Gerald S. Shedler
Jonathan Southard

Regenerative Simulation of Networks of Queues with General Service Times: Passage Through Subnetworks

A linear "job stack," an enumeration by service center and job class of all the jobs, is an appropriate state vector for simulation of closed networks of queues with priorities among job classes. Using a representation of the job stack process as an irreducible generalized semi-Markov process, we develop a regenerative simulation method for passage times in networks with general service times. Our estimation procedure avoids Cox-phase representation of general service time distributions and is applicable to networks with "single states" for passage times. Based on a single simulation run, the procedure provides point estimates and confidence intervals for characteristics of limiting passage times.

1. Introduction

Simulation is usually the only method available for studying passage times in closed, multiclass networks of queues with priorities among job classes. Informally, a passage time is the time for a job to traverse a portion of the network. Such quantities are important in computer and communication system models where they represent job "response times." For networks with Cox-phase service times, Iglehart and Shedler [1, Sec. 8] and Shedler and Slutz [2] have shown that point estimates and confidence intervals for characteristics of passage times through a subnetwork can be obtained by observing a single sample path of an irreducible Markov chain. The states of this Markov chain are configurations of a "fully augmented linear job stack." Shedler and Southard [3] extended the applicability of this estimation method to networks with unrestricted priorities; for such networks the underlying Markov chain need not be irreducible. In this paper we derive an analogous estimation procedure (termed the "labeled jobs method") for networks with general service times having density functions that are continuous and positive on $(0, \infty)$. In our procedure, passage times for all the jobs enter into the construction of point and interval estimates. The related procedure of Iglehart and Shedler [4] is based on observations of passage times for a single job.

An appropriate state vector for simulation of closed, multiclass networks of queues is a linear "job stack," an enumeration of the jobs by service center and job class. Passage times are recorded by observing a "fully augmented job stack process," which maintains the position of each of the jobs in the job stack. We show under a mild restriction on the priorities among job classes that the job stack process observed at the epochs at which passage times terminate is a regenerative process in discrete time. As a consequence, point and interval estimates for characteristics of limiting passage times can be obtained from a single simulation run. To establish these results, we use a representation of the job stack process as a generalized semi-Markov process in the sense of Whitt [5].

We show that terminations of passage times with no other passage times underway and exactly one job in service are regeneration points for the job stack process observed at termination times. In order for such epochs to exist we must exclude passage times which are complete circuits as well as passage times which always terminate with two or more jobs in service. A mild restriction on the priorities among job classes ensures that infinitely many such epochs occur.

Section 2 contains a description of the networks under consideration, a definition of the job stack, and the formal specification of passage times. The generalized semi-Markov process representation for the job stack process and stochastic process results underlying the estimates for passage times are in Section 3. The estimation procedure for passage times is in Section 4.

2. Closed, multiclass networks of queues and passage times

As in [1], we consider closed networks of queues having a finite number of *jobs* (customers), N, a finite number of (single or multiple server) *service centers*, s, and a finite number of (mutually exclusive) *job classes*, c. At every epoch of continuous time each job is in exactly one job class, but jobs may change class as they traverse the network. Upon completion of service at center i a job of class j goes to center k and changes to class l with probability $p_{ij,kl}$, where $\mathbf{P} = \{p_{ij,kl} : (i, j), (k, l) \in C\}$ is a given irreducible Markov matrix and $C \subseteq \{1, 2, \cdots, s\} \times \{1, 2, \cdots, c\}$ is the set of (center, class) pairs in the network. At each service center jobs queue and receive service according to a fixed priority scheme among classes; the priority scheme may differ from center to center. Within a class at a center, jobs receive service according to a fixed queue service discipline, *e.g.*, first-come, first-served (FCFS). Note that in accordance with the matrix $\mathbf{P}$, some centers may never see jobs of certain classes. According to a fixed procedure for each center, a job in service may or may not be preempted if another job of higher priority joins the queue at the center. For expository convenience we assume that any interruption of service is of the preemptive-repeat type. A job that has been preempted receives additional service at the center before any other job of its class at the center receives service.

All service times are assumed to be mutually independent. We also suppose that service times at a center have finite mean but otherwise arbitrary density function which is continuous and positive on $(0, \infty)$. Parameters of the service time distribution may depend on the service center, the class of job in service, and the "state" [as defined in Eq. (1) below] of the entire network at the time service begins. In order to characterize the state of the network at time t, we let $S_i(t)$ denote the class of the job in service at center i at time t, where $i = 1, 2, \cdots, s$; by convention $S_i(t) = 0$ if at time t there are no jobs at center i. [If center i has more than one server, we let $S_i(t)$ be a vector identifying the class of the job receiving service from each server at center i.] The classes of jobs serviced at center i ordered by decreasing priority are $j_1(i), j_2(i), \cdots, j_{k(i)}(i)$, elements of the set $\{1, 2, \cdots, c\}$. We denote by $C^{(i)}_{j_1}(t), \cdots, C^{(i)}_{j_{k(i)}}(t)$ the number of jobs in queue at time t of the various classes of jobs serviced at center i, $i = 1, 2, \cdots, s$.

