function Φ such that a Markov chain defined by (1) has transition matrix P [14].

Based on a stochastic recurrent sequence formulation, the following algorithm provides directly a sample of the steady state distribution.

Algorithm 1 Backward-coupling simulation (general version)

```

for all  $x \in \mathcal{X}$  do
   $y(x) \leftarrow x$  {choice of the initial value of the vector  $y$ ,
   $n = 0$ }
end for
repeat
   $u \leftarrow \text{Random}$ ; {generation of  $u_{-n}$ }
  for all  $x \in \mathcal{X}$  do
     $y(x) \leftarrow y(\Phi(x, u))$ ; {computation of the state at
    time 0 of the trajectory issued from  $x$  at time  $-n$ }
  end for
until All  $y(x)$  are equal
return  $y(x)$ 

```

Provided that the operator Φ is monotonous, as shown in section 2, the algorithm could be simplified by making iteration only on maximum and minimum values of the state space. In the open queuing networks situation, there is a unique minimum (all queues are empty) and a unique maximum (all queues are full). Then We only iterates simultaneously 2 trajectories and the time reduction is in the order of the size of the state space. Moreover, it has been shown [11] that the mean coupling time is optimal when steps in the past are multiplied by 2 when trajectories issued from maximum and minimum states have not coupled at time 0. In the algorithm M (resp. m) denotes the set of maximal (resp. minimal) elements in the state space.

4. Examples

All simulations have been done on a 1Ghz Pentium 4 with 512Mo RAM, which is sufficient for our purpose.

4.1. Two queues

Consider the system in figure 1 with capacity 300, rates $\lambda = 1.2$, $\mu = 2$, $\nu = 0.8$ and $p = 0.6$ the routing probability. The estimation of saturation of queue 2 that generates

Algorithm 2 Backward-coupling simulation (monotonous version)

```

 $n=1; R[1]=\text{Random}$ ;
repeat
   $n=2.n$ ;
  for all  $x \in M \cup m$  do
     $y(x) \leftarrow x$ 
  end for
  for  $i=n$  downto  $n/2+1$  do
     $R[i]=\text{Random}$ ;
    for all  $x \in M \cup m$  do
       $y(x) \leftarrow \Phi(y(x), R[i])$ 
    end for
  end for
  for  $i=n/2$  downto  $1$  do
    for all  $x \in M \cup m$  do
       $y(x) \leftarrow \Phi(y(x), R[i])$ 
    end for
  end for
until All  $y(x)$  are equal
return  $y(x)$ 

```

losses is done on sample with size 10^6 . Saturation probability is estimated to $2.27 \cdot 10^{-3} \pm 0.08 \cdot 10^{-3}$ with a 95% confidence interval.

The size of state space of this network is 90601. The mean number of iterations before coupling is about 10^7 simple transitions. The computation time of the generation of one state distributed according steady state is $2ms$.

4.2. Priority servers

The basic model in performance evaluation of network is the Erlang model. It consists in parallel servers, arriving packets are served by the first non empty server. If all servers are busy the packet is rejected. If all servers have

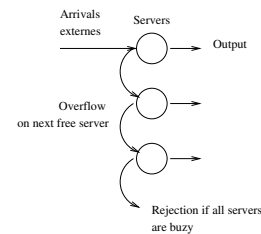


Figure 2. Erlang queueing system (overflow)

the same service rate, it is well known that the system is reversible and the stationary distribution could be analytically computed. This property has been used to test the simulation software. By goodness fitting tests, it has been shown

that the simulated sample is accepted by χ^2 tests as representative of the theoretical distribution.

Moreover, with non homogeneous servers, the system could be simulated easily. An experiment has been done with 30 servers, $\lambda = 20$ and $\mu_i = 1$ for $i \in \{1, \dots, 10\}$, $\mu_i = 0.8$ for $i \in \{11, \dots, 20\}$ and $\mu_i = 0.5$ for $i \in \{21, \dots, 30\}$. Saturation probability is estimated to 0.0579 ± 4.710^{-4} , loss rate is estimated to 1.1 packets per second (the system is loaded).

In that case, the size of the system is $2^{30} \simeq 10^9$ states, the computation time of the generation of one state distributed according to steady state is less than $0.4ms$ and the mean number of iterations is 577 per generated state.

