Exact sampling for some multi-dimensional queueing models with renewal input

Karl Sigman
Columbia University
New York City
USA

BANFF: Applied Probability Frontiers: Computational and Modeling Challenges
Monday, June 1, 2015

(Joint work with Jose Blanchet and Yanan Pei (Ph.D. student))

Outline

1. We will present some new algorithms for exactly simulating from the stationary distribution of the (stable) FIFO $GI/GI/c$ queue. (Renewal arrivals is the key new model assumption versus only Poisson arrivals.)
2. We will show how to use this algorithm to exactly simulate from the stationary distribution of other multi-server models, such as LIFO, “randomly select the next customer", and even the $GI/GI/\infty$ queue, and Fork-Join Queues.

Background

The first algorithms for exactly simulating from the stationary distribution of the FIFO multi-server queue, are fairly recent and assume Poisson arrivals:

- K. Sigman (2011). Exact simulation of the stationary distribution of the FIFO M/G/c queue. *Journal of Applied Probability*. **Special Volume 48A: New Frontiers in Applied Probability**, 209-216.
- K. Sigman (2011). Exact simulation of the stationary distribution of the FIFO M/G/c queue: The general case of $\rho < c$. *Queueing Systems*. **70**, 37-43.
- S. B. Connor and W. S. Kendall (2015). Perfect simulation of M/G/c queues. *Journal of Applied Probability*, **47**, 4.

Background

The first algorithm assumes $\rho < 1$ (super stability), as opposed to $\rho < c$, uses a backwards in time Dominated Coupling from the Past (DCFP) with a single-server $M/G/1$ queue operating under Processor Sharing(PS) (as upper bound) together with its time-reversibility properties. The algorithm has finite expected termination time if service times have finite variance. (*DCFP methods evolved from Coupling from the Past (CFP) methods as introduced in Propp and Wilson (1996) for finite state discrete-time Markov chains.*)

The second algorithm allows $\rho < c$, and uses a forward-time regenerative process technique using a theorem (that Peter Glynn co-authored!!). This second algorithm used the Random Assignment $M/G/c$ model (RA) as a sample-path upper bound.

Simulating the stationary distr. of a reg. proc. (1)

Suppose that $\{X_n : n \geq 0\}$ is a positive recurrent non-delayed discrete-time regenerative process, with iid cycle lengths generically denoted by T distributed as $F(n) = P(T \leq n),\ n \geq 0,$ with finite and non-zero mean $E(T) = 1/\lambda$. A generic length T cycle is thus $C = \{X_n : 0 \leq n < T\}$. From regenerative process theory, the (marginal) stationary distribution π is given by (expected value over a cycle divided by the expected cycle length)

$$\pi(\cdot) = \lambda E \sum_{n=0}^{T-1} I\{X_n \in \cdot\} = \lambda E \sum_{n=1}^{T} I\{X_n \in \cdot\}. \qquad (1)$$

Simulating the stationary distr. of a reg. proc. (2)

Due to Assmusen, Glynn and Thorisson (1992):

Proposition

1. *Suppose we can and do sequentially simulate iid copies of* $C = \{X_n : 0 \le n < T\}$ *(the first cycle), denoted by* $C_j = \{X_n(j) : 0 \le n < T_j\},\ j \ge 1$*, having iid cycle lengths* $\{T_j\}$ *distributed as* F*.* $E(T) = 1/\lambda$*.*
2. *Suppose further that we can and do simulate (independently) one copy* T^e *having probability mass function* $P(T^e = n) = \lambda P(T \ge n),\ n \ge 1$*. (Stationary excess)*
3. *Let* $\tau = \min\{j \ge 1 : T_j \ge T^e\}$*.*
4. *Use cycle* C_τ *to construct* $X^* = X_{T^e}(\tau)$ *(e.g., if* $T^e = n$ *and* $\tau = j$*, then* $X^* = X_n(j)$*).*

Then the simulated random element X^* *is distributed as* π*.*

Simulating the stationary distr. of a reg. proc. (3)

Unfortunately, $E(\tau) = \infty$ (in Step 3) if T is unbounded (the most common situation).

