Proceedings of the 2008 Winter Simulation Conference
S. J. Mason, R. R. Hill, L. Mönch, O. Rose, T. Jefferson, J. W. Fowler eds.

STATISTICAL ANALYSIS OF SIMULATION OUTPUT

Marvin K. Nakayama

Computer Science Department
New Jersey Institute of Technology
Newark, NJ 07102, U.S.A.

ABSTRACT

We discuss methods for statistically analyzing the output from stochastic discrete-event or Monte Carlo simulations. Terminating and steady-state simulations are considered.

1 INTRODUCTION

So you've finally finished developing your simulation model! You spent countless hours developing an understanding of the underlying processes, collecting data, fitting the data to various probability distributions, and coding and debugging your simulation program. You carefully selected a performance measure you felt was appropriate to evaluate the system, and your program outputs an estimate of this measure. You then ran the simulation program once, and the results seemed to indicate that if the system design in your program was actually put into practice, it would perform well. You showed your boss the results, who then gave you the green light to implement this system design. However, once the system was in place, it performed poorly, not at all like the results that you obtained from your one simulation run. What went wrong?

Example 1 Figure 1 shows a (simplistic) model of a manufacturing system. Jobs arrive at the system at random times according to some stochastic process. After arriving, jobs require processing sequentially on two machines, labeled 1 and 2. Immediately after completing processing on Machine 1, a job goes to Machine 2, and a job leaves the system after the processing at Machine 2. The processing times on Machine 1 (resp., Machine 2) are independent random variables with distribution G_1 (resp., G_2). A job's processing times on the two machines are independent. If a job arrives at a machine that is currently busy, the job enters a first-in-first-out (FIFO) queue with unlimited capacity. We are interested in computing μ, the long-run average flow time of jobs, where the flow time of a job is the time that elapses between its arrival to and departure from the system. We ran one simulation of 1000 jobs, and we averaged the flow times of all the jobs to obtain an estimate $\hat{\mu} = 40.1$ of μ. (In this example, we took the arrival process to be a Poisson process with rate $1/10$, so the interarrival times are independent and exponentially distributed with mean 10, i.e., their distribution function is $F(x) = 1 - e^{-\lambda x}$ for $x \geq 0$, where $\lambda = 1/10$. Also, for $i = 1,2$, the processing time on machine i has distribution function $G_i(x) = 1 - \exp(-(x/\beta_i)^{\alpha_i})$ for $x \geq 0$, i.e., G_1 (resp., G_2) is Weibull with shape parameter $\alpha_1 = 4$ (resp., $\alpha_2 = 3$) and scale parameter $\beta_1 = 10$ (resp., $\beta_2 = 9$). All time units are minutes.)

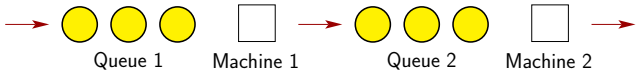

Figure 1: A manufacturing system.

Many simulations include randomness, which can arise in a variety of ways. For example, in our example of a manufacturing system, the processing times required at each machine follows a given probability distribution and the arrival times of new jobs are stochastic. In a simulation of an automatic teller machine (ATM) at a bank, customers arrive at random times and the amount of time each customer occupies the ATM is stochastic. Future returns of risky assets (e.g., stocks) in financial simulations are often modeled as random variables. Because of the randomness in the components driving a simulation, its output is also random.

Example 1 (continued) We ran the manufacturing simulation 5 different times, with each run consisting of 1000 jobs, and we obtained the following 5 estimates of μ from the runs: 40.1, 87.8, 54.2, 60.4, and 74.1. Thus, we see that the estimate 40.1 from the first simulation run seems to be unusually low.

The randomness of simulation output requires that it be analyzed using statistical techniques. The data-analysis methods taught in introductory statistics courses typically assume that the data are independent and identically dis-

978-1-4244-2708-6/08/$25.00 ©2008 IEEE

tributed (i.i.d.) with a normal distribution, but the output data from simulations are often not i.i.d. normal. For example, consider the flow times of the jobs in our manufacturing system example. If one job has an unusually long flow time, then the next job probably also will, so the flow times of the two jobs are dependent. Moreover, the first job in the system does not have to wait before being processed, whereas other jobs may have to wait, so their flow times are not identically distributed. Finally, flow times are always nonnegative and often skewed to the right, so flow times are not normally distributed. For these reasons one often cannot analyze simulation output using the classical statistical techniques developed for i.i.d. normal data.

