GENERATING RANDOM VARIATES FROM A DISTRIBUTION OF PHASE TYPE

Marcel F. Neuts
and
Miriam E. Pagano
Department of Mathematical Sciences
University of Delaware
Newark, Delaware 19711

ABSTRACT

An algorithm for generating phase type random variates is presented. A phase type distribution is any distribution which may be that of the time until absorption in a finite state Markov process. The exponential sojourn times allow us to record only the number of visits, k, to each state before absorption and then generate appropriate Erlang-k random variates. A related algorithm for the superposition of renewal processes of phase type is also discussed.

I. INTRODUCTION

Random variate generation is an important tool in simulation as indicated by the abundance of literature in this area. This literature spans over twenty years of research and is still growing. Sampling algorithms have been devised and implemented for almost all of the classical univariate distributions. These have then been examined, corrected and enhanced by various authors. A basic reference on random variate generation is Fishman [4].

Our present concern is to study the generation of random variates from distributions defined as absorption time distributions in absorbing Markov chains. Such distributions are called phase type distributions (PH-distributions) and were first introduced in [8]. Their use in queues and reliability is discussed in [9] and [10]. Erlang, hyperexponential and geometric distributions are very special cases of PH-distributions.

It is necessary to develop algorithms for both continuous and discrete PH-distributions. These algorithms are analogous but the desire for efficient implementation warranted careful consideration of each case.

As an application of the methods for the generation of PH-variates, we developed algorithms for realizations of the superposition of stationary PH-renewal processes. These are renewal processes in which the distribution of the time between renewals is of phase type. PH-renewal processes were first discussed in [8], explored in more detail in [11] and are also treated in [9]. The Markovian nature of the PH-distribution leads to efficient and exact algorithms with applications in other contexts, such as multi-server queues and stochastic networks. These applications will not be discussed here.

II. BACKGROUND MATERIAL

A. PH-Distributions

There are two parallel discussions of PH-distributions; one corresponding to distributions on $[0,\infty)$ obtained from absorption times in continuous parameter Markov chains; the other to distributions on the nonnegative integers obtained from discrete parameter Markov chains. The detailed discussion of the continuous case, along with the appropriate discrete analogues, is presented in [9]. Only relevant definitions and properties shall be mentioned here.

A distribution $F(\cdot)$ on $[0,\infty)$ is a PH-distribution if it is that of the time until absorption in a finite state Markov process with generator

$$1) \qquad Q = \begin{vmatrix} T & \underline{T}^\circ \\ \underline{0} & 0 \end{vmatrix}$$

and initial probability vector $(\underline{\alpha},\alpha_{m+1})$. T is a non-singular matrix of order m and satisfies $T_{ii} < 0$, for $1 \leq i \leq m$, and $T_{ij} \geq 0$, for $i \neq j$. Also $T\underline{e} + \underline{T}^\circ = \underline{0}$ and $\underline{\alpha}\,\underline{e} + \alpha_{m+1} = 1$, where $\underline{e}$ denotes a column vector of appropriate dimension with all components equal to one. The distribution $F(\cdot)$ is then given by

$$2) \qquad F(x) = 1 - \underline{\alpha}\exp(Tx)\underline{e}\,, \qquad \text{for } x \geq 0$$

and we say F has the representation $(\underline{\alpha},T)$. If $\alpha_{m+1} > 0$ then F has a jump of height α_{m+1} at

81CH1709-5/81/0000-0381$00.75 © 1981 IEEE

the origin. The mean, μ_1', is given by $\mu_1' = -\underline{\alpha}T^{-1}\underline{e}$.

The analogous definition for the density $\{r_k\}$ on the nonnegative integers is as follows. If $\{r_k\}$ is the density of the time until absorption in a finite state Markov chain with stationary transition probability matrix given by

$$P \begin{vmatrix} T & \underline{T}^\circ \\ \underline{0} & 1 \end{vmatrix}$$

and initial probability vector $(\underline{\alpha},\alpha_{m+1})$, the density $\{r_k\}$ is of phase type. T is an $m \times m$ substochastic matrix such that $T\underline{e} + \underline{T}^\circ = \underline{e}$ and $I - T$ is nonsingular. The density of the time until absorption is given by $r_0 = \alpha_{m+1}$ and $r_k = \underline{\alpha}\, T^{k-1}\underline{T}^\circ$, for $k \geq 1$. Note that when $m = 1$, the exponential and geometric distributions are obtained for the continuous and discrete cases respectively.

