Proceedings of the 1995 Winter Simulation Conference
ed. C. Alexopoulos, K. Kang, W. R. Lilegdon, and D. Goldsman

A SET OF EXTENSIONS TO THE SIMAN/ARENA SIMULATION ENVIRONMENT

Marcelo J. Torres
Peter W. Glynn

Department of Operations Research
Stanford University
Stanford, CA 94305-4022, U.S.A.

ABSTRACT

In this paper, we introduce a set of SIMAN/ARENA language constructs. These constructs are intended to add correlated variate generation and importance sampling capabilities to the SIMAN/ARENA simulation environment. It is our purpose to demonstrate that such enhancements are feasible in virtually all simulation languages. These extensions are useful in the analysis of rare events and/or in systems driven by processes exhibiting some degree of correlation. Models arising in communications, manufacturing, physics, and biology may require this type of simulation capability. We describe how the constructs function, including activation, parameter specification, and behavior. In the case of the correlated variate construct, we describe the Markov modulation used to generate our correlations. For the importance sampling construct, we discuss the class of "exponentially twisted" changes-of-measure implemented within our construct, and provide some explanatory theoretical background.

1 INTRODUCTION

The rapid evolution of simulation tools over the last few years has greatly improved the efficiency of the model implementation process. Simulation languages have been extended to allow for a wider variety of constructs from which models can be built. More recently, graphical interfaces to the model building process have become popular. In addition, templates, which provide simulation design environments tailored to a number of specialized application areas, have become indispensable to many practitioners. It is now common for simulation tools to include templates for manufacturing systems, communications, transportation, and other areas. These enhancements have helped reduce the learning curve, shorten the development period, and add greater value to the end result of the simulation process. The presence of these tools to aid model building has had a significant impact in the simulation world.

The field of output analysis for stochastic simulations has also seen major developments and improvements. Recent developments include powerful and efficient methods useful in the areas of rare event probability estimation, parametric sensitivity analysis, and stochastic system optimization. (See for example, Glynn & Iglehart 1989, Glynn 1987).

It is recognized that these ideas have not been implemented in high level simulation languages. As a result, applying these techniques when constructing simulation models becomes a complicated process. The work required to apply output analysis techniques may in some cases require model logic to be modified, data to be imported and/or exported from outside applications, and in the worst case, low level programming language code to be compiled and incorporated into the simulation. This is especially true when we speak of correlated variate generation models and certain variance reduction techniques.

With this in mind, it is our goal to provide a set of easy-to-use constructs within the SIMAN simulation language environment, thereby making implementation of these features more palatable. One set of constructs allows for easy invocation of correlated variate generation, using the Markov modulated model. The second extension we have developed allows us to apply a certain class of importance sampling "changes-of-measure" within the SIMAN/ARENA modeling environment.

Recent research work in the communications area has been a motivation for the implementation of these constructs. Anick et al. (1982), Kesidis et al. (1993), Stern and Elwalid (1991) as well as others have used Markov modulation to model the arrival of information to multiplexors, switches and processors in communication networks. Markov modulation is seen as a natural modeling technique in this setting because the rate at which information arrives to communication networks fluctuates randomly over time, often with a high degree of correlation. Estimation of buffer overflow probabilities in communication networks by the use of simulation has been discussed in Parekh and Walrand (1989), Chang et al. (1994), Nicola et al. (1994), and Shahabuddin (1994). To get reasonable confidence intervals for performance measures of interest, conventional simulations need to be run for prohibitive lengths of time. The aforementioned papers suggest that importance sampling can cut down on simulation time by several orders of magnitude.

We have abided by the following three rules while developing these constructs. First, the constructs must be easy to use, requiring only a minimal set of parameters to be specified. Secondly, they should not be tied to particular models or performance measures. Thirdly, the constructs adhere to the general block structure followed by the SIMAN simulation language.

The organization of the paper is as follows. The Markov modulated variate processes and corresponding constructs are presented in Section 2. Section 3 summarizes the theoretical foundations of importance sampling and presents the construct used to apply this methodology. The concluding section presents our recommendations for future work. The appendix provides complete specifications of the constructs as well as directions for access of the constructs via ftp and the World Wide Web.