Outline

The 3rd algorithm (Connor and Kendall (2015)) generalized the first algorithm (DCFP) to allow $\rho < c$ for the M/G/c queue. It used the RA model under PS at each node and the time-reversibility properties, etc.

Background

As for renewal arrivals, the various methods used above break down for various reasons, primarily because while under Poisson arrivals the c stations under RA become independent, they are not independent for general renewal arrivals. Also, the time-reversibility property of PS no longer holds, nor does Poisson Arrivals See Time Averages (PASTA). New methods are needed. Recently, Blanchet and Pei developed a new algorithm that uses a vacation model as an upper bound process for a DCFP approach. In the present paper we will use a different approach that uses the RA model.

Background

Recently, Blanchet and Wallwater developed an algorithm for exactly simulating the maximum M of a negative drift random walk endowed with iid (generally distributed) increments Δ. The only requirement is that one can simulate from the increment distribution $F(x) = P(\Delta \leq x)$, one has an explicit density function $f(x)$ for it and some known bounds such as second moments and (for finite expected termination time): For some (explicitly known) $\epsilon > 0$, $E(|\Delta|^{2+\epsilon}) < \infty$.

Exponential moments are not required (because, only *truncated* exponential moments are needed $E(e^{\gamma\Delta} I\{|\Delta| \leq a\})$, which in turn allow for the simulation of the exponential tilting of truncated Δ, via acceptance-rejection).

Background

One can apply this to exactly simulate from the stationary delay D of a FIFO GI/GI1 queue, defined recursively by the Markov chain

$$D_{n+1} = (D_n + \Delta_n)^+,$$

with $\Delta_n = S_n - T_n$, with iid service times $\{S_n\}$, iid interarrival times $\{T_n\}$.
The random walk is defined by $R_0 = 0$ and

$$R_n = \sum_{j=1}^{n} \Delta_j, \ n \geq 1. \qquad (2)$$

$$D \stackrel{\text{dist}}{=} M = \max_{m \geq 0} R_m.$$

Background

The algorithm allows us to construct a finite segment of a “from the past" stationary version

$$(D^0_{-N}, D^0_{-N+1}, \ldots, D^0_0).$$

N can be a deterministic integer or a stopping time.
Of particular interest is the stopping time

$$N = \min\{n \geq 0 : D^0_{-n} = 0\},$$

and ensuring that $E(N) < \infty$.

Two-sided increments $\{\Delta_j : -\infty < j < +\infty\}$

$$R_n^{(r)} = \sum_{j=1}^{n} \Delta_{-j},\ n \geq 1. \tag{3}$$

A (from the infinite past) stationary version of $\{D_n\}$ denoted by $\{D_n^0 : n \leq 0\}$ can be (theoretically) constructed from Loynes' Lemma

$$\begin{aligned}
D_0^0 &= \max_{m \geq 0} R_m^{(r)} && (4)\\
D_{-1}^0 &= \max_{m \geq 1} R_m^{(r)} - R_1^{(r)} && (5)\\
D_{-2}^0 &= \max_{m \geq 2} R_m^{(r)} - R_2^{(r)} && (6)\\
&\vdots && (7)\\
D_{-n}^0 &= \max_{m \geq n} R_m^{(r)} - R_n^{(r)} && (8)
\end{aligned}$$

Background

The algorithm actually allows the simulation of:

$$\{(R_n^{(r)}, D_n^0) : -N \leq n \leq 0\}$$

for any desired $0 \leq N < \infty$.

Multi-dimensional random walks

In the present paper, we first generalize this result to $c \geq 2$ dimensional random walks, under suitable assumptions, of the kind

$$R_n(i) = \sum_{j=1}^{n} \Delta_j(i), \; n \geq 1. \; (1 \leq i \leq c.) \tag{9}$$

Of particular interest to us (for our applications) is when

$$\Delta_n(i) = S_n(i) - T_n,$$

which means that they each have the same iid T_n.