B. PH-Renewal Processes

The definition of the PH-renewal process is used in the development of the superposition algorithms. Consider the following construction. Suppose that upon absorption into the state $m + 1$ of the Q chain given in (1), independent multinomial trials are performed with probabilities $(\underline{\alpha},\alpha_{m+1})$ until one of the alternatives $1,2,\ldots,m$ occurs. The Markov chain is restarted in the state so determined and this procedure is repeated at the times of successive absorptions. By considering path functions which are right-hand continuous, a Markov chain with the m states $1,2,\ldots,m$ is obtained. The times between absorption into state $m + 1$ are independent and identically distributed with common distribution $F(\cdot)$ given in (2). The infinitesimal generator for this chain is given by Q^* where $Q^* = T + T^\circ A^\circ$. T° is an $m \times m$ matrix with identical columns $\underline{T}^\circ$ and $A^\circ = (1 - \alpha_{m+1})^{-1}\text{diag}(\alpha_1,\ldots,\alpha_m)$. Note that $T^\circ A^\circ$ may also be written as $(1 - \alpha_{m+1})^{-1}\underline{T}^\circ\underline{\alpha}$. The representation $(\underline{\alpha},T)$ is said to be irreducible if and only if Q^* is irreducible. In [9] it is shown that we may always choose Q^* to be irreducible.

Since Q^* is irreducible, existence of the stationary probability vector, $\underline{\pi}$, of Q^* is assured. The following results are proved in [9].

Theorem: If $F(\cdot)$ is a PH-distribution with, mean μ_1', with (irreducible) representation $(\underline{\alpha},T)$ and with corresponding vector $\underline{\pi}$, then

$$F^*(x) = \frac{1}{\mu_1'}\int_0^x [1-F(u)]du \ , \quad \text{for} \quad x \geq 0,$$

is a PH-distribution with representation $(\underline{\pi},T)$.

Definition: A probability distribution $F(\cdot)$ is asymptotically exponential if and only if for some $K > 0$ and $\eta > 0$

$$1 - F(x) = Ke^{-\eta x} + o(e^{-\eta x}), \quad \text{as} \quad x \to \infty .$$

Theorem: If the matrix T is irreducible, any PH-distribution $F(\cdot)$ with representation $(\underline{\alpha},T)$ is asymptotically exponential and $-\eta$ is the eigenvalue with largest real part of T. The constant $K = \underline{\alpha}\ \underline{v}$, where $\underline{v}$ is the positive right eigenvector of T, corresponding to $-\eta$, is uniquely determined by the requirements that $\underline{u} \cdot \underline{v} = \underline{u} \cdot \underline{e} = 1$, where $\underline{u}$ is a left eigenvector of T, associated with $-\eta$.

C. The Alias Method

Walker's alias method as explained by Kronmal and Peterson [6] is a modified acceptance/rejection technique for generating random variates from a discrete distribution. It is based on the following theorem proved in [6].

Theorem: Any discrete distribution with density $p_n(k_i)$, $1 \leq i \leq n$, with a finite number (n) of outcomes can be expressed as an equiprobable mixture of n two-point distributions $q_{k_i}(\cdot)$, $1 \leq i \leq n$, in such a way that each mass point, k_i, of $p_n(\cdot)$ is a mass point of $q_{k_i}(\cdot)$.

Definition: For $1 \leq i \leq n$, $q_{k_i}(k_i)$ is called the cutoff value for k_i and the mass point of $q_{k_i}(\cdot)$ that is not k_i is the alias of k_i.

By generating only one uniform variate we can produce a variate from $p_n(\cdot)$. In the notation of Kronmal and Peterson the algorithm is as follows.

Alias

1] Generate U uniform on $(0,n)$.

2] Set $I \leftarrow \uparrow U \uparrow$, i.e. I is the smallest integer greater than or equal to U. I is now an integer uniform on $\{1,2,\ldots,n\}$.

3] Set $U \leftarrow I - U$. U is now uniform $(0,1)$.

4] If $U \leq q_{k_I}(k_I)$ then return k_I; else return j, the alias of k_I.