We think of the N jobs being ordered in a linear stack according to the following scheme. For $t \geq 0$ define the state vector $Z(t)$ at time t by

$$Z(t) = (C^{(1)}_{j_{k(1)}}(t), \cdots, C^{(1)}_{j_1}(t), S_1(t); \cdots; C^{(s)}_{j_{k(s)}}(t), \cdots, C^{(s)}_{j_1(t)}, S_s(t)). \tag{1}$$

The *job stack at time t* then corresponds to the order of components in the vector $Z(t)$ after ignoring any zero components. Within a class at a particular service center, jobs waiting appear in the job stack in FCFS order, *i.e.*, in order of their arrival at the center, the latest to arrive being closest to the top of the stack. The *job stack process* $\mathbf{Z} = \{Z(t) : t \geq 0\}$ has a finite state space, D^*. For any service center i that sees only one class of job, *i.e.*, such that $k(i) = 1$, it is possible to simplify the state vector by replacing $C^{(i)}_{j_{k(i)}}(t), S_i(t)$ by $Q_i(t)$, the total number of jobs at center i. Note that the state vector definition does not take into account explicitly that the total number of jobs in the network is fixed. In the case of complex networks, the use of this resulting somewhat larger state space facilitates generation of the state vector process; for relatively simple networks, it may be desirable to remove the redundancy.

- *Definition of passage times*

Arbitrarily choose one of the jobs and refer to this distinguished job as the "marked job." For $t \geq 0$ denote by $N(t)$ the position (from the top) of the marked job in the job stack at time t. Then set

$$X(t) = (Z(t), N(t)) \tag{2}$$

and call $\mathbf{X} = \{X(t) : t \geq 0\}$ the *augmented job stack process*. Passage times are specified in terms of the marked job by means of four subsets (A_1, A_2, B_1, and B_2) of the state space, E^*, of the stochastic process $\mathbf{X}$. The sets A_1, A_2 [resp. B_1, B_2] jointly define the random times at which passage times for the marked job start [resp. terminate]. The sets A_1, A_2, B_1, and B_2 in effect determine when to start and stop the clock measuring a particular passage time of the marked job.

Denote the jump times of the process $\mathbf{X}$ by $\{\tau_n : n \geq 0\}$. For $k, n \geq 1$, we require that the sets A_1, A_2, B_1, and B_2 satisfy the following:

if $X(\tau_{n-1}) \in A_1, X(\tau_n) \in A_2, X(\tau_{n-1+k}) \in A_1$ and $X(\tau_{n+k}) \in A_2$,

then $X(\tau_{n-1+m}) \in B_1$ and $X(\tau_{n+m}) \in B_2$ for some $0 < m \leq k$;

and

if $X(\tau_{n-1}) \in B_1, X(\tau_n) \in B_2, X(\tau_{n-1+k}) \in B_1$ and $X(\tau_{n+k}) \in B_2$,

then $X(\tau_{n-1+m}) \in A_1$ and $X(\tau_{n+m}) \in A_2$ for some $0 \leq m < k$.

These conditions ensure that the start and termination times for the specified passage time strictly alternate.

In terms of the sets A_1, A_2, B_1, and B_2, we define two sequences of random times, $\{S_j : j \geq 0\}$ and $\{T_j : j \geq 1\}$, where

S_{j-1} is the start time of the jth passage time for the marked job and T_j is the termination time of this jth passage time. Assuming that the initial state of the process $\mathbf{X}$ is such that a passage time for the marked job begins at $t = 0$, let

$$S_0 = 0,$$

$$S_j = \inf\{\tau_n \geq T_j : X(\tau_n) \in A_2, X(\tau_{n-1}) \in A_1\}, \; j \geq 1,$$

and

$$T_j = \inf\{\tau_n \geq S_{j-1} : X(\tau_n) \in B_2, X(\tau_{n-1}) \in B_1\}, \; j \geq 1.$$

Then the jth passage time for the marked job is $P_j = T_j - S_{j-1}$, $j \geq 1$.

Example 1 Consider a network with two service centers and a single job class such that the set C of (center, class) pairs is $C = \{(1, 1), (2, 1)\}$. Suppose that a job completing service at center 1 joins the tail of the queue at center 1 with probability p and (with probability $1 - p$) joins the tail of the queue at center 2. A job completing service at center 2 joins the tail of the queue at center 1. For $t \geq 0$ let $Z(t)$ be the number of jobs waiting or in service at center 1 at time t and set $X(t) = (Z(t), N(t))$, where $N(t)$ is the position of the marked job in the job stack at time t. The state space of the augmented job stack process $\mathbf{X} = \{X(t) : t \geq 0\}$ is

$$E = \{(i, j) : 0 \leq i \leq N, 1 \leq j \leq N\}.$$

Thus, for example, if all N jobs are at center 2 and the marked job is in service, the augmented job stack process is in state $(0, N)$. Upon completion of service at center 2, the marked job goes into service at center 1, the remaining $N - 1$ jobs are at center 2, and the process is in state $(1, 1)$.

Consider the passage time which starts when a job completes service at center 2 (and joins the tail of the queue at center 1) and terminates when the job next joins the tail of the queue at center 2. This passage time is specified by the sets

$$A_1 = \{(i, N) : 0 \leq i < N\},$$

$$A_2 = \{(i, 1) : 0 < i \leq N\},$$

$$B_1 = \{(i, i) : 0 < i \leq N\},$$

and

$$B_2 = \{(i - 1, i) : 0 < i \leq N\}.$$

For $N = 2$ jobs, these sets are shown in Fig. 1.

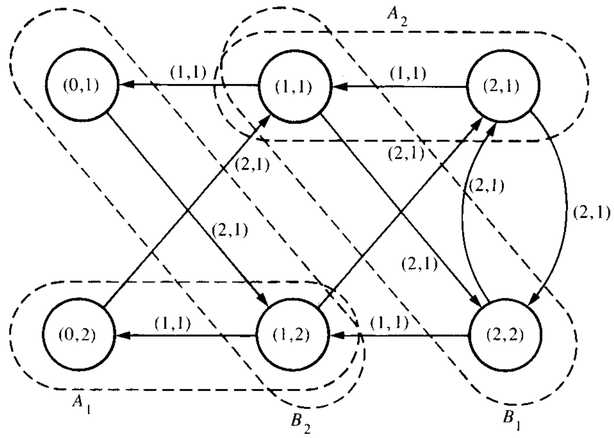

Figure 1 Subsets of E for passage time P.