Multi-dimensional random walks

We are able to simulate

$$\{\{(R_n^{(r)}(i), D_n^0(i)) : -N \leq n \leq 0\}, 1 \leq i \leq c\}.$$

FIFO GI/GI/c Model

It is a c–server in parallel multi-server queue, with only one queue (line). It has iid service times $\{S_n\}$ distributed as $G(x) = P(S \leq x),\ x \geq 0$, with finite and non-zero mean $0 < E(S) = 1/\mu < \infty$. Independently, the arrival times $\{t_n : n \geq 0\}$ ($t_0 = 0$) form a renewal process with iid interarrival times $T_n = t_{n+1} - t_n,\ n \geq 0$, distributed as $A(x) = P(T \leq x),\ x \geq 0$, and finite non-zero arrival rate $0 < \lambda = E(T)^{-1} < \infty$. $\rho \stackrel{\text{def}}{=} \lambda/\mu$.

FIFO GI/GI/c Model

$\mathbf{W}_n = (W_n(1), \ldots, W_n(c))$ denotes the *Kiefer-Wolfowitz workload vector*; it defines a Markov chain and satisfies the recursion

$$\mathbf{W}_{n+1} = R(\mathbf{W}_n + S_n\mathbf{e} - T_n\mathbf{f})^+, \ n \geq 0, \tag{10}$$

where $\mathbf{e} = (1, 0, \ldots 0)$, $\mathbf{f} = (1, 1, \ldots, 1)$, R places a vector in ascending order, and $^+$ takes the positive part of each coordinate. $D_n = W_n(1)$ is then customer delay in queue (line) of the n^{th} customer. With $\rho < c$ (stability), it is well known that $\mathbf{W}_n$ converges in distribution as $n \to \infty$ to a proper stationary distribution. Let π denote this stationary distribution. Our objective is to provide a simulation algorithm for sampling exactly from π.

RA GI/GI/c Model

Each of the c servers forms its own FIFO single-server queue, and each arrival to the system, independent of the past, randomly chooses queue i to join with probability $1/c,\ i \in \{1, 2, \cdots, c\}$

RA GI/GI/c Model upper bound

Lemma

Let $Q_F(t)$ denote total number of customers in system at time $t \geq 0$ for the FIFO GI/GI/c model, and let $Q_{RA}(t)$ denote total number of customers in system at time t for the corresponding RA GI/GI/c model in which both models are initially empty and fed exactly the same input of renewal arrivals $\{t_n\}$ and iid service times $\{S_n\}$. Assume further that for both models the service times are used by the servers in the order in which service initiations occur (S_n is the service time used for the n^{th} such initiation). Then

$$P(Q_F(t) \leq Q_{RA}(t), \text{ for all } t \geq 0) = 1. \tag{11}$$

RA GI/GI/c Model upper bound

In particular, whenever an arrival finds the RA model empty, the FIFO model is found empty as well. (But we need to impose further conditions if we wish to ensure that indeed the RA GI/GI/c will empty with certainty.) Letting time t be sampled at arrival times of customers, $\{t_n : n \geq 0\}$, we thus also have

$$P(Q_F(t_n-) \leq Q_{RA}(t_n-), \text{ for all } n \geq 0) = 1. \tag{12}$$

To use these results it is crucial to simply let the server hand out service times one at a time when they are needed for a service initiation. Thus, customers waiting in a queue before starting service do not have a service time assigned until they enter service. In simulation terminology, this amounts to generating the service times in order of when they are needed.

RA GI/GI/c Model upper bound

But doing that does not allow us to define total workload; so we can instead generate service times upon arrival of customers, and give them to the server to be used in order of service initiation. If we let $\{V_F(t) : t \geq 0\}$ and $\{V_{RA}(t) : t \geq 0\}$ denote total work in the two models with the service times used in the manner just explained, then in addition to Lemma 1 we have

$$P(V_F(t) \leq V_{RA}(t), \text{ for all } t \geq 0) = 1. \tag{13}$$

$$P(V_F(t_n-) \leq V_{RA}(t_n-), \text{ for all } n \geq 0) = 1. \tag{14}$$

Recursion for the RA GI/GI/c

Let

$$\mathbf{V}_n = (V_n(1), \ldots, V_n(c))$$

denote workload (at each node) as found by the n^{th} arriving customer to the RA model, and let $S_n(i) = S_n I\{U_n = i\}$, where independently $\{U_n : n \geq 0\}$ denotes an iid sequence of random variables with the discrete uniform distribution over $\{1, 2, \ldots, c\}$. Then for each node $i \in \{1, 2, \ldots, c\}$,