To implement the algorithm, three full-word vectors each of dimension n are required. These contain

1) the original density $p_n(\cdot)$,

2) the cutoff values for each k_i,

3) the aliases of each k_i.

An overhead cost of $3n$ words is small for a method which requires only one comparison and one table lookup per variate generated.

To generate these vectors of cutoff values and aliases, Kronmal and Peterson devised an algorithm with two appealing features. No extra "temporary" storage is required and the number of operations to create the list is proportional to n. The procedure systematically reduces the support of $p_n(\cdot)$ by one point so that after k reductions:

$$p_n(\cdot) = \frac{k-1}{n}\, p_{n-k}(\cdot) + \frac{1}{n}\,[q_{j_1}(\cdot) + q_{j_2}(\cdot) + \dots + q_{j_k}(\cdot)] \ .$$

The mass point j_i is chosen such that $p_{n+1-i}(j_i) < 1/(n+1-i)$, while its alias, ℓ, has $p_{n+1-i}(\ell) \geq 1/(n+1-i)$. A linked list structure determines the indices j_i and the corresponding ℓ-values efficiently.

D. Generating Random Variates from an Erlang Distribution

In the sequel we shall need to generate Erlang (k,λ) variates. Since an Erlang(k,λ) random variable is the sum of k independent exponential random variables, each with parameter λ, the random variate may be generated as such. The algorithm, given below, is constructed so that the logarithm function need only be executed once.

```
Erlang (k,λ)
1] Generate $U_i$ uniform on (0,1) for i=1,2,...,k.
2] Set $U = \prod_{j=1}^{k} U_j$.
3] $X \leftarrow -\frac{1}{\lambda}\,\ell n(U)$.
```

The execution time is directly proportional to k. If k is greater than fifteen, we use the method of Cheng [3] to generate a gamma (k,1) variate, X. $Y \leftarrow \lambda X$ is then gamma (k,λ).

Cheng's method is an acceptance/rejection technique whose efficiency increases as k increases. It requires two uniform and three log/exponential evaluations per trial. The expected number of trials per random variate is 1.47 for $k = 1$ and tends to 1.13 as $k \to \infty$. The algorithm in compact form is as follows.

```
Gamma
1] Generate $U_1$, and $U_2$ uniform (0,1).
2] Set $V \leftarrow a\,\ell n[U_1/(1-U_1)]$.
   Set $X \leftarrow ke^V$.
3] (Accept or Reject). If $b + cV - X \geq \ell n(U_1^2 U_2)$ then accept X; else return to 1.
```

The constants are $a = (2k-1)^{-1/2}$, $b = k - \ell n4$, and $c = k + \frac{1}{a}$. For our implementation a, b, and c must be computed each time the algorithm is invoked.

III. THE PROCEDURE FOR THE DISCRETE CASE

Consider a discrete PH-distribution on m states with representation $(\underline{\alpha},T)$ and transition probability matrix

$$P = \begin{vmatrix} T & \underline{T}^\circ \\ \underline{0} & 1 \end{vmatrix} .$$

Let $\underline{T}_i$ be the ith row of T and T_i° the ith component of $\underline{T}^\circ$. The algorithm simulates the Markov chain and retains only the number of time epochs required before absorption into state m + 1. This count is a PH random variate.

The simulation is accomplished through use of the alias method for the initial probability vector $(\underline{\alpha},\alpha_{m+1})$ as well as for each of the rows $(\underline{T}_i,T_i^\circ)$, $1 \leq i \leq m$. Since the absorbing state m + 1 requires neither storage nor computational effort, the total storage needed is $3(m+1)^2$ full words. The simulation algorithm is as follows.

```
PH1
1] Initialization
   A) Create aliases and cutoff values for $(\underline{\alpha},\alpha_{m+1})$.
   B) Create aliases and cutoff values for $(\underline{T}_i,T_i^\circ)$, i=1,2,...,m.
2] Repeat for the number of variates desired.
   A) Initialize random variate $X_{PH}$ to 0.
   B) From $(\underline{\alpha},\alpha_{m+1})$ choose current state, I.
   C) If (I is state m+1) then branch to D;
      else
        i)   $X_{PH} \leftarrow X_{PH} + 1$ ,
        ii)  choose next state J from $(\underline{T}_I,T_I^\circ)$ ,
        iii) $I \leftarrow J$ ,
        iv)  return to C.
   D) (Absorbed). Return $X_{PH}$.
```