• *Restriction of the job stack process*

To obtain recurrence results it is necessary to restrict the job stack process to a suitably chosen subset of its state space. Let $U(z)$ be the set of all (center, class) pairs $(i, j) \in C$ such that in state $z \in D^*$ there is a job of class j in service at center i. For $z, z' \in D^*$ and $u = (i, j) \in U(z)$, let $q(z'; z, u)$ be the probability that the job stack process $\mathbf{Z}$ jumps (in one step) to state z', given that in state z there is a completion of service to a job of class j at center i. When $q(z'; z, u) > 0$ for some $u \in U(z)$, we write $z \rightarrow z'$. For all $z, z' \in D^*$, we say that z' *is accessible from* z and write $z \rightsquigarrow z'$ if there exists a finite sequence $u'_0, z_1, u'_1, \cdots, u'_n$ of (center, class) pairs and job stacks such that

$$q(z_1; z, u'_0)q(z_2; z_1; u'_1) \cdots q(z'; z_n, u'_n) > 0. \tag{3}$$

When $z \rightsquigarrow z'$ and $z' \rightsquigarrow z$, we say that z *and* z' *communicate* and we write $z \leftrightsquigarrow z'$.

We also define $U(x)$ for $x \in E^* : U(x) = U(z)$ when $x = (z, n)$ for some $z \in D^*$ and $n \in \{1, 2, \cdots, N\}$. For $x, x' \in E^*$ and $u = (i, j) \in U(x)$, we denote by $p(x'; x, u)$ the probability that the augmented job stack process $\mathbf{X}$ jumps to state x', given that in state x there is a completion of service to a job of class j at center i. We write $x \rightarrow x'$ when $p(x'; x, u) > 0$ for some $u \in U(x)$. We say that x' *is accessible from* x and write $x \rightsquigarrow x'$ if there exists a finite sequence $u'_0, x_1, u'_1, \cdots, u'_n$ of (center, class) pairs and augmented job stacks such that

$$p(x_1; x, u'_0)\, p(x_2; x_1, u'_1) \cdots p(x'; x_n, u'_n) > 0. \tag{4}$$

When $x \rightsquigarrow x'$ and $x' \rightsquigarrow x$ we say that x *and* x' *communicate* and we write $x \leftrightsquigarrow x'$.

Recurrence results can be obtained by restricting the job stack process to a set D within which all states communicate. We make the further assumption that for some $z^* \in D^*$ the set

$$D = \{z \in D^* : z^* \rightsquigarrow z\}$$

is irreducible, *i.e.*, $z \rightsquigarrow z'$ for all $z, z' \in D^*$. It is sufficient that for some service center, i_0, either $k(i_0) = 1$ or service at

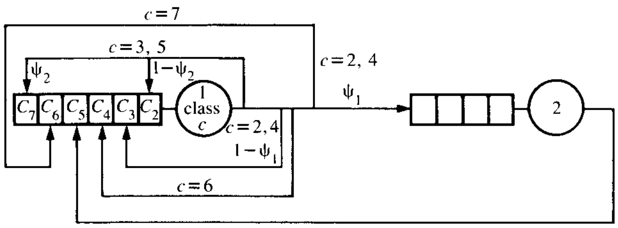

Figure 2 Closed multiclass network of queues.

center i_0 to a job of class $j_{k(i_0)}(i_0)$ (the lowest priority job class seen by center i_0) is preempted when a job of higher priority joins the queue. Let $z^*_{i_0} \in D^*$ be the job stack in which there is one job of class $j_{k(i_0)}(i_0)$ in service at center i_0 and $N-1$ jobs of class $j_{k(i_0)}(i_0)$ in queue at center i_0. Define D to be the set of all states of the job stack process $\mathbf{Z}$ that are accessible from $z^*_{i_0}$, *i.e.*,

$$D = \{z \in D^* : z^*_{i_0} \rightsquigarrow z\}. \tag{5}$$

Then define a subset E of the state space E^* of the augmented job stack process $\mathbf{X}$ according to

$$E = \{(z, n) \in E^* : z \in D\}. \tag{6}$$

We assume that the sets A_1, A_2, B_1, and B_2 which define the starts and terminations of passage times are subsets of $E = \{(z, n) : z \in D, 1 \le n \le N\}$.

Proposition (2) provides conditions which ensure that all states of the job stack process $\mathbf{Z}$ restricted to the set D communicate and that all states of the augmented job stack process $\mathbf{X}$ restricted to the set E communicate. The argument used to establish the result is similar to the proof of Proposition (3.4) in Shedler and Slutz [6].

Proposition 2 Let the number of service centers $s > 1$. Suppose that the routing matrix $\mathbf{P}$ is irreducible and that for some service center i_0 either $k(i_0) = 1$ or service to a job of class $j_{k(i_0)}(i_0)$ at center i_0 is preempted when a job of higher priority joins the queue. Then $z \leftrightsquigarrow z'$ for all $z, z' \in D$ and $x \leftrightsquigarrow x'$ for all $x, x' \in E$.

Example 3 illustrates these ideas for a model proposed for processor scheduling in data base management systems; see Lavenberg and Shedler [7] and Iglehart and Shedler [1, Sec. 5.2].