$$V_{n+1}(i) = (V_n(i) + S_n(i) - T_n)^+, \ n \geq 0. \tag{15}$$

Because of all the iid assumptions, $\{\mathbf{V}_n\}$ forms a Markov chain. By defining $\mathbf{S}_n = (\mathbf{S}_n(1), ..., \mathbf{S}_n(c))$, and $\mathbf{T}_n = (T_n, T_n, \ldots, T_n)$ (length c vector) we can express (15) in vector form :

$$\mathbf{V}_{n+1} = (\mathbf{V}_n + \mathbf{S}_n - \mathbf{T}_n)^+, \ n \geq 0. \tag{16}$$

How we will couple service times

Lemma

Let $\{S'_n\}$ be an iid sequence of service times distributed as G, and assign S'_n to C_n in the RA model. Define S_n as the service time used in the n^{th} service initiation. Then $\{S_n\}$ is also iid distributed as G.

How we will couple service times

The point of the above Lemma is that we can, if we so wish, simulate the RA model by assigning S'_n to C_n (to be used as their service time), but then assigning S_n to C_n in the FIFO model. By doing so the sample-path upper bounds hold.

Interestingly, however, it is not possible to first simulate the RA model up to a fixed time t, and then stop and reconstruct the FIFO model up to this time t: At time t, there may still be RA customers waiting in lines and hence not enough of the S_n have been determined yet to construct the FIFO model. But all we have to do, if need be, is to continue the simulation of the RA model beyond t until enough S_n have been determined to construct fully the FIFO model up to t.

Setting up the c random walks

Defining iid "increments" $\Delta_n(i) = S_n(i) - T_n,\ n \geq 0$, each node i has an associated negative drift random walk $\{R_n(i) : n \geq 0\}$ with $R_0(i) = 0$ and

$$R_n(i) = \sum_{j=1}^{n} \Delta_j(i),\ n \geq 1. \tag{17}$$

$$E(\Delta(i)) = \frac{1}{c} E(S) - E(T) < 0.$$

Setting up the time-reversed c random walks

$$R_n^{(r)}(i) = \sum_{j=1}^{n} \Delta_{-j}(i), \ n \geq 1. \ (1 \leq i \leq c.) \tag{18}$$

A (from the infinite past) stationary version of $\{V_n(i)\}$ denoted by $\{V_n^0(i) : n \leq 0\}$ is then constructed via

$$V_0^0(i) = \max_{m \geq 0} R_m^{(r)}(i) \tag{19}$$

$$V_{-1}^0(i) = \max_{m \geq 1} R_m^{(r)}(i) - R_1^{(r)}(i) \tag{20}$$

$$V_{-2}^0(i) = \max_{m \geq 2} R_m^{(r)}(i) - R_2^{(r)}(i) \tag{21}$$

$$\vdots \tag{22}$$

$$V_{-n}^0(i) = \max_{m \geq n} R_m^{(r)}(i) - R_n^{(r)}(i) \tag{23}$$

Setting up the time-reversed c random walks

$\mathbf{V}_n^0 = (V_n^0(1), \ldots, V_n^0(c)),\ n \leq 0$, is jointly stationary representing a (from the infinite past) stationary version of $\{\mathbf{V}_n\}$, and satisfies the forward-time recursion (16):

$$\mathbf{V}_{n+1}^0 = (\mathbf{V}_n^0 + \mathbf{S}_n - \mathbf{T}_n)^+,\ n \leq 0. \tag{24}$$

Thus, by starting at $n = 0$ and walking backwards in time, we have (theoretically) a time-reversed copy of the RA model. Furthermore, $\{\mathbf{V}_n^0\}$ can be extended to include forward time $n \geq 1$ via using the recursion further:

$$\mathbf{V}_n^0 = (\mathbf{V}_{n-1}^0 + \mathbf{S}_{n-1} - \mathbf{T}_{n-1})^+,\ n \geq 1. \tag{25}$$

In fact once we have a copy of just $\mathbf{V}_0^0$, we can start off the Markov chain with it as initial condition and use (25) to obtain a forward in time stationary version $\{\mathbf{V}_n^0 : n \geq 0\}$.