In this tutorial, we will examine some statistical methods for designing and analyzing simulation experiments. In the next section we begin by distinguishing between two types of performance measures: terminating (or transient) and steady-state (or infinite-horizon or long-run). These two types of measures require different statistical techniques to analyze the results, and Section 3 reviews methods for analyzing output from terminating simulations, while Section 4 covers techniques for steady-state simulations. In Section 5 we discuss the estimation of multiple performance measures, and Section 6 briefly covers other methods useful for analyzing simulation output. Some concluding remarks are given in Section 7. The current paper is a modification of Nakayama (2006).

2 PERFORMANCE MEASURES

One of the first steps in any simulation study is choosing the *performance measure(s)* to compute. In other words, what measures will be used to evaluate how "good" the system is? For example, the performance of a queueing system may be measured by its expected number of customers served in a day, or we may use the long-run average daily cost as a measure of the performance of a supply chain.

There are primarily two types of performance measures for stochastic systems, which we now briefly describe:

1. *Transient performance measures*, also known as *terminating* or *finite-horizon* measures, evaluate the system's evolution over a finite time horizon.
2. *Steady-state performance measures* describe how the system evolves over an infinite time horizon. These are also known as *long-run* or *infinite-horizon* measures.

A simulation in which a transient (resp., steady-state) measure is estimated is called a *transient simulation* (resp., *steady-state simulation*). We now describe these concepts in more depth.

2.1 Transient Performance Measures

A *transient simulation* is one for which there is a "natural" event B that specifies the length of time in which one is interested for the system. The event B often occurs either at a time point beyond which no useful information is obtained, or when the system is "cleaned out." For example, if we are interested in the performance of a system during the first 10 time units of operation of a day, then B would denote the event that 10 time units of system time have elapsed. If we want to determine the first time at which a queue has at least 8 customers, then B is the event of the first time the queue length reaching 8. (See Law 2007, Section 9.3, for more details.)

Since we are interested in the behavior of the system over only a finite time horizon, the "initial conditions" $\mathscr{C}$ (i.e., conditions under which the system starts) can have a large impact on the performance measure. For example, queueing simulations often start with no customers present, which would be the conditions $\mathscr{C}$ in this setting.

In many transient simulations, the goal is to compute

$$\mu = E[X], \tag{1}$$

where X is a random variable representing the (random) performance of the system over some finite horizon and E denotes expectation (or average). If X is a discrete random variable that only assumes values $x_1, x_2, \ldots$, where the probability that X takes on value x_i is p_i, then $E[X] = \sum_{i=1}^{\infty} x_i p_i$. If X is a continuous random variable with density function f, then $E[X] = \int_{-\infty}^{\infty} x f(x)\,dx$.

We now examine some examples of transient performance measures.

Example 2 Consider the manufacturing example from before, and now assume that jobs are only accepted into the system between 9:00am and 5:00pm. Jobs arriving after 5:00pm are turned away, but any job arriving before 5:00pm is accepted and processed, even if the processing occurs after 5:00. Each day, the system starts with no jobs present and shuts down after all of the accepted jobs in the system have completed processing. All days are identically distributed. Let Z be the number of jobs completed in the first 4 hours of operation in a day, and we may be interested in determining the following transient performance measures:

- $E[Z]$, the expected value of Z. To put things in the framework of (1), we set $X = Z$.
- $P\{Z \geq 50\} = E[I(Z \geq 50)]$, which is the probability that at least 50 jobs are completed in the first 4 hours in a day, where $I(A)$ is the indicator function of an event A, which takes on the value 1 if A occurs, and 0 otherwise. In the notation of (1), we take $X = I(Z \geq 50)$ in this case.

The initial conditions $\mathscr{C}$ might be that the system starts out empty each day, and the terminating event B is that 4 hours of operation have completed.

2.2 Steady-State Performance Measures

Now we consider steady-state performance measures. Let $\mathbf{Y} = (Y_1, Y_2, Y_3, \ldots)$ be a (discrete-time) stochastic process representing the output of a simulation. For example, if the manufacturing system in our previous example runs continuously and accepts jobs 24 hours a day, then Y_i might represent the flow time of the ith job since the system first began operations. Let $F_i(y|\mathscr{C}) = P(Y_i \le y|\mathscr{C})$ for $i = 1, 2, \ldots$, where as before, $\mathscr{C}$ represents the initial conditions of the system at time 0. Observe that $F_i(\cdot\,|\mathscr{C})$ is the distribution function of the ith output Y_i given the initial conditions $\mathscr{C}$. We are now interested in the behavior of the system over an infinite time horizon, and it is often the case that the effects of the initial conditions $\mathscr{C}$ become negligible after a sufficiently long time has elapsed.