2 CORRELATED VARIATE EXTENSION

The method we use to induce autocorrelations into sequences of variates relies upon a background discrete-time Markov chain (DTMC) $X = (X_n : n \geq 1)$ or continuous-time Markov chain (CTMC) $X = (X(t) : t \geq 0)$. In the discrete-time model, the behavior of X induces autocorrelations into a sequence of variates $(\beta_n : n \geq 1)$. When the n'th variate is generated, the system observes the state of the DTMC and generates a variate having distribution associated with the observed state. In the continuous-time model, the rate at which a counting process experiences jumps is modulated by the CTMC X. We require that X have finite state space in both versions.

Markov modulation has been widely used to model systems in which the inter-event times exhibit statistical dependencies. Similarly, it has also been used to model arrival processes whose arrival rates vary randomly in time. In this context, the source responsible for introducing entities into a system can be considered to be in one of several states at any given time. These point processes are referred to in the literature as Markov modulated arrival processes.

2.1 Discrete-Time Modulation

In this case, we are interested in generating a sequence of variates $(\beta_n : n \geq 0)$ whose distribution depends on the states visited by a discrete-time Markov chain $X = (X_n : n \geq 0)$. Let S be the state space of our Markov chain, $P = (P(x,y) : x, y \in S)$ be the transition matrix, and $\mu = (\mu(x) : x \in S)$ the initial distribution. X is fully characterized by the initial distribution μ and transition matrix P. Define, for each $x \in S$, a distribution function $F(x, \cdot)$. We say that the β_n's are discrete-time modulated by X if

$$P\{\beta_1 \leq t_1, ..., \beta_n \leq t_n | X\} = \prod_{i=1}^{n} F(X_i, t_i).$$

Consider the event-stationary version of the discrete-time process $(\beta_n : n \geq 0)$; this amounts to starting X according to its stationary distribution $\pi = (\pi(x) : x \in S)$. When fitting Markov modulated models to real world data, one may want to consider the mean, variance, and k-step autocovariances associated with this stationary sequence. These can be computed from our model as follows:

$$E_\pi \beta_n = \sum_{x \in S} \pi(x)\nu(x)$$

$$var_\pi \beta_n = \sum_{x \in S} \pi(x)(\sigma^2(x) + \nu^2(x)) - (E_\pi \beta_n)^2$$

$$cov_\pi(\beta_0, \beta_n) = \sum_{x,y \in S} \pi(x)(P^n(x,y) - \pi(y))\nu(x)\nu(y)$$

where $\nu(x) = E(\beta_0 | X_0 = x)$, $\sigma^2(x) = var(\beta_0 | X_0 = x)$, and $(P^n(x,y) : x, y \in S)$ are the n-step transition probabilities of X.

At this point it is worth emphasizing that this variate generation model can be used in any of several ways. As indicated above, one may choose to have $\beta_0, \beta_1, \beta_2, ..., \beta_n$ represent inter-event times; for instance, inter-arrival times of entities to a system or consecutive service times at a processing unit. However, one could also choose this sequence to represent the number of arrivals at times $k = 0, 1, 2, ...$ in a discrete-time simulation.

To be more specific, suppose we consider using this variate generation method to model arrivals to a communications network operating in discrete-time. Packets arrive to a terminal node in our network according to an on-off source model. The source experiences active periods and silent periods whose distributions are geometric with parameter p_{active} and

p_{silent} respectively. When active, the source produces a Poisson number of packets with mean λ in a unit of discrete-time. When silent the source produces no packets. Let us also consider the source to be in the silent state initially.

To construct the arrival process, the transition matrix P for the Markov chain is defined as:

$$P = \begin{bmatrix} p_{active} & 1 - p_{active} \\ 1 - p_{silent} & p_{silent} \end{bmatrix}.$$

The initial distribution is:

$$\mu = \begin{bmatrix} 0 & 1 \end{bmatrix}.$$

The variate mass functions $f_i(.)$ are defined as:

$$f_{active}(k) = \frac{e^{-\lambda}\lambda^k}{k!} \text{ for } k \geq 0$$

$$f_{silent}(k) = \begin{cases} 1 & \text{if } k = 0 \\ 0 & \text{otherwise.} \end{cases}$$

We will describe how the SIMAN extensions can be used to construct this arrival process model in Section 2.3.2.