4.3. Blocking line

To observe effect of blocking a line of queues is considered. Queue capacities have been fixed to 100, arrival

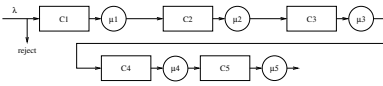


Figure 3. Line of queues with blocking

rate is 1.2 and service rates 0.8. Blocking probabilities for each queue have been estimated : $b_1 = 0.34$, $b_2 = 0.02$, $b_3 = 0.02$, $b_4 = 0.02$. In that case, the size of the system is $100^6 = 10^{12}$ states, the computation time of the generation of one state distributed according to steady state is less than $1ms$.

4.4. Delta networks

To deal with larger networks, a delta network with 4 stages have been implemented. All queues have a capacity of 100 packets. With a load of 0.9 on each input queue and

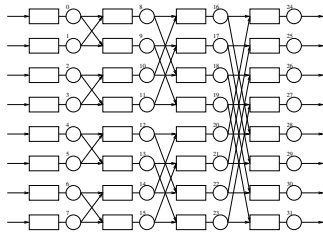


Figure 4. A regular network

a uniform routing at each stage (probability $1/2$), the mean number of iterations to get a state is 400000, the corresponding computation time is $135ms$ per generated state.

It is important to notice that the state space is huge 10^{128} and the presented method remains efficient.

5. Future works

In this article have been established monotonicity properties of Markovian queueing networks with finite capacity queues. These properties are fundamental to improve stochastic simulation of such networks. It appears that the simulation time is short enough to estimate probability of rare events with a sufficient confidence interval. Moreover, perfect simulation of performances indexes could also be done by the method developed by [14].

A free software is under development. Current version is at <http://www-id.imag.fr/Software/PSI2>. Based on an event description of the queueing network, it offers a simulation kernel based on "perfect simulation" algorithms.

Acknowledgement: Author would like to thank Bernard Tanzi for his very efficient development of PSI2 code.

References

- [1] S. Balsamo, V. De Nitto Person, and P. Inverardi. A review of queueing network models with finite capacity queues for software architectures performance prediction. *Performance Evaluation*, 51, 2003.
- [2] S. Balsamo, V. De Nitto Person, and R. Onvural. *Analysis of Queueing Networks with Blocking*. Kluwer Academic Publishers, 2001.
- [3] J. Banks, J. Carson, B. Nelson, and D. Nicol. *Discrete-Event System Simulation*. Prentice-Hall, 2001.
- [4] G. Bolch, S. Greiner, H. de Meer, and K. Trivedi. *Queueing Networks and Markov Chains*. John Wiley & Sons, 1998.
- [5] A. Borovkov and S. Foss. Two ergodicity criteria for stochastically recursive sequences. *Acta Appl. Math.*, 34, 1994.
- [6] P. Brémaud. *Markov Chains: Gibbs fields, Monte Carlo Simulation and Queues*. Springer-Verlag, 1999.
- [7] P. Diaconis and D. Freedman. Iterated random functions. *SIAM Review*, 41(1):45–76, 1999.
- [8] D. Mattson. *On Perfect Simulation of Markovian Queueing Networks with Blocking*. PhD thesis, Chalmers Göteborg University, 2002.
- [9] R. Onvural. Survey of closed queueing networks with blocking. *ACM Computing Surveys*, 22(2):83–121, 1990.
- [10] H. Perros. *Queueing Networks with Blocking Exact and Approximate Solutions*. Oxford University Press, 1994.
- [11] D. Propp, J. and Wilson. Exact sampling with coupled Markov chains and applications to statistical mechanics. *Random Structures and Algorithms*, 9(1&2):223–252, 1996.
- [12] O. Stenflo. *Ergodic theorems for Iterated Function Systems controlled by stochastic sequences*. Doctoral thesis n. 14, Umea university, 1998.
- [13] O. Stenflo. Ergodic theorems for markov chains represented by iterated function systems. *Bull. Polish Acad. Sci. Math*, 2001.
- [14] J.-M. Vincent and C. Marchand. On the exact simulation of functionals of stationary markov chains. *Linear Algebra and its Applications*, 386:285–310, 2004.