Example 3 Consider the network shown in Fig. 2. There are two service centers and seven job classes, and the set C of (center, class) pairs is

$$C = \{(1, 2), (1, 3), (1, 4), (1, 5), (1, 6), (1, 7), (2, 1)\}.$$

Suppose that the classes of jobs serviced at center 1 ordered by decreasing priority are $j_1(1) = 2$, $j_2(1) = 3, \cdots,$ $j_6(1) = 7$. Service to jobs of class 7 (at center 1) is subject to preemption when a job of higher priority joins the queue at center 1. Service to any other job class is not subject to preemption. Also suppose that (for $0 < \psi_1, \psi_2 < 1$) the (irreducible) routing matrix $\mathbf{P}$ is as shown in Fig. 3. For $t \ge 0$ let

$$Z(t) = (C_7^{(1)}(t), \cdots, C_2^{(1)}(t), S_1(t), Q_2(t)),$$

where the number of jobs of class j in queue at center 1 at time t is $C_j^{(1)}(t)$, $S_1(t)$ is the class of the job in service at center 1, and $Q_2(t)$ is the number of jobs waiting or in service at center 2. As $k(2) = 1$ and service to jobs of class $j_{k(1)}(1) = 7$ at center 1 is subject to preemption, either state

$$z_1^* = (N-1, 0, 0, 0, 0, 0, 7, 0)$$

or state

$$z_2^* = (0, 0, 0, 0, 0, 0, 0, N)$$

can serve as the state $z^*_{i_0}$ which defines the set D. For $N \ge 2$ jobs the set $D^* - D \ne \varnothing$ is nonempty; *e.g.*, the state $(0, 0, 0, k-1, 0, 0, 4, N-k)$ is an element of $D^* - D$ provided that $k \ge 2$.

- *Single states of the job stack process*

We now identify special "single states" of the job stack process which are used explicitly in the labeled jobs method of Section 4. A single state of the job stack process is a configuration of the job stack such that no passage times are underway, all jobs are at the same center with exactly one job in service, and entrances to the state (from a fixed state) correspond to the terminations of passage times. Our notion of single state coincides with that of Fossett [8] for generalized semi-Markov processes.

Formally, define two sets S and T according to

$$S = \{(k, m) : k \in A_1, m \in A_2, \text{ and } k \rightarrow m\}$$

and

$$T = \{(k, m) : k \in B_1, m \in B_2, \text{ and } k \rightarrow m\}.$$

Let H be the set of all (center, class) pairs (i, j) such that a marked job can be of class j at center i when the passage time specified by the sets A_1, A_2, B_1, and B_2 terminates or is not underway. The labeled jobs method applies to passage times through a subnetwork, *i.e.*, to passage times for which $S \cap T = \varnothing$ so that the set H is nonempty.

Also define a function h taking values in C and having domain $D \times \{1, 2, \cdots, N\}$ such that the value of $h(z, n)$ is (i, j) when the job in position n of the job stack associated with $z \in D$ is of class j at center i.

Definition 4 An element z of the set D is called a *single state of the job stack process* for the passage time specified by the sets A_1, A_2, B_1, and B_2 if

1. $h(z, n) = (i_0, j_{l_n}(i_0)) \in H$ for some single server center i_0, $n = 1, 2, \cdots, N$; and
2. there exists $(z_1, m) \in B_1$ such that $(z_1, m) \rightarrow (z, n)$ for some $(z, n) \in B_2$.

According to this definition a single state of the job stack process is a configuration, z, of the job stack such that no passage times are underway [$h(z, n) \in H$ for all n], all jobs are at the same center (i_0) with exactly one job in service, and there exists z_1 such that a passage time for some job terminates when the job stack process jumps from z_1 to z. [$j_1(i_0), j_2(i_0), \cdots, j_{k(i_0)}(i_0)$ are the classes of jobs serviced at center i_0; the definition requires that $j_{l_n}(i_0)$, the class at center i_0 to which the job in position n (of job stack z) belongs, satisfy $j_{l_n}(i_0) \in \{j_1(i_0), j_2(i_0), \cdots, j_{k(i_0)}(i_0)\}$ for all n.] We assume that a single state of the job stack process exists. In addition, we assume that the initial state, z_0, of the job stack process is a single state.

3. The underlying stochastic structure

The labeled jobs method rests on the result that the job stack process observed at the terminations of passage times is a regenerative process in discrete time. To show this, we use a representation of the job stack process (restricted to the set D) as an irreducible generalized semi-Markov process (GSMP).

Following Whitt [5], a GSMP moves from state to state in accordance with the occurrence of events associated with the occupied state. Each of the several events associated with a state has its own jump distribution for determining the next state, and these events compete with respect to triggering the next transition. At each transition of the GSMP, new events may be scheduled. For each of these new events, a clock indicating the time until the event is scheduled to occur is set according to an independent (stochastic) mechanism. If a scheduled event does not trigger a transition but is associated with the next state, its clock continues to run; if such an event is not associated with the next state, it is abandoned.

The GSMP associated with the job stack process $\mathbf{Z} = \{Z(t) : t \geq 0\}$ has state space D and the completion of service to a job of class j at center i is an event associated with state $z \in D$ provided that $(i, j) \in U(z)$. Jumps of the process from state z triggered by event u are governed by the probability mass function $p(\cdot\,; z, u)$. At a jump from state z to state z' triggered by event u, new clock values are generated for each $\nu \in U(z') - (U(z) - \{u\})$. The distribution function of such a new clock time (a service time for a job class at some center) is denoted by $F(\cdot; z', \nu, z, u)$; by the assumptions in Section 2, each has finite mean and a density function which is continuous and positive on $(0, \infty)$. For $\nu \in U(z') \cap (U(z) - \{u\})$, the old clock reading is retained after the jump. For each $\nu \in (U(z) - \{u\}) - U(z')$,

$$
\begin{array}{c|ccccccc}
 & (1,2) & (1,3) & (1,4) & (1,5) & (1,6) & (1,7) & (2,1) \\
\hline
(1,2) & 0 & 1-\psi_1 & 0 & 0 & 0 & 0 & \psi_1 \\
(1,3) & 1-\psi_2 & 0 & 0 & 0 & 0 & \psi_2 & 0 \\
(1,4) & 0 & 1-\psi_1 & 0 & 0 & 0 & 0 & \psi_1 \\
(1,5) & 1-\psi_2 & 0 & 0 & 0 & 0 & \psi_2 & 0 \\
(1,6) & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
(1,7) & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
(2,1) & 0 & 0 & 0 & 1 & 0 & 0 & 0
\end{array}
$$