Implementation of the algorithm was accomplished in FORTRAN on a Burroughs 7700. As this is a computer with a 48-bit word length, adequate accuracy is granted by single precision. To gain efficiency, our data structures take advantage of the standard FORTRAN features <u>column major ordering</u> and <u>call by reference</u>. The matrix $\hat{P} = |T\ \underline{T}^\circ|$ is stored in transposed form as the FORTRAN variable XGEN, dimensioned (m+1) × (m+1), so creating stochastic columns. The (m+1)st column contains the vector $(\underline{\alpha},\alpha_{m+1})$. The aliases and cutoff values needed are stored in the corresponding columns of matrices which we denote by L and F respectively. By referencing these matrices only in subroutines, double indexing computations are avoided. The implementation herein allows the user to specify the transition matrix in the usual probabilistic row format so that this transposition technique is of no consequence to him. This should, however, be borne in mind if XGEN is accessed for later computation. Throughout this work uniform variates on (0,1) are generated by the congruential generator devised by Lewis et al. [7] available through IMSL (version 8).

IV. THE PROCEDURE FOR THE CONTINUOUS CASE

Consider the continuous PH-distribution on m states with representation $(\underline{\alpha},T)$. Let the infinitesimal generator

$$Q = \begin{vmatrix} T & \underline{T}^\circ \\ \underline{0} & 0 \end{vmatrix}$$

have elements Q_{ij}, for $i,j=1,2,\ldots,m+1$, where $Q_{ii} < 0$ for each i.

In the simulation of the underlying Markov chain the time spent in state i before making a transition to state j is a random variate contributing to the total time until absorbtion. Such sojourn times in state i are exponentially distributed with parameter $-Q_{ii}$. The transition to state j occurs with probability $Q_{ij}/(-Q_{ii})$. By retaining the number of times, k_i, that state i was encountered before absorption, a PH variate results as the sum of Erlang variates. That is,

$$X_{PH} = \sum_{i=1}^{m} \text{Erlang } (k_i, -Q_{ii}) \; .$$

To simulate the state transitions by using the alias method, it is necessary to consider the discrete time Markov chain embedded at the transitions of the Markov process Q. For $1 \leq i \leq m$, $-Q_{ii}^{-1}$ is placed in a temporary location, XEXP(i), to be used in the generation of Erlang variates. Let $\hat{Q}$ denote the probability transition matrix of the embedded chain. Then for $1 \leq i \leq m$,

$$\hat{Q}_{ij} = Q_{ij} * \text{XEXP}(i) \; , \quad \text{for} \quad 1 \leq j \leq m \; ,$$

$$\hat{Q}_{ii} = 0 \; ,$$

$$\hat{Q}_{i,m+1} = 1 - \sum_{j=1}^{m} \hat{Q}_{ij} \; .$$

The normalizing condition applied to $\hat{Q}_{i,m+1}$ assures that the rows of $\hat{Q}$ sum to one within computer limitations. The necessary tables needed for the alias method are computed from the stochastic rows of $\hat{Q}$.

The extra matrices required of the alias method, the matrix Q and the Erlang parameter vector XEXP, bring the storage requirements to $3(m+1)^2 + m$ full words. Q is stored to utilize column major ordering. Q is then replaced by $\hat{Q}$. This avoids increasing the storage by m^2 words.

Let k_i be the number of visits to state i. Let $\underline{\hat{Q}}_i$ be the ith row of $\hat{Q}$. The simulation algorithm is as follows.

```
PH2
1]  (Initialization).
    A)  Compute $\hat{Q}$.
    B)  $k_i \leftarrow 0$;  for  $i=1,2,\ldots,m$.
    C)  Generate alias tables for  $\hat{Q}$.
2]  (Simulate the underlying Markov chain).
    Repeat for the number of variates desired.
    A)  $X_{PH} \leftarrow 0$.
    B)  From  $(\underline{\alpha},\alpha_{m+1})$  choose current state,  $i$.
    C)  If  $(i=m+1)$  then branch to D;  else
          i)  $k_i \leftarrow k_i + 1$ ,
         ii)  from $\underline{\hat{Q}}_i$ choose next state,  $j$,
        iii)  $i \leftarrow j$ ,
         iv)  branch to C.
    D)  (Absorbed).  Repeat  $i = 1$  to  $m$.
          i)  $X_{PH} = X_{PH}$ + Erlang $(k_i$,XEXP$(i))$.
         ii)  $k_i \leftarrow 0$ .
        End repeat.
    End repeat.
```