2.2 Continuous-Time Modulation

Continuous-time modulation is inherently different from discrete-time modulation. Discrete-time modulation works directly with the variates generated (i.e. inter-event times, arrival quantities, etc.), while continuous-time modulation only works directly with cumulative counting processes. In discrete-time modulation models, the state of the modulating process changes only at event transition epochs; not so with continuous-time modulation models.

Describe below is a form of continuous-time modulation known as the Markov modulated Poisson process (MMPP). In this case, the instantaneous arrival rate is modulated by the state of a continuous-time Markov chain. When the Markov chain is in state x, arrivals occur according to a Poisson process with arrival rate $\lambda(x)$. Thus, the goal is to construct a Poisson-like counting process $(A(t) : t \geq 0)$ modulated by the CTMC $X = (X(t) : t \geq 0)$. For example, $A(t)$ might represent the cumulative number of arrivals over $[0, t]$.

Let S be the state space of our Markov chain, $Q = (Q(x, y) : x, y \in S)$ be the associated generator of X, and $\mu = (\mu(x) : x \in S)$ the initial distribution. Define $\Lambda = (\lambda(x) : x \in S)$ to be the set of instantaneous arrival rates corresponding to the different states in the state space of the Markov chain. Specifically, it is assumed that

$$P\{A(t+h) - A(t) = 0|X\} = 1 - \lambda(X(t))h + o(h),$$

$$P\{A(t+h) - A(t) = 1|X\} = \lambda(X(t))h + o(h),$$

and

$$P\{A(t+h) - A(t) \geq 2|X\} = o(h),$$

as $h \downarrow 0$.

Suppose $N = (N(t) : t \geq 0)$ is a unit rate Poisson process, independent of X. We can construct an MMPP $A = (A(t) : t \geq 0)$ by setting $A(t) = N(\Gamma(t))$ where $\Gamma(t) = \int_0^t \lambda(X(s))ds$. For a more in depth treatment of this material, including moments and conditional moments for inter-arrival times, see Fischer, Meier-Hellstern (1993).

2.3 Invoking the Constructs

Our correlated variate enhancements include three new SIMAN constructs. The first two deal with the discrete-time model, while the third modulates in continuous-time. The first construct is used to generate a general sequence of autocorrelated variates. The second construct is used to introduce entities into a discrete-time simulation where the batch sizes introduced at each unit of discrete-time are autocorrelated. The third construct enables the user to introduce entities into a simulation according to a MMPP. For full specifications of these constructs see Appendix A.

The parameters used to determine our Markov chain, the variate distributions in the case of the first two constructs, and the arrival rates in the case of the last construct are specified in the experiments file. We suggest using the EXPRESSIONS element to define these values. The PARAMETERS element can also be used to specify the transition matrices, generators, and/or initial distributions but we have found the EXPRESSIONS element to be more suited to this task. More information on these built in SIMAN elements can be found in Appendix E of Pegden et al. (1995) or in the *SIMAN V Reference Guide.*

2.3.1 General Variate Generation: MMDT

In the following example, suppose we would like to generate a sequence of random variables modulated by a three state discrete-time Markov chain with state space $S = \{1, 2, 3\}$. The transition matrix for this Markov chain is

$$P = \begin{bmatrix} .2 & .4 & .4 \\ .5 & .5 & .0 \\ .2 & 0 & .8 \end{bmatrix}$$

and the initial distribution is

$$\mu = \begin{bmatrix} .5 & .5 & 0 \end{bmatrix}.$$

We choose state 1 to be associated with a uniform distribution over the interval [5, 10], state 2 with the constant 5, and state 3 with an exponential distribution of mean 10.