Figure 3 Routing matrix **P**.

event ν ceases to be scheduled after the jump. Let $U = \{u_1, u_2, \cdots, u_e\}$ be the set of all possible events that can occur, *i.e.*,

$$U = \bigcup_{z \in D} U(z).$$

Observe that $U = C$, the set of (center, class) pairs in the network.

With each $z \in D$, associate the set of clock readings

$$C(z) = \{(c_1, c_2, \cdots, c_e) : c_i \geq 0 \text{ and } c_i > 0 \text{ if and only if } u_i \in U(z)\},$$

where c_i is the reading on the clock corresponding to event $u_i \in U(z)$. It follows from Proposition 2 that the GSMP is irreducible; *i.e.*, for all states z and z' there exists a finite sequence $u_0', z_1, u_1', \cdots, u_n'$ of events and states satisfying

$$p(z_1; z, u_0')\, p(z_2; z_1, u_1') \cdots p(z'; z_n, u_n') > 0.$$

Label the jobs from 1 to N. For $t \geq 0$ denote by $N^j(t)$ the position of job j in the job stack at time t, $j = 1, 2, \cdots, N$. Then in terms of the vector $Z(t)$ of Eq. (1), set

$$X^0(t) = (Z(t), N^1(t), \cdots, N^N(t))$$

and call the process $\mathbf{X}^0 = \{X^0(t) : t \geq 0\}$ the *fully augmented job stack process*. Denote the state space of this process by G^0. Next select $(z_0, n^1, \cdots, n^N) \in G^0$ and set $X^0(0) = (z_0, n^1, \cdots, n^N)$. Denote by $\{P_n^0 : n \geq 1\}$ the successive passage times (irrespective of job identity) in termination order. Let $\{T_n^0 : n \geq 1\}$ be the corresponding sequence of termination times and set $T_0^0 = 0$.

We now show that the job stack process returns infinitely often (i.o.) to a single state. This leads to the result that $\{(Z(T_n^0), P_{n+1}^0) : n \geq 0\}$ is a regenerative process in discrete time.

Proposition 5 For $t \geq 0$ let $M(t)$ be the last state of $\mathbf{Z} = \{Z(t) : t \geq 0\}$ before entrance to the state occupied at time t and set $Y(t) = (M(t), Z(t))$. Suppose that the routing matrix $\mathbf{P}$ is irreducible and for some service center, i_0, either $k(i_0) = 1$ or service to a job of class $j_{k(i_0)}(i_0)$ at center i_0 is preempted when a job of higher priority joins the queue. Denote by $(\tau_k : k \geq 0\}$ the jump times of $\mathbf{Y} = \{Y(t) : t \geq 0\}$. Then

$$P\{Y(\tau_k) = (z, z') \text{ i.o.}\} = 1$$

for all $z, z' \in D$ such that $z \rightarrow z'$.

Proof The process $\mathbf{Y}$ has state space $F = \{(z, z') : z, z' \in D$ and $z \rightarrow z'\}$. Associate with $(z, z') \in F$ the set of events $U(z')$ and vector of clock settings $C(z')$. Then $\{Y(t) : t \geq 0\}$ is a GSMP with finite state space. Select $(z, z') \in F$. We appeal to Glynn [9] to show that the general state space Markov chain (GSSMC) associated with the GSMP $\mathbf{Y}$ returns infinitely often to the set $\{(z, z')\} \times C(z')$; it follows immediately that $P\{Y(\tau_k) = (z, z') \text{ i.o.}\} = 1$. Three conditions must be checked:

a. The GSMP $\mathbf{Y}$ is irreducible in the sense that $y \curvearrowright y'$ for all $y, y' \in F$;
b. The density functions associated with the clock readings c_i have finite means and are continuous and positive on $(0, \infty)$; and
c. A "recurrence measure" assigns positive measure to the set $\{(z, z')\} \times C(z')$ for $(z, z') \in F$.

Condition (a) holds as a consequence of Proposition 2; $z_2 \curvearrowright z_1'$ for $y = (z_1, z_2)$ and $y' = (z_1', z_2')$. By assumption, condition (b) holds. With respect to condition (c), note that because the set $U(z')$ is nonempty, at least one element of the set $C(z')$ is positive. This implies that the recurrence measure of $\{(z, z')\} \times C(z')$ is positive, in fact, infinite. Hence condition (c) holds and the GSSMC hits $\{(z, z')\} \times C(z')$ infinitely often with probability one. $\square$

Proposition 6 Let z be a single state of the job stack process for the passage time specified by the sets A_1, A_2, B_1, and B_2. Then under the conditions of Proposition 5,

$$P\{Z(T_n^0) = z \text{ i.o.}\} = 1. \tag{7}$$

Proof Condition (2) of the definition of a single state implies that there exists $(z_1, z) \in F$ such that a passage time for some job terminates when $\mathbf{Z}$ jumps from z_1 to z. Therefore, every jump time τ_k such that $Y(\tau_k) = (z_1, z)$ is an element of $\{T_n^0 : n \geq 0\}$ and is such that $Z(\tau_k) = z$. The result now follows from Proposition 5. $\square$

Proposition 7 Under the conditions of Proposition 5, the process $\{(Z(T_n^0), P_{n+1}^0) : n \geq 0\}$ is a regenerative process in discrete time.