V. SUPERPOSITION OF STATIONARY PH-RENEWAL PROCESSES

It is well known that under suitable conditions the superposition of independent renewal processes is asymptotically Poisson. This limit theorem as an approximation theorem has been recently discussed in Albin [2]. We now define an exact process for the simulation of r independent stationary PH-renewal processes.

A. The Continuous Case

Suppose $F(\cdot)$ is a continuous PH-distribution with representation $(\underline{\alpha},T)$ where $T = \{T_{ij}\}$, $i,j = 1,2,\ldots,m$. Let $Q^* = T + T^{\circ}A^{\circ}$ be the infinitesimal generator for the Markov chain associated with the PH-renewal process with underlying distribution F. Consider the superposition of r i.i.d. stationary renewal processes each with underlying distribution $F(\cdot)$. The superposition variate consists of the time between successive renewals among these r processes. Let X_s be the sth variate generated for $s \geq 1$.

In the simulation we maintain one counter for each of the m phases of the Markov chain with infinitesimal generator T. Let $N_j(i)$ be the number of processes in phase i at the jth iteration of determining X_s, for $1 \leq i \leq m$, $j \geq 0$. Then,

$$\sum_{i=1}^{m} N_j(i) = r, \quad \text{for} \quad j \geq 0 \; .$$

To begin the simulation we generate r multinomial observations from the stationary vector, $\underline{\pi}$, of Q^*. $N_0(\cdot)$ is then initialized.

Consider the process in the (n-1)st iteration where $n \geq 1$. From the exponential sojourn times characterizing the states of the r individual processes, we have that the time until the next phase transition is exponentially distributed with parameter θ_{n-1} given by

$$\theta_{n-1} = - \sum_{j=1}^{m} N_{n-1}(j)\, T_{jj} \; .$$

Generate such an exponential variate, $E(\theta_{n-1})$.

Let the phase from which this transition will occur be denoted by j. Using a probabilistic argument we may generate j as a discrete variate from $\hat{P}_{n-1}(\cdot)$, where

$$3) \quad P_{n-1}(\nu) = -N_{n-1}(\nu)T_{\nu\nu} \; ,$$

$$\hat{P}_{n-1}(\nu) = P_{n-1}(\nu)/\theta_{n-1} \; , \qquad \text{for} \quad 1 \leq \nu \leq m \; .$$

The state h to which the process moves is determined by T. As in the algorithm PH2, h is an observation from $q_j(\nu)$ where

$$4)\quad q_j(\nu)=\begin{cases}-T_{j\nu}/T_{jj}\ , & \text{for}\quad 1\le\nu\le m,\quad \nu\ne j,\\ \quad 0\ , & \text{for}\quad \nu=j\ ,\\ -T_j^{\circ}/T_{jj}\ , & \text{for}\quad \nu=m+1\ .\end{cases}$$

If <u>h is not m+1</u> then update the nth iteration for X_s as follows.

<u>UPDATE1</u>

```
1] $N_n(j) = N_{n-1}(j) - 1$
2] $N_n(h) = N_{n-1}(h) + 1$
3] $N_n(i) = N_{n-1}(i)$,  for  $i \ne h,j$
4] $\theta_n = \theta_{n-1} + T_{jj} - T_{hh}$
5] $P_n(j) = P_{n-1}(j) + T_{jj}$
6] $P_n(h) = P_{n-1}(h) - T_{hh}$
```

If <u>h is m+1</u>, then a renewal among the r processes has occurred. The variate X_s is the sum of n exponential variates. That is,

$$X_s = \sum_{i=0}^{n-1} E(\theta_i)\ .$$

We now restart the process which has renewed by observing variates from the discrete density $(\underline{\alpha},\alpha_{m+1})$ until one of the states $1,2,\ldots,m$ occurs. Suppose the ℓth trial is the first trial that a state, k_ℓ, not equal to $m+1$ is observed. If $\ell-1>0$, instantaneous renewals occur and the next $\ell-1$ interevent intervals of the superposition process are equal to zero. That is,

$$X_{s+1} = X_{s+2} = \ldots = X_{s+\ell-1} = 0\ .$$

To begin the first iteration for the random variate $X_{s+\ell}$, we update the necessary parameters as follows.