The following "expressions" can be used in the experiment file to define the appropriate parameters.

```
EXPRESSIONS:PM(9), .2, .4, .4, .5,
            .5, 0, .2, 0, .8:
            mu(3), .5, .5,  0:
            ProcessTime(3),
            UNIFORM(5,10),
            5,
            EXPONENTIAL(10);
```

Note that a one dimensional array with $|S|^2$ elements is used to represent our transition matrix. The constructs require transition matrices and generators to be specified in a one-dimensional array format.

The above values are referenced in the model file by the MMDT construct in the same fashion that built-in SIMAN blocks or elements reference EXPRESSIONS. For example, we can choose the variate sequence to be used by a delay block in a simulation model. This would be accomplished as follows.

```
DELAY:  MMDT(1, PM, u, ProcessTime);
```

The first argument to the construct is an identifier which can be a number or a name. The second argument, **PM**, identifies the expression representing the transition matrix. The parameter **u** references the initial distribution vector. **ProcessTime**, the last argument, is the collection of distribution functions.

2.3.2 Discrete-Time Arrival Process Construct: MMDTCREATE

Now, suppose we would like to use our Markov modulated model to generate arrivals to our system in discrete-time. SIMAN introduces entities into the simulation through the use of the CREATE block or the ARRIVALS block. We have created a construct named MMDTCREATE that operates exactly like the CREATE block, only our construct introduces batches of entities whose batch sizes are modulated by a DTMC.

Consider the communications network example developed earlier in this section. The parameters can be specified in the experiment file as follows:

```
EXPRESSIONS:PM(4), .95, .05, .01, .99:
            mu(2), 1, 0:
            Batching(2), Poisson(10), 0;
```

The MMDTCREATE block can be invoked in the model file as follows:

```
MMDTCREATE:1, PM, mu, Batching;
```

The parameter list passed to the MMDTCREATE construct is identical in form to that of the MMDT construct. Note that the set of distributions passed to this construct can only assign mass to the non-negative integers. It is also worth menitioning that there are a few optional arguments which can be passed as well. These arguments and the appropriate syntax for all the constructs are described in Appendix A.

2.3.3 MMPP Construct: MMPPCREATE

The MMPPCREATE construct follows the same argument form used by our previous two constructs. Only now, the first argument is the generator Q in a one-dimensional array format, and the third argument is an array of arrival rates. The following is one possible valid EXPRESSION specification and construct invocation.

```
EXPRESSIONS:QM(9), -10, 6, 4, 1.5,
            -2.5, 1, 0, 5, -5:
            mu(3), 1, 0:
            Rate(3), 100, 10, 25;
```

```
MMPPCREATE:1, QM, mu, Rate;
```

3 IMPORTANCE SAMPLING EXTENSION

Importance sampling is a variance reduction technique that can be applied to a broad class of stochastic simulations. It is especially useful in "rare event" simulation contexts. The idea is to modify the stochastic dynamics of the system, in order to cause the rare event to occur more frequently. To compensate for the fact that the dynamics have been altered, one needs to compute a quantity known as the likelihood ratio. Applying the likelihood ratio to the performance estimates derived from the simulation adjusts for the altered dynamics, and provides estimators that are valid for the original system.

For example, one might be interested in computing buffer overflow probabilities in a queueing context. In order to increase the likelihood of observing such a buffer overflow event, one might alter the stochastic dynamics of the arrival sequence, so that the arrival rate is increased. In other words the interarrival distribution could be modified to increase the arrival rate. In a Poisson arrival setting, this would just amount to increasing the arrival rate of the Poisson process. Altering such a distribution is known, in the importance sampling literature, as a "change-of-measure." In fact, it turns out that in the buffer overflow context just described, one generally wants to modify both the inter-arrival distributions and the processing time distribution. This is typical of discrete-event simulation applications of importance sampling; some or all of the basic "building block" distributions underlying the simulation may be altered, so as to force the rare event of interest to occur more frequently. The "building block" distributions include both the event scheduling distributions and, in some simulation settings, the distributions that are used to route entities through the system.

As indicated earlier, it is critical, in applying importance sampling, to compute the corresponding likelihood ratio. A key component of the constructs described below is an associated computation of the corresponding likelihood ratio. It turns out that the overall likelihood ratio for a given simulation is just the product of the likelihood ratios associated with each altered distribution. As a consequence, it is sufficient that our construct compute only the likelihood ratio associated with each individual distribution. A formal treatment of these ideas is presented in Glynn & Iglehart (1987).