Proof Recall that $Z(0) = z_0$, a single state of the job stack process. Let $\{T_{\beta_k}^0 : k \geq 0\}$ be the successive epochs at which a passage time terminates with $\mathbf{Z}$ in state z_0, and set $T_{\beta_0}^0 = 0$. We must show that

a. $\{\beta_k : k \geq 0\}$ is a renewal process, and
b. $P\{Z(T_{\beta_k+m}^0) = z, P_{\beta_k+m+1}^0 \leq x\} =$
$P\{Z(T_m^0) = z, P_{m+1}^0 \leq x\}$ for $m \geq 0, k \geq 0$
and independent of $\{(Z(T_j^0), P(T_{j+1}^0)) : 0 \leq j < \beta_k\}$.

Let $\tau(\beta_k) = \min\{\tau_n : \tau_n > T_{\beta_k}^0\}$. At time $T_{\beta_k}^0$, no passage times are underway. Therefore, $\{Z(t) : t \geq \tau(\beta_k)\}$ determines the distributions of $Z(T_{\beta_k+m}^0)$, $P_{\beta_k+m+1}^0$, and $\beta_{k+1} - \beta_k$ for $m \geq 0$. (Note that $\beta_{k+1} - \beta_k$ is finite with probability one by Proposition 6.) Now observe that each of the clocks running at time $\tau(\beta_k)$ was set at time $\tau(\beta_k)$. The joint distribution of $Z(\tau(\beta_k))$ and the clocks set at $\tau(\beta_k)$ depends on the past history of $\mathbf{Z}$ only through z_0 and the event occurring at time $\tau(\beta_k)$. Hence this distribution is the same for all β_k, independent of $\{(Z(T_n^0), P_{n+1}^0) : n < \beta_k\}$. The future course of $\mathbf{Z}$ is governed by $Z(\tau(\beta_k))$ and the clocks running at time $\tau(\beta_k)$. $\square$

Proposition 8 asserts that the sequences of passage times associated with starting the job stack process in a fixed single state converge to a common random variable. The proof is similar to that of Proposition (4.9) of Shedler and Southard [3].

Proposition 8 Suppose that $x \curvearrowright x'$ for all $x, x' \in E$. Let z_0 be a single state of the job stack process. Denote $X^0(0)$ by $(z_0, n^1, \cdots, n^N)$ and let $\{P_n^0 : n \geq 1\}$ be the successive passage times (irrespective of job identity) enumerated in termination order. Then

$$P_n^0 \Rightarrow P^0 \text{ as } n \rightarrow \infty$$

for all $(z_0, n^1, \cdots, n^N) \in G^0$,

where P^0 is the limiting passage time for any marked job.

4. Simulation for passage times

The goal of the simulation is estimation of

$$r^0(f) = E\{f(P^0)\},$$

where f is a real-valued (measurable) function and P^0 is the limiting passage time for any marked job.

Select a single state, z_0, of the job stack process and an initial state $(z_0, n^1, \cdots, n^N)$ for the fully augmented job stack process $\mathbf{X}^0$. Carry out the simulation of the fully augmented job stack process in blocks defined by the successive epochs $\{T_{\beta_k}^0 : k \geq 1\}$ at which a passage time terminates with the job stack process in state z_0 ($\beta_0 = 0$ and $T_{\beta_0}^0 = 0$). Let

$$Y_m^0(f) = \sum_{j=\beta_{m-1}+1}^{\beta_m} f(P_j^0)$$

and let $\alpha_m^0 = \beta_m - \beta_{m-1}$, $m \geq 1$.

Propositions 9 and 10 lead to point and interval estimates for $E\{f(P^0)\}$. The results follow directly from Proposition 7 and general theorems for regenerative processes; see Crane and Iglehart [10].

Proposition 9 The pairs of random variables

$$\{(Y_m^0(f), \alpha_m^0) : m \geq 1\}$$

are independent and identically distributed.

Proposition 10 Let $D(f)$ be the set of discontinuities for the function f. Provided that $P\{P^0 \in D(f)\} = 0$ and $E\{|f(P^0)|\} < \infty$,

$$E\{f(P^0)\} = E\{Y_1^0(f)\}/E\{\alpha_1^0\}.$$

Given Propositions 9 and 10, the standard regenerative method applies and (from a fixed number, n, of blocks) provides the strongly consistent point estimate

$$\hat{r}_n^0(f) = \overline{Y}_n^0(f)/\overline{\alpha}_n^0,$$

where

$$\overline{Y}_n^0(f) = n^{-1}\sum_{m=1}^{n} Y_m^0(f)$$

and

$$\overline{\alpha}_n^0 = n^{-1}\sum_{m=1}^{n} \alpha_m^0.$$

Confidence intervals for $r^0(f)$ are based on the central limit theorem

$$n^{1/2}[\hat{r}_n^0(f) - r^0(f)]/[\sigma(f)/E\{\alpha_1^0\}] \Rightarrow N(0, 1),$$

where $\sigma^2(f)$ is the variance of $Y_1^0(f) - r^0(f)\alpha_1^0$ and $N(0, 1)$ is a standardized (mean 0, variance 1) normal random variable. It follows that (for $0 < \gamma < 1/2$)

$$\hat{I}_n(f) = [\hat{r}_n^0(f) - z_{1-\gamma}s_n/(\overline{\alpha}_n^0 n^{1/2}), \hat{r}_n(f) + z_{1-\gamma}s_n/(\overline{\alpha}_n^0 n^{1/2})]$$

is a $100(1 - 2\gamma)\%$ confidence interval for $r^0(f)$. Here $z_{1-\gamma} = \Phi^{-1}(1 - \gamma)$, where $\Phi(\cdot)$ is the distribution function of a standardized normal random variable, and s_n is the quantity

$$s_n = [s_{11} - 2\hat{r}_n(f)s_{12} + (\hat{r}_n(f))^2 s_{22}]^{1/2}.$$

The quantities s_{11}, s_{22}, and s_{12} are the usual unbiased estimates for $\sigma^2\{Y_k(f)\}$, $\sigma^2\{\alpha_k^0\}$, and cov $\{Y_k^0(f), \alpha_k^0\}$, respectively.