<u>UPDATE2</u>

```
1] $N_0(j) = N_{n-1}(j) - 1$
2] $N_0(k_\ell) = N_{n-1}(k_\ell) + 1$
3] $N_0(i) = N_{n-1}(i)$,  for  $i \ne j,\ i \ne k_\ell$
4] $P_0(j) = P_{n-1}(j) + T_{jj}$
5] $P_0(k_\ell) = P_{n-1}(k_\ell) - T_{k_\ell k_\ell}$
6] $\theta_0 = \theta_{n-1} + T_{jj} - T_{k_\ell k_\ell}$ .
```

The method continues until all required variates are generated.

The superposition algorithm only requires that we generate random variates from discrete distributions and an exponential distribution. The variates from the discrete densities $(\underline{\alpha},\alpha_{m+1})$ and $q_j(\cdot)$ for $1\le j\le m$ defined in (4) are generated by the alias method. At each iteration the exponential distribution parameter θ_{n-1} is changed. For this reason exponential derivatives are generated by calling a logarithm function.

To avoid m unnecessary multiplications per iteration, $\hat{P}_{n-1}(\cdot)$ defined in (3) is not stored. Instead we generate an observation from $\hat{P}_{n-1}(\cdot)$ directly from $P_{n-1}(\cdot)$ using the inverse distribution method as follows.

<u>PN</u>

```
1] Generate U uniform (0,1).
2] $U \leftarrow U * \theta_{n-1}$.  U is now uniform $(0,\theta_{n-1})$ .
3] $X \leftarrow P_{n-1}(1)$
   $I \leftarrow 2$ .
4] If $(X \ge U)$ then return X; else $X \leftarrow X +$
   $P_{n-1}(I)$,  $I \leftarrow I + 1$,  branch to 4.
```

At each iteration of the superposition algorithm, two mass points of $P_{n-1}(\cdot)$ are altered. The commonly used sorting of the set of probability weights is not carried out here. Since this would require up to m comparisons per iteration, it would not enhance the algorithm PN.

The superposition algorithm begins by generating variates from the discrete distribution $\underline{\pi}$. After simplifying we may write $\underline{\pi}$ as

$$\underline{\pi} = (\underline{\alpha}T^{-1}\underline{e})^{-1}\,\underline{\alpha}T^{-1}\ .$$

Let $\underline{\pi}' = (\underline{\alpha}T^{-1}\underline{e})\underline{\pi}$. Then $\underline{\pi}'$ is computed using Gauss-Seidel iteration to solve the system

$$\pi' T = \underline{\alpha}\ .$$

Then we see that

$$\pi_i = \pi_i'/(\sum_{i=1}^{m}\pi_i')\ , \quad \text{for}\quad 1\le i\le m\ .$$

As the number r of superimposed processes is typically large, the alias method is used to generate variates from $\underline{\pi}$. The extra storage that this method requires is then reused for the alias tables required to generate variates from $q_j(\cdot)$ for $1\le j\le m$. The total storage required is $3(m+1)^2 + 3m$ full words.

Let $N(j)$ be the number of processes in phase j for $1\le j\le m$. Let $P(\cdot)$ denote $P_{n-1}(\cdot)$ and let θ denote θ_{n-1}. The superposition algorithm follows.

<u>SUPER1</u>

```
1] (Generate initial variates from $\underline{\pi}$)
    A) Compute $\underline{\pi}$.
    B) Compute alias tables for $\underline{\pi}$.
    C) Generate r variates from $\underline{\pi}$ initial-
       izing N(j) for $1 \le j \le m$.
2] (Initialize)
    A) Compute alias tables for $q_j(\cdot)$ for
       $1 \le j \le m$.
    B) Compute alias tables for $(\underline{\alpha},\alpha_{m+1})$ .
    C) Initialize P(·) and θ.
    D) $s \leftarrow 1$ .
3] (Generate random variates, $X_s$)
    A) $X_s \leftarrow 0$.
    B) $X_s \leftarrow X_s + E(\theta)$.
    C) $j \leftarrow$ PN.  j is a variate from P(·).
    D) Generate h from $q_j(\cdot)$.
    E) If (h=m+1) then
       1) store $X_s$,
       2) UPDATE2,
        3) s ← s + 1,
        4) branch to A;
   else 1) UPDATE1,
        2) branch to B.
```