3.1 Change-of-Measure Via Exponential Twisting

The choice of appropriate change-of-measure in a discrete-event simulation setting is a difficult problem. A number of theoretical papers on the subject have suggested using an appropriate "exponential twisting" of the distributions embedded in our simulation model. For instance, let us suppose our simulation model requires us to generate a variate X with distribution F. We can embed F in a parametric family of distributions $\{F(\theta)\}$ if we let

$$F_\theta(dx) = F(dx)\exp(\theta x - \psi(\theta)) \qquad (1)$$

where $\psi(\theta) = \log E\exp(\theta X)$ is the cumulant generating function of the r.v. X. The theory of "large deviations" makes clear that such "exponential twists" are often asymptotically optimal, in the sense that the variance of our estimator is minimized; see, for example, Billingsley (1979) and Bucklew (1990).

It turns out that many families of distributions are closed under "exponential twisting" in the sense that F_θ is of the same parametric form as is F. For continuous random variables, the exponential, gamma, and normal distributions belong to this group. In addition, the Poisson, Bernoulli, binomial, and geometric distributions are discrete random variables exhibiting this property. Furthermore, a similar method can be used to accomplish an exponential change-of-measure for the Markov modulated processes described in Section 2.

For example, let us show that the family of exponential distributions is closed in the above sense. The distribution function of interest is:

$$F(dx) = \lambda e^{-\lambda x} dx.$$

Our twisted distribution is:

$$\begin{aligned} F_\theta(dx) &= \lambda e^{-\lambda x} e^{\theta x} \left(\frac{\lambda}{\lambda - \theta}\right)^{-1} dx \\ &= (\lambda - \theta) e^{-(\lambda-\theta)x} dx, \end{aligned}$$

which (for real values of $\theta < \lambda$) is the distribution of an exponential random variable with parameter $(\lambda - \theta)$.

Similarly, we can derive this closure property for the binomial distribution. Let N be a binomial random variable with parameters p and n. Then:

$$P[N = k] = \binom{n}{k} p^k (1-p)^{n-k}.$$

Now, consider the twisted mass function,

$$\frac{P[N = k]e^{\theta k}}{Eexp(\theta N)} = \frac{\binom{n}{k} p^k (1-p)^{n-k} e^{\theta k}}{(1 - p + pe^\theta)^n}$$

$$= \binom{n}{k} \left(\frac{pe^\theta}{1 - p + pe^\theta}\right)^k \left(\frac{1-p}{1 - p + pe^\theta}\right)^{n-k},$$

which, for all real values of θ, is the probability mass function of another binomial random variable.

Here are the exponentially twisted parametric distributions implemented within the SIMAN/ARENA importance sampling simulation environment:

A. Poisson(λ) The distribution of the twisted Poisson random variable with mean λ is that of a Poisson random variable with mean λe^θ.

B. Discrete($p_1, v_1, p_2, v_2, p_3, v_3, \ldots, p_n, v_n$) The mass function of SIMANs built in empirical Discrete distribution is twisted exactly as we have specified above. Suppose N is an exponentially twisted random variable coming from this discrete distribution. The new probability N takes on the value v_i is $(p_i - p_{i-1})e^{\theta v_i}/Ee^{\theta N}$. Note that the p_i's represent cumulative distributions in SIMANs discrete distribution construct.

C. Exponential(λ) As shown above, an exponential with parameter λ when twisted becomes another exponential with parameter $(\lambda - \theta)$.

D. Gamma(α,β) The derivation of the twisted gamma distribution is much like that of the exponential. A Gamma(α,β) r.v. when twisted becomes a Gamma($\alpha,\beta/(1-\theta\beta)$).

E. Normal(μ,σ^2) The distribution of a twisted Normal random variable with mean μ and variance σ^2 is that of a normal random variable with mean $(\mu+\sigma^2\theta)$ and variance σ^2.