Example 11 Consider a network with two service centers and two job classes such that $C = \{(1, 1), (2, 1), (2, 2)\}$ and jobs of class 2 have nonpreemptive priority over jobs of class 1 at center 2. For $t \geq 0$ let

$$Z(t) = (Q_1(t), C_1^{(2)}(t), C_2^{(2)}(t), S_2(t)).$$

Suppose the irreducible routing matrix $\mathbf{P}$ is

$$\begin{array}{c} \\ (1,1) \\ (2,1) \\ (2,2) \end{array} \begin{array}{c} \begin{array}{ccc} (1,1) & (2,1) & (2,2) \end{array} \\ \left[\begin{array}{ccc} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{array}\right] \end{array}.$$

Since center 1 sees only one job class, we can take $i_0 = 1$ and $z_{i_0}^* = (2, 0, 0, 0)$. Since all elements of the set D^* except $(0, 0, 2, 2)$ and $(0, 1, 1, 1)$ are accessible from z_1^*, the set $D = D^* - \{(0, 0, 2, 2,), (0, 1, 1, 1)\}$. According to Proposition 2, any two elements of D communicate.

Consider the passage time specified by the subsets A_1, A_2, B_1, and B_2 of E:

$$A_1 = \{(1, 0, 1, 2, 1), (1, 1, 0, 1, 1), (2, 0, 0, 0, 2)\},$$

$$A_2 = \{(0, 1, 1, 2, 1), (0, 2, 0, 1, 1), (1, 1, 0, 1, 2)\},$$

$$B_1 = \{(0, 1, 1, 2, 2), (1, 0, 1, 2, 2)\},$$

and

$$B_2 = \{(1, 1, 0, 1, 1), (2, 0, 0, 0, 1), (1, 0, 1, 2, 1)\}.$$

Observe that $S \cap T = \varnothing$. The passage time starts when a job joins the queue at center 2 as class 1 and terminates when the job completes service at center 2 as class 2. [Note that because $(0, 0, 2, 2,) \notin D$, $(0, 0, 2, 2, 2) \notin B_1$.] For this passage time, $H = \{(1, 1)\}$ and $z_0 = (2, 0, 0, 0)$ is a single state. Clearly, z_0 satisfies condition (1) of Definition 4. Condition (2) is also satisfied because

$$(1, 0, 1, 2, 2) \rightarrow (2, 0, 0, 0, 1).$$

Now consider the passage time specified by the subsets A_1, A_2, B_1, and B_2 of E:

$$A_1 = \{(1, 0, 1, 2, 2), (0, 0, 2, 2, 2), (0, 1, 1, 2, 2)\},$$

$$A_2 = \{(2, 0, 0, 1, 1), (1, 0, 1, 2, 1), (1, 1, 0, 1, 1)\},$$

$$B_1 = \{(1, 1, 0, 1, 2), (0, 2, 0, 1, 2)\},$$

and

$$B_2 = \{(1, 0, 1, 2, 2), (0, 1, 1, 2, 2)\}.$$

The passage time starts when a job completes service at center 2 as class 2 and terminates when the job completes service at center 2 as class 1. For this passage time, the set $H = \{(2, 2)\}$. There is at least one passage time underway unless the configuration of the job stack is $(0, 0, 2, 2)$. Since $(0, 0, 2, 2) \notin D$, there is no single state and the estimation method of this paper does not apply.

5. Concluding remarks

When developing simulation methodology it is important to be able to assess the statistical efficiency of proposed estimation procedures. With the labeled jobs method developed in this paper, the half-length of the confidence interval (obtained from a simulation of fixed length) for the expected value of a general function f of the limiting passage time is proportional to a quantity $e^0(f)$. This quantity is independent of the blocks of the underlying regenerative process and therefore is an appropriate measure of the statistical efficiency of the estimation procedure. For Markovian networks of queues, it is possible to compute theoretical values for expected passage times and the associated variance constants appearing in central limit theorems used to form confidence intervals for passage time characteristics. This leads to a quantitative assessment (Iglehart and Shedler [2]) of the relative statistical efficiencies of the estimation procedures in [1] for networks with Cox-phase service times. For networks of queues with general service times, there is little hope of computing the needed theoretical values, even for expected passage times. Central limit theorems and continuous mapping theorem arguments can be used to show [11] that the confidence intervals obtained from the labeled jobs method are shorter than those obtained by observation of a single "marked job." This is consistent with intuition since the labeled jobs method extracts more passage time information from a fixed length simulation run.

The requirements for applicability of the labeled jobs method of Section 4 are that (a) there exist sets D and E as in Eqs. (6) and (7), (b) the sets A_1, A_2, B_1, and B_2 which define the passage time are subsets of E, and (c) there is recurrence in the sense of Proposition 5. We have shown that when all service time density functions are continuous and positive on $(0, \infty)$, it is sufficient that either some service center see only one job class or the lowest priority job class seen by the center be subject to preemption. The requirement that there exist a "single state" of the job stack process [as in Eq. (10)] is essential.