Algorithm when $P(T > S) > 0$

If $P(T > S) > 0$, the stable ($\rho < c$) RA and FIFO Markov chains will visit 0 infinitely often with certainty; otherwise they will not. The most common sufficient conditions are that T has unbounded support, $P(T > t) > 0,\ t \geq 0$, or S has mass arbitrarily close to 0, $P(S < t) > 0,\ t > 0$.

Since $Q_{RA}(t_n-) = 0$ if and only if $\mathbf{V}_n = \mathbf{0}$, we conclude that if at any time n it holds that $\mathbf{V}_n = \mathbf{0}$, then $\mathbf{W}_n = \mathbf{0}$. By the Markov property, given that $\mathbf{V}_n = \mathbf{0} = \mathbf{W}_n$, the future is independent of the past for each model, or said differently, *the past is independent of the future*. This remains valid if n is replaced by a stopping time (strong Markov property).

(We show that without loss of generality, one can assume that interrival times are bounded. And later in this talk we will modify our algorithm to handle the case when $P(T > S) = 0$.)

Algorithm when $P(T > S) > 0$

1. Simulate $\{\{(R_n^{(r)}(i), V_n^0(i)) : -N \leq n \leq 0\}, 1 \leq i \leq c\}$, where $N = \min\{n \geq 0 : \mathbf{V}_{-n}^0 = \mathbf{0}\}$. If $N = 0$ stop. Otherwise, having stored all data, reconstruct $\mathbf{V}_n^0$ forwards in time from $n = -N$ (initially empty) until $n = 0$, using the recursion (24). During this forward-time reconstruction, re-define S_j as the j^{th} service initiation used by the RA model (e.g., we are using Lemma 2 to gather service times in the proper order to feed to the FIFO model, that is why we do the re-construction).

Algorithm when $P(T > S) > 0$

2. If at time $n = 0$ there have not yet been N service initiations, then continue simulating the RA model out in forward time until finally there is a N^{th} service initiation, and then stop. This will require, at most, simulating out to $n = N^{(+)} = \min\{n \geq 0 : \mathbf{V}_n^0 = \mathbf{0}\}$. Take the vector $(S_{-N}, S_{-N+1}, \ldots, S_{-1})$ and reset $(S_0, S_1, \ldots, S_{N-1}) = (S_{-N}, S_{-N+1}, \ldots, S_{-1})$. Also, store the interarrival times $(T_{-N}, T_{-N+1}, \ldots, T_{-1})$, and reset $(T_0, \ldots, T_{N-1}) = (T_{-N}, T_{-N+1}, \ldots, T_{-1})$.

Algorithm when $P(T > S) > 0$

3. If $N = 0$, then set $\mathbf{W}_0 = \mathbf{0}$ and stop. Otherwise use the Kiefer-Wolfowitz recursion

$$\mathbf{W}_{n+1} = R(\mathbf{W}_n + S_n\mathbf{e} - T_n\mathbf{f})^+, \; n \geq 0, \tag{26}$$

with $\mathbf{W}_0 = \mathbf{0}$, recursively forwards in time N times until obtaining $\mathbf{W}_N$, by using the N re-set service and interarrival times $(S_0, S_1, \ldots, S_{N-1})$ and $(T_0, \ldots, T_{N-1})$. Reset $\mathbf{W}_0 = \mathbf{W}_N$ and stop.

4. Output $\mathbf{W}_0$

Algorithm when $P(T > S) = 0$: Harris reccurent regeneration

For illustration here, let us consider $c = 2$ servers and let assume that $1 < \rho < 2$. (Note that if $\rho < 1$, then equivalently $E(T) > E(S)$ and so $P(T > S) > 0$; that is why we rule out $\rho < 1$ here.) If $P(T > S) = 0$, then for $\underline{s} \stackrel{\text{def}}{=} \inf\{s > 0 : P(S > s) > 0\}$ and $\bar{t} \stackrel{\text{def}}{=} \sup\{t > 0 : P(T > t) > 0\}$, we must have $0 < \bar{t} < \underline{s} < \infty$.