F. MMDT Process(P,F,μ) Using a slightly different approach, twisting the dynamics of the discrete-time Markov modulated process amounts to altering the transition matrix $P=(P(x,y):x,y\in S)$, as well as twisting the collection of distributions $F=(F(x,\cdot):x\in S)$ in the standard way described by Equation (1). Define $\phi_\theta(x)=E(e^{\theta\beta_0}|X_0=x)$. The new transition matrix $P_\theta=(P_\theta(x,y):x,y\in S)$ is derived by setting

$$P_\theta(x,y)=\frac{\phi_\theta(y)}{sp(\Phi_\theta)}\frac{h_\theta(y)}{h_\theta(x)}P(x,y)$$

where $\Phi_\theta=(\phi_\theta(x,y):x,y\in S)$ is a matrix with $\phi_\theta(x,y)=P(x,y)\phi_\theta(y)$, $sp(\Phi_\theta)$ its maximal eigenvalue, and $h_\theta=(h_\theta(x):x\in S)$ the corresponding right eigenvector.

G. MMPP(Q,Λ,π_0) This exponential change-of-measure is accomplished by modifying the generator Q. Set $\Lambda=diag(\lambda(x):x\in S)$. Let ρ_θ be the largest eigenvalue of the matrix $(Q+(e^\theta-1)\Lambda)$ and $h_\theta=(h_\theta(x):x\in S)$ the corresponding right eigenvector. The new generator $Q_\theta=(Q_\theta(x,y):x,y\in S)$ is related to Q via

$$\begin{aligned}Q_\theta(x,x)&=Q(x,x)+(e^\theta-1)\lambda(x)-\rho_\theta\\ Q_\theta(x,y)&=\frac{h_\theta(y)}{h_\theta(x)}Q(x,y)\qquad\text{for } x\neq y.\end{aligned}$$

Related derivations and theoretical results on the effect achieved by twisting Markovian models like those assumed by our Markov modulated constructs, see Andradottir et al. (1995), Chang et al. (1994), Kesidis et al. (1993), and Shahabuddin (1994).

3.2 General Change-of-Measure

In principal, any distribution having finite or compact support is one whose measure can be twisted in the fashion described by our previous section. The implementation of algorithms to generate random variables coming from general twisted distributions are notably more complex. Some suggested algorithms for Weibull and truncated Normal random variables are provided in Nakayama (1991).

In order to keep implementation of the constructs manageable, we have limited support of the built-in exponential change-of-measure feature in our importance sampling construct to include only those variates listed in the previous section.

In addition to supporting exponential changes-of-measure, our SIMAN/ARENA importance sampling extension can be used to implement general changes-of-measure. When invoking the importance sampling in this fashion, the user must specify the new distribution of the variate as well as the original distribution. The only restriction we impose on the change-of-measure is that the support of the original distribution must be a subset of the support of the simulated distribution.

3.3 Invoking the Construct

The name of our importance sampling construct is `IS`. As noted earlier, the construct can be invoked in one of two different fashions: the exponential change-of-measure and the general change-of-measure. When invoking the importance sampling construct with the exponential change-of-measure feature, the user must specify the original distribution as well as the parameter θ. The general change-of-measure feature requires the specification of the original distribution and the new distribution. For complete specifications see Appendix A.

The following SIMAN code shows how the construct can be invoked to alter the exponential distribution of the delay experienced by an entity in a simulation model.

```
Delay: IS(EXPO(2.0), .05);
```

The construct is used as the argument to the DELAY block. Note how two parameters have been passed; the first parameter being the original exponential distribution with a mean of 2.0, the second parameter being the value of θ used to determine the exponential change of measure. We have taken the original delay distribution and altered it so that its new mean is 1.95.

Now we illustrate the second method by which `IS` can be invoked. Suppose we wish to apply importance sampling by replacing a Weibull delay distribution with that of an exponential. In this case, instead of specifying a constant value θ for the second argument, we specify an entire distribution as follows:

```
Delay: IS(WEIB(2,4), EXPO(2.0));
```

The simulation will use the exponential distribution to drive this delay, but it will compute a likelihood ratio accordingly so that performance estimates can be made for the system driven by Weibull distributed delays.