For networks in which all centers are multiple servers, and for some passage times in networks with single servers, no single state will exist. Extending the procedure to handle such situations is an open problem.

We have assumed that any interruption of service at a center is of the preemptive-repeat (rather than preemptive-resume) type. Passage times for preemptive-resume networks can be recorded by observing a fully augmented job stack process which maintains the position of each of the jobs in the job stack along with whether or not the job has been preempted. Specifically, let

$$X^0(t) = (Z(t), Q^1(t), Q^2(t), \cdots, Q^N(t), N^1(t), N^2(t), \cdots, N^N(t)),$$

where $Q^j(t)$ equals 1 if the job in position j of the job stack at time t is in queue at some service center and its most recent service has not been completed, and equals 0 otherwise. (Note that we do not incorporate the remaining service time of a preempted job into the state definition.) Then estimates for passage times can be obtained as in Section 4 provided that the process $\{(Z(t), Q^1(t), Q^2(t), \cdots, Q^N(t)) : t \geq 0\}$ observed at terminations of passage times is a regenerative process in discrete time. This will be the case if the process $\{(Z(t), Q^1(t), Q^2(t), \cdots, Q^N(t)) : t \geq 0\}$ returns infinitely often to a single state. The process $\{(Z(t), Q^1(t), Q^2(t), \cdots, Q^N(t)) : t \geq 0\}$ can be represented as a GSMP in which the trigger event depends on the clocks for events associated with the occupied state as well as the speeds (possibly zero) at which these clocks run. [Completion of service to some job labeled l of class j at center i is an event associated with a state $(z, q^1, q^2, \cdots, q^N)$ of the GSMP provided that $(i, j) \in U(z)$. If $q^k = 1$, then (for some i and j) completion of service to job k as class j at center i is an event associated with state $(z, q^1, q^2, \cdots, q^N)$ and in this state the clock for this event runs at zero speed.] We conjecture that even if there are states in which some of the clocks run at zero speed, the process $\{(Z(t), Q^1(t), Q^2(t), \cdots, Q^N(t)) : t \geq 0\}$ returns infinitely often to a single state if conditions (a), (b), and (c) in the proof of Proposition 5 are satisfied.

Acknowledgment
Gerald S. Shedler gratefully acknowledges partial support from the National Science Foundation under Grant MCS-7909139.

References

1. D. L. Iglehart and G. S. Shedler, *Regenerative Simulation of Response Times in Networks of Queues, Lecture Notes in Control and Information Sciences,* Volume 26, Springer-Verlag New York, Inc., New York, 1980.
2. D. L. Iglehart and G. S. Shedler, "Regenerative Simulation of Response Times in Networks of Queues: Statistical Efficiency," *Acta Informatica* **15,** 347–363 (1981).
3. G. S. Shedler and J. Southard, "Simulation for Passage Times in Closed Multiclass Networks of Queues with Unrestricted Priorities," *Research Report RJ3190,* IBM Research Laboratory, San Jose, CA, 1981.
4. D. L. Iglehart and G. S. Shedler, "Simulation for Passage Times in Closed, Multiclass Networks with Queues with General Service Times," *Research Report RJ3191,* IBM Research Laboratory, San Jose, CA, 1981.
5. W. Whitt, "Continuity of Generalized Semi-Markov Processes," *Math. Oper. Res.* **5,** 494–501 (1980).
6. G. S. Shedler and D. R. Slutz, "Irreducibility in Closed Multiclass Networks of Queues with Priorities: Passage Times of a Marked Job," *Perf. Eval.* **1,** 334–343 (1981).
7. S. S. Lavenberg and G. S. Shedler, "Stochastic Modeling of Processor Scheduling with Application to Data Base Management Systems, *IBM J. Res. Develop.* **20,** 437–448 (1976).
8. L. D. Fossett, "Simulating Generalized Semi-Markov Processes," *Technical Report No. 2,* Department of Operations Research, Stanford University, Stanford, CA, 1979.
9. P. W. Glynn, Department of Operations Research, Stanford University, Stanford, CA, private communication.
10. M. A. Crane and D. L. Iglehart, "Simulating Stable Stochastic Systems. III, Regenerative Processes and Discrete Event Simulation," *Oper. Res.* **23,** 33–45 (1975).
11. D. L. Iglehart and G. S. Shedler, "Comparison of Regenerative Simulation Methods for Networks of Queues with General Service Times," *Research Report RJ3415,* IBM Research Laboratory, San Jose, CA, 1982.

Received September 10, 1981; revised February 8, 1982

Gerald S. Shedler *IBM Research Division, 5600 Cottle Road, San Jose, California 95193.* Mr. Shedler has been a Research staff member at IBM since 1965, initially at the Thomas J. Watson Research Center, Yorktown Heights, New York, and, since 1970, in the Computer Science Department at the Research laboratory in San Jose. During 1973–1974, while on sabbatical from IBM, he was associated with Stanford University as Acting Associate Professor in the Department of Operations Research, and subsequently has been Consulting Associate Professor in the same department. He has worked extensively on applications of stochastic processes, particularly to performance evaluation of computer systems. His current research is on discrete event methods for simulation of stochastic systems. Mr. Shedler is coauthor of *Regenerative simulation of response times in networks of queues,* published in 1980.

Jonathan Southard *Stanford University, Stanford, California 94305.* Mr. Southard is a graduate student in the Department of Mathematics at Stanford University. He received his B.S. from Stanford University in 1982. In 1981 Mr. Southard was employed by IBM in San Jose, California, at the Research laboratory working on methodology for discrete event computer simulation.