B. The Discrete Case

Let $F(\cdot)$ be a PH-distribution with representation $(\underline{\alpha},T)$ where T is the transition probability matrix for an m state Markov chain. The vector $(\underline{\alpha},\alpha_{m+1})$ is the initial probability vector for the chain. Let $Q^* = T + T^\circ A^\circ$ be the transition probability matrix for the Markov chain associated with the PH-renewal process with underlying distribution F. A realization of the superposition of r i.i.d. stationary renewal processes is fully characterized by specifying the number, X_n, of processes which renew at time epoch n for $n \geq 1$. X_n counts any instantaneous renewals that may occur.

As in the continuous case we begin the simulation by calculating the stationary vector, $\underline{\pi}$, of Q^*. The simplification of $\underline{\pi}Q^* = \underline{\pi}$ results in $\underline{\pi}$ being given by

$$\underline{\pi} = (\underline{\alpha}T^{-1}\underline{e})^{-1}\,\underline{\alpha}\,(I-T)^{-1} .$$

Let $\underline{\pi}' = (\underline{\alpha}T^{-1}\underline{e})\underline{\pi}$. The system

$$\underline{\pi}'(I-T) = \underline{\alpha}$$

is solved using Gauss-Seidel iteration and $\underline{\pi}$ is determined from

$$\underline{\pi} = \underline{\pi}'/(\sum_{i=1}^{m} \pi_i') .$$

Let $N_n(j)$ be defined as the number of processes in phase j at time epoch n for $1 \leq j \leq m+1$, $n \geq 0$. Using the alias method, we generate r variates from $\underline{\pi}$ to initialize $N_0(j)$, for $1 \leq j \leq m$ with $N_0(m+1) = 0$.

To proceed from the (n-1)st epoch to the nth, we need to generate $N_{n-1}(j)$ variates from the discrete density $q_j(\cdot)$ where

$$5) \quad q_j(\nu) = \begin{cases} T_{j\nu}\ , & \text{for } 1 \leq \nu \leq m \\ \\ T_j^\circ\ , & \text{for } \nu = m+1 \end{cases}$$

for $1 \leq j \leq m$. To that effect the alias method is used and the resulting summary of phases partially determines $N_n(j)$ for $1 \leq j \leq m+1$.

If $N_n(m+1) = \ell$ then ℓ processes have renewed. The number of times that each of these ℓ processes will then instantaneously renew is geometrically distributed with parameter $\gamma = 1 - \alpha_{m+1}$. Since the sum of ℓ geometric random variables is a negative binomial random variable, we generate X_n as a negative binomial variate with parameters ℓ (number of successes) and γ (probability of success).

The phases in which the ℓ processes will restart are determined by ℓ random variates generated from $\underline{\alpha}' = \underline{\alpha}/\gamma$. We generate ℓ such variates using the alias method and update $N_n(j)$ appropriately. By setting $N_n(m+1)$ equal to zero, we are now ready to begin the next iteration. The iterations are continued until all variates are generated.

The method used for negative binomial generation is described fully in Fishman [4]. By generating a variate, Y, from a gamma $(\ell,1)$ distribution and a variate, X, from a Poisson $(Y\gamma/(1-\gamma))$ distribution, we have that X is a variate from a negative binomial distribution with parameters ℓ and γ. Gamma generation is accomplished using Cheng's method [3]. Poisson variates are generated by an inverse distribution method utilizing the recursive property

$$\text{Poisson}(\lambda,x+1) = \text{Poisson}(\lambda,x) \cdot \frac{\lambda}{x+1} ,$$