3.4 Interpreting the Output

The SIMAN simulation output summary provides point estimates of performance measures of interest. When the `IS` construct is invoked, the simulation summary will include a likelihood ratio statistic. In order to get a point estimate for a performance measure from the original simulation model, the user should multiply the likelihood ratio by the point estimate output associated with the modified model.

4 RECOMMENDATIONS AND FUTURE WORK

The approach used to enhance the SIMAN simulation environment with a few powerful simulation features can be extended further. It is possible to extend the flexibility of variate generation capabilities as well as output analysis techniques with similarly designed constructs. We are actively involved in creating further extensions similar to the ones described in this paper. We are also interested in developing a design framework by which these tools can be provided effectively in a simulation environment setting. Some of these ideas are described below.

In the area of variate generation, extensions that perform generation of TES sequences among other useful variate models are being explored. Another useful feature would be an input processing extension that, given a correlated variate model and data, determines the appropriate model parameters. For instance, one could envision that in a particular simulation model setting it is believed that a certain data set has been generated by a source resembling those described in our Markov modulation setting. A useful construct would be one that estimates the parameters of the hidden Markov model which modulates the source.

Gradient estimation constructs enabling parametric sensitivity analysis, estimation, and stochastic optimization would also allow for easy implementation of powerful simulation features. A well thought out construct would allow practitioners to exploit the power of this theory while being sheltered from some of the finer details. Efforts here could help open the door to new application areas for practitioners. By helping automate output analysis techniques and the application areas that use them, we may further encourage the use of well founded simulation modeling.

Our last recommendation for future work concerns the incorporation of output analysis constructs into templates. As mentioned earlier, application specific constructs are organized into template sets easily accessible through powerful graphical simulation environments. Developing a similar template idea for output analysis techniques would be of further help in broadening the usability of our constructs.

ACKNOWLEDGEMENTS

The authors acknowledge Shane Henderson who provided helpful suggestions and made the SNET library software area available to us. This work was supported by the Army Research Office under contract no. DAAL03-91-G-0319.

APPENDIX A: CONSTRUCT SPECIFICATIONS

MMDT(P,μ,F)

Parameters	The $\|S\| \times \|S\|$ transition matrix P specified in a one-dimensional array, format with each of the rows of P having non-negative elements summing to 1, μ an array with $\|S\|$ non-negative elements summing to 1, and F an array of SIMAN distributions. All parameters should be specified as expressions in the experiment file.

MMDTCREATE,Offset:P,μ,F,maxbatches

Operand	Description	Default
Offset	Time of first creation	Begin
P	See MMDT	none
μ	See MMDT	none
F	Valid distributions are non-negative and integer	none
max batches	Maximum number of batches to be created.	infinite

MMPPCREATE,Offset:Q,μ,Λ,maxbatches

Operand	Description	Default
Offset	See MMDTCREATE	Begin
Q	Generator matrix Q, must have non-negative off-diagonal elements ($q(x,y) : x \neq y$) with the diagonal elements $q(x,x) = -\sum_{y \neq x} q(x,y)$	none
μ	See MMDT	none
Λ	An $\|S\|$ dimensional array of non-negative real numbers	none
max batches	Maximum number of batches to be created.	infinite

IS(Dist1,θ) or IS(Dist1, Dist2)

Parameters	Dist1 (and Dist 2) must come from one of the following SIMAN built-in distributions: Discrete, Erlang, Exponential, Gamma, Normal, or Poisson. Valid distribution parameters must be passed as well. To specify an exponential change-of measure, a real number is passed as the second argument. To specify a general change of measure, a distribution is passed.

APPENDIX B: DOWNLOAD INSTRUCTIONS

Information regarding the work done in this paper can be found at the SNET site maintained by the Operations Research department at Stanford University. The software and specifications are kept in the software area found at SNET. Information regarding specification modifications, additions, and changes will also be posted here.

via ftp connect to ftp-or.stanford.edu and download the file /ftp/simpaper/public_html/software /SIMAN/ISMMP.tar.

via www connect to http://www-or.stanford.edu /~simpaper/papers.html and navigate to the software area of the *SNET* library. The ISMMP.tar file can be downloaded automatically through a link on this web page.