Algorithm when $P(T > S) = 0$: Harris reccurent regeneration

It can be shown (1988) that for $\epsilon > 0$ sufficiently small, the following event will happen infinitely often (in n) with probability 1

$$\{Q_{RA}(t_n-) = 1,\ V_n(1) = 0,\ V_n(2) \leq \epsilon,\ T_n > \epsilon,\ U_n = 1\}. \tag{27}$$

Algorithm when $P(T > S) = 0$: Harris reccurent regeneration

If n is such a time, then at time $n + 1$, we have

$$\{Q_{RA}(t_{n+1}-) = 1,\ V_{n+1}(2) = 0,\ V_{n+1}(1) = (S_n - T_n) \mid T_n > \epsilon\}. \quad (28)$$

Algorithm when $P(T > S) = 0$: Harris reccurent regeneration

The point is that C_n finds one server (server 1) empty, and the other queue with only one customer in it, and that customer has a remaining service time $\leq \epsilon$. C_n then enters service at node 1 with service time S_n; but since $T_n > \epsilon$, C_{n+1} arrives finding the second queue empty, and the first server has remaining service time $S_n - T_n$ conditional on $T_n > \epsilon$.

Algorithm when $P(T > S) = 0$: Harris reccurent regeneration

Under the coupling of Lemma 1, the same will be so for the FIFO model (see Remark 1 below): At such a time n,

$$\{Q_F(t_n-) = 1,\ W_n(1) = 0,\ W_n(2) \leq \epsilon,\ T_n > \epsilon\} \tag{29}$$

and at time $n + 1$ we have

$$\{Q_F(t_{n+1}-) = 1,\ W_n(1) = 0,\ W_n(2) = (S_n - T_n) \mid T_n > \epsilon\}. \tag{30}$$

(28) and (30) define positive recurrent regeneration points for the two models (at time $n + 1$); the consecutive times at which regenerations occur forms a (discrete-time) positive recurrent renewal process.

Algorithm when $P(T > S) = 0$: Harris reccurent regeneration

To put this to use, we change the stopping time N used in our earlier algorithm to:

$$N + 1 = \min\{n \geq 1 : Q^0_{RA}(t_{-(n+1)}-) = 1,\ V^0_{-(n+1)}(1) = 0, \tag{31}$$

$$V^0_{-(n+1)}(2) \leq \epsilon,\ T_{-(n+1)} > \epsilon,\ U_{-(n+1)} = 1\}. \tag{32}$$

Then we do our reconstructions for the previous algorithm by starting at time $-N$, with both models starting with the same starting value

$$\{Q_{RA}(t_{-N}-) = 1,\ V^0_{-N}(2) = 0,\ V^0_{-N}(1) = (S_{-(N+1)} - T_{-(N+1)}) \mid T_{-(N+1)} > \epsilon\} \tag{33}$$

$$\{Q_F(t_{-N}-) = 1,\ W_{-N}(1) = 0,\ W_{-N}(2) = (S_{-(N+1)} - T_{-(N+1)}) \mid T_{-(N+1)} > \epsilon\}. \tag{34}$$

Algorithm when $P(T > S) = 0$: Harris reccurent regeneration

Remark

The service time used in (33) and (34) for coupling via Lemma 2, $S_{-(N+1)}$, is in fact identical for both systems because (subtle): At time $-(N+1)$, both systems have only one customer in system, and thus total work is in fact equal to the remaining service time; so we use Equation (14) to conclude that both remaining service times (even if different) are $\leq \epsilon$ (e.g., that is why (29) follows from (27)). Meanwhile, $C_{-(N+1)}$ enters service immediately across both systems, so it is indeed the same service time $S_{-(N+1)}$ used for both for this initiation.

Other models that can exactly be simulated using these results

1. $GI/GI/\infty$ queue: Find the smallest integer c such that $\rho < c$. Then workload and number in system for the corresponding stable FIFO $GI/GI/c$ queue serves as an upper bound for the $GI/GI/\infty$.
2. Fork-Join queues (FJ): The recursion for the RA model is a special case of that for FJ; $\mathbf{S}_n = (\mathbf{S}_n(1), ..., \mathbf{S}_n(c))$ general iid vectors;

$$\mathbf{V}_{n+1} = (\mathbf{V}_n + \mathbf{S}_n - \mathbf{T}_n)^+, \; n \geq 0. \tag{35}$$