$$\text{for } x \geq 0 .$$

Two vectors, N of dimension m and NEW of dimension $m+1$, are used in the discrete simulation. The jth element of each vector accounts for the number of processes in phase j. A summary of the discrete superposition algorithm to generate M variates, for $M \geq 1$, is as follows.

```
SUPER2
1] (Generate initial state configuration from $\underline{\pi}$)
   A) Using Gauss-Seidel compute $\underline{\pi}$.
   B) Compute the alias tables for $\underline{\pi}$.
   C) Generate $r$ variates from $\underline{\pi}$ initial-
      izing $N(j)$ for $1 \leq j \leq m$.
2] (Initialize)
   A) Compute alias tables for $q_i(\cdot)$ defined
      in (5), for $1 \leq i \leq m$.
   B) $NEW(j) \leftarrow 0$ for $1 \leq j \leq m+1$; $\gamma \leftarrow 1.0 -$
      $\alpha_{m+1}$.
   C) Compute alias tables for $\underline{\alpha}' = \underline{\alpha}/\gamma$.
3] (Generate random variates, $X_n$, for
   $1 \leq n \leq M$)
   A) $X_n \leftarrow 0$
   B) Repeat for $1 \leq j \leq m$. Generate $N(j)$
      variates from $q_j(\cdot)$ updating $NEW(i)$
      for $1 \leq i \leq m+1$.
   C) Generate $\omega$ from negative binomial
      $(NEW(m+1),\gamma)$.
   D) $X_n \leftarrow \omega$.
   E) Repeat for $1 \leq j \leq NEW(m+1)$.
       i) Generate $k$ from $\underline{\alpha}'$.
      ii) $NEW(k) \leftarrow NEW(k) + 1$.
   F) (Update for the next iteration)
        i) $n \leftarrow n+1$.
       ii) $N(j) \leftarrow NEW(j)$, for $1 \leq j \leq m$.
      iii) $NEW(m+1) \leftarrow 0$.
       iv) Branch to A.
```

At each time epoch n, the cost of generating variates from the discrete superposition process is the sum of three factors: generating r variates using the alias method, generating $\ell(n)$ variates to "restart" the $\ell(n)$ processes that have renewed, and one negative binomial variate. Let C_A and C_{NB} be the cost associated with the alias method and negative binomial variate generation respectively. In (0,T) the total cost of this simulation is C_T where

$$C_T = rTC_A + \sum_{n=1}^{T} \ell(n) \cdot C_A + TC_{NB} .$$

Remarks. A University of Delaware Technical Report containing documented computer programs and samples of numerical results is available upon request from the authors at the Department of Mathematical Sciences, University of Delaware, Newark, DE, 19711, U.S.A.

The authors acknowledge with deep appreciation the helpful comments of Professor B. Schmeiser of Purdue University who suggested improvements in the computer implementation of the algorithms.

REFERENCES

Ahrens, J. H. and U. Dieter, "Computer Methods for Sampling from the Exponential and Normal Distributions", Comm. ACM, 15, 10, Oct. 1972, pp. 873-882.

Albin, S. L., "On Poisson Approximations for Superposition Arrival Processes in Queues", Manage. Sci., (to appear).

Cheng, R. C. H., "The Generation of Gamma Variables with Non-Integral Shape Parameter", Appl. Statist., 26, 1, 1977, pp. 71-75.

Fishman, G., Principles of Discrete Event Simulation, John Wiley & Sons, New York, 1978.

Karlin, S. and H. M. Taylor, A First Course in Stochastic Processes, Academic Press, New York, 1975.

Kronmal, R. A. and A. V. Peterson, Jr., "On the Alias Method for Generating Random Variables from a Discrete Distribution", Statistical Computing, 33, 4, Nov. 1979, pp. 214-218.

Lewis, P. A. W., A. S. Goodman and J. M. Miller, "Pseudorandom Number Generator for the System/360", IBM Systems Journal, 8, 2, 1969, pp. 136-146.

Neuts, M. F., "Probability Distributions of Phase Type", in Liber Amicorum Prof. Emeritus H. Florin, Dept. Math., Univ. Louvain, Belgium, 1975, pp. 173-206.

Neuts, M. F., Matrix-Geometric Solutions in Stochastic Models--An Algorithmic Approach, The Johns Hopkins University Press, Baltimore, Maryland, 1981.

Neuts, M. F. and K. S. Meier, "On the Use of Phase Type Distributions in Reliability Modeling of Systems with a Small Number of Components", OR Spectrum, 2, 1981, pp. 227-234.

Neuts, M. F., "Renewal Processes of Phase Type", Nav. Res. Logist. Quart., 25, 1978, pp. 445-454.