Once, the tar file has been loaded, the appropriate files can be extracted on the users local host by executing the command **tar -xf ISMMP.tar**.

REFERENCES

Andradottir, S., D. P. Heyman, and T. J. Ott. 1995. On the Choice of Alternative Measures in Importance Sampling with Markov Chains. *Operations Research* 43:509–519.

Anick, D., D. Mitra, and M. M. Sondhi. 1982. Stochastic theory of a data handling system with multiple sources. *Bell System Technical Journal* 61:1871–1894.

Billingsley, P. 1979. *Probability and Measure*. New York: John Wiley.

Bucklew, J. A. 1990. *Large Deviation Techniques In Decision, Simulation, and Estimation*. New York: John Wiley.

Chang, C. S., P. Heidelberger, S. Juneja, and P. Shahabuddin. 1994. Effective bandwidth and fast simulation of ATM intree networks. *Performance Evaluation* 20:45–65.

Fischer, W., and K. Meier-Hellstern. 1992. The Markov-modulated Poisson process (MMPP) cookbook. *Performance Evaluation* 18:149–171.

Glynn, P. W., and D. L. Iglehart. 1989. Importance Sampling for Stochastic Simulations. *Management Science* 35:1367–1393.

Glynn, P. W. 1987. Likelihood Ratio Gradient Estimation: An Overview. In *Proceedings of the 1987 Winter Simulation Conference*, ed. A. MacNair, K. J. Musselman, and P. Heidelberger, 366–375. Institute of Electrical and Electronics Engineers, Atlanta, Georgia.

Kesidis, G., J. Walrand, and C. S. Chang. 1993. Effective Bandwidths for Multiclass Markov Fluids and Other ATM Sources. *IEEE/ACM Transactions on Networking* 1:424–428.

Nakayama, M. 1991. Simulation of Highly Reliable Markovian and Non-Markovian Systems. Ph.D. Thesis, Department of Operations Research, Stanford University, Stanford, California.

Nicola, V. F., G. A. Hagesteijn, and B. G. Kim. 1994. Fast Simulation of the Leaky Bucket algorithm. In *Proceedings of the 1994 Winter Simulation Conference*, ed. J. D. Tew, M. S. Manivannan, D. A. Sadowski, and A. F. Seila, 266–273. Institute of Electrical and Electronics Engineers, Lake Buena Vista, Florida.

Parekh, S., and J. Walrand. 1989. A Quick Simulation Method for Excessive Backlogs in Networks of Queues. *IEEE Transactions on Automatic Control* 34:54–66.

Pegden, C. S., R. E. Shannon, and R. P. Sadowski. 1994. *Introduction to Simulation Using SIMAN*. New York: McGraw Hill.

Shahabuddin, P. 1994. Fast Simulation of Packet

Loss Rates in Communication Networks with Priorities. In *Proceedings of the 1994 Winter Simulation Conference*, ed. J. D. Tew, M. S. Manivannan, D. A. Sadowski, and A. F. Seila, 399–405. Institute of Electrical and Electronics Engineers, Lake Buena Vista, Florida.

Stern, T. E., and A. I. Elwalid. 1991. Analysis of Separable Markov-Modulated Rate Models for Information-Handling Systems. *Advances in Applied Probability* 23:105–139.

AUTHOR BIOGRAPHIES

MARCELO J. TORRES is a graduate student working towards his Ph.D. degree at Stanford University. His research interests include performance modeling of communication networks, queueing theory, and simulation. He is currently involved in modeling proposed ATM network traffic management systems.

PETER W. GLYNN received his Ph.D. from Stanford University, after which he joined the faculty of the Department of Industrial Engineering at the University of Wisconsin-Madison. In 1987, he returned to Stanford, where he currently holds the Thomas Ford Faculty Scholar Chair in the Department of Operations Research. He was a co-winner of the 1993 Outstanding Simulation Publication Award sponsored by the TIMS College on Simulation. His research interests include discrete-event simulation, computational probability, queueing, and general theory for stochastic systems.