   The sojourn time of the i^{th} component is given by $V_n(i) + S_n(i)$, and thus the sojourn time of the n^{th} job, C_n, is given by

$$H_n = \max_{1 \leq i \leq c} \{V_n(i) + S_n(i)\}. \tag{36}$$

$$H^0 \overset{\text{dist}}{=} \max_{1 \leq i \leq c} \{V_0^0(i) + S(i)\}. \tag{37}$$

Other models that can exactly be simulated using these results

3. $GI/GI/c$ queues operating under LIFO (non-preemptive), or under “select randomly from the queue (line)" (RS) the next customer to enter service. In both cases, if service times are handed out by the server, then the stationary distribution of number in system, and the remaining service times of those in service (joint process) is identical to that of FIFO. So: At $t = 0$ start off with a stationary copy of FIFO. Tag a customer C_0 and simulate under LIFO (or RS) via discrete simulation until C_0 enters service. That length of time is stationary delay.

References

S. Asmussen (2003). *Applied Probability and Queues* (2nd Ed.). Springer-Verlag, New York.

S. Asmussen, P. W. Glynn (2007). *Stochastic Simulation*, Springer-Verlag, New York.

S. Asmussen, P. W. Glynn, & H. Thorisson (1992). Stationary detection in the initial transient problem. *ACM TOMACS*, **2**, 130-157.

J. Blanchet and A. Wallwater (2015). Exact sampling of stationary and time-reversed queues.

S. B. Connor and W. S. Kendall (2015). Perfect simulation of M/G/c queues. *Journal of Applied Probability*, **47**, 4.

Hongcheng Dai (2011). Exact Monte Carlo simulation for fork-join networks. *Advances in Applied Prob.*, **43**, 484-503.

Flatto, L., and Hahn, S. (1984). Two parallel queues created by arrivals with two demands. *SIAM J. Appl. Math.*, **44**, 1041-1053.

S. G. Foss (1980). Approximation of mutichannel queueing systems. *Siberian Math. J.*, **21**, 132-40.

S. G. Foss and N. I. Chernova (2001). On optimality of the FCFS discipline in mutiserver queueing systems and networks. *Siberian Math. J.*, **42**, 372-385.

J. G. Propp and D.G. Wilson (1996) Exact sampling with coupled Markov chains and applications to statistical mechanics. *Random Structures and Algorithms*, **9**, 223-253.

References

K. Sigman (2011). Exact simulation of the stationary distribution of the FIFO M/G/c queue. *Journal of Applied Probability*. **Special Volume 48A: New Frontiers in Applied Probability**, 209-216.

K. Sigman (2011). Exact simulation of the stationary distribution of the FIFO M/G/c queue: The general case of $\rho < c$. *Queueing Systems*. **70**, 37-43.

K. Sigman (1995). *Stationary Marked Point Processes: An Intuitive Approach*, Chapman and Hall, New York.

K. Sigman (1988). Regeneration in Tandem Queues with Multi-Server Stations. *Journal of Applied Probability*, **25**, 391-403.

R.W. Wolff (1987). Upper bounds on work in system for multi-channel queues. *J. Appl. Probab.* **14**, 547-551.

R.W. Wolff (1982). Poisson Arrivals See Time Averages. *Operations Research*. **30**, 223-231.

Peter and his Latex/Computer Use

http://www.smashinglists.com/15-unforgettable-mac-computers-old-is-gold/

http://upload.wikimedia.org/wikipedia/commons/0/08/Apple_Laserwriter_II.jpg

http://www.old-computers.com/museum/photos/Kaypro_Desktop_System_s1.jpg

Peter and politics

EDWIN MEESE III STORY (Ronald Reagan Distinguished Fellow Emeritus at The Heritage Foundation) Attorney General under Ronald Reagan, and was involved in shredding evidence in the Iran Contra scandal (Illegally selling arms to Iran and using the profits to arm the Nicaraguan Contras)
"This is Attorney General Ed Meese. I have me surrounded. If I don't come out with my hands up, I'm coming in after me".
(1) Almost knocked him over with my bicycle
(2) Telling Pete the story and he says, "Why did you turn on your breaks?"
(3) Then, at the Faculty house, where Pete kindly treated me to a fine meal, Pete suddenly says, "Speaking of the Devil!!". Ed Meese was sitting at the table next to us!