Definition 1 *If*

$$F_i(y|\mathscr{C}) \to F(y) \text{ as } i \to \infty \tag{2}$$

for all y and for any initial conditions $\mathscr{C}$, then $F(\cdot)$ is called the steady-state distribution *of the process* $\mathbf{Y}$. *If Y is a random variable with distribution F, we say that Y has the steady-state distribution, and we sometimes write (2) as $Y_i \xrightarrow{\mathscr{D}} Y$ as $i \to \infty$, which is read as "Y_i converges in distribution to Y."*

The interpretation of (2) is that for all i sufficiently large,

$$F_i(y|\mathscr{C}) \approx F(y), \text{ for all } y. \tag{3}$$

The value of i for which the above approximation holds depends very much on the particular system being simulated. Note that (3) does not mean that the *values* of the Y_i are all the same for large i, but rather that the *distribution* of Y_i (given the initial conditions $\mathscr{C}$) is close to F for large i. Indeed, the steady-state random variable Y (and also the Y_i for large i) may still exhibit plenty of variability. When Y is a random variable with distribution F, the expectation $E(Y)$ is a *steady-state performance measure*. It can be shown under great generality that $E(Y_i|\mathscr{C}) \to E(Y)$ as $i \to \infty$ for all initial conditions $\mathscr{C}$ when (2) holds. Let f_i (resp., f) be the density function corresponding to distribution $F_i(\cdot\,|\mathscr{C})$ (resp., F). Figure 2, which is a modification of one from Law (2007), gives an example of the density functions f_i approaching the limiting density f as i gets larger. Although Figure 2 seems to show that the limiting density is from a normal distribution, this is not necessarily the case.

Example 3 Consider the manufacturing system from before, but now suppose that it runs continuously and accepts jobs 24 hours a day. We now allow the distribution of the interarrival times of jobs to the system to vary over time, but in such a way that the system eventually reaches steady state. Let Y_i be flow time of the ith job to enter the system since the system first began operation, and suppose that over time, the flow times "settle down" into steady state; i.e., $Y_i \xrightarrow{\mathscr{D}} Y$ as $i \to \infty$. We now may be interested in determining the following steady-state performance measures:

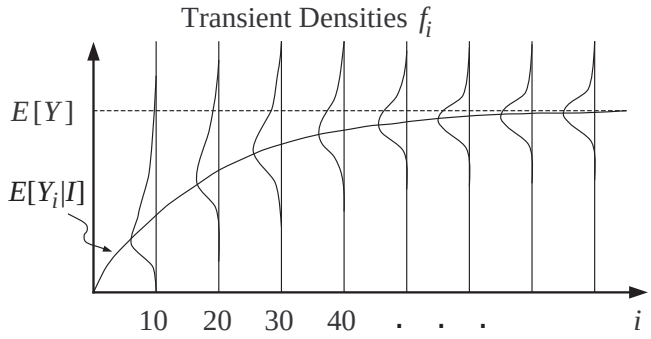

Figure 2: Densities f_i of an output process $(Y_1, Y_2, \ldots)$.

- $E[Y]$, which is the steady-state expected flow time of jobs;
- $P\{Y \ge 30\} = E[I(Y \ge 30)]$, which is the steady-state probability that a job's flow time is at least 30 minutes.

Again, we may let the initial conditions $\mathscr{C}$ denote that the system begins operations on the first day with no jobs present, and over time, the effects of the initial conditions typically "wash away."

Many systems do not have a steady state. For example, consider our previous example of a manufacturing system that only accepts jobs between 9:00am and 5:00pm. Let Y_i be the flow time of the ith job to arrive since the system was first installed. Then, the process $\mathbf{Y}$ does not have a steady state because the first job of each day always has no wait, whereas other jobs may have to wait. For example, suppose 500 jobs are processed on the first day, so day 2 begins with job 501, which has no wait since there are no jobs ahead of it on that day. Since this happens every day, (2) cannot hold. On the other hand, if the system runs continuously and jobs are accepted 24 hours a day, then a steady state may exist.

3 OUTPUT ANALYSIS FOR TRANSIENT SIMULATIONS

We now discuss how to analyze the output from a transient simulation. As we noted in (1), our goal is to calculate

$\mu = E[X]$, where X is a random variable representing the performance of the system over some finite horizon with initial conditions $\mathscr{C}$, and E denotes expectation. The basic approach to estimate μ using simulation is as follows: Generate $n \geq 2$ i.i.d. replicates of X, say $X_1, X_2, \ldots, X_n$, and form the (point) estimator

$$\bar{X}(n) = \frac{1}{n}\sum_{i=1}^{n} X_i, \tag{4}$$

which is the sample average of the n replicates. Under very mild conditions, the law of large numbers guarantees $\bar{X}(n) \approx \mu$ for large sample sizes n. Thus, the sample mean $\bar{X}(n)$ is a reasonable estimator of the true mean μ.

We generate i.i.d. replicates of X by running independent simulations of the system under study. We make the replicates independent by using non-overlapping streams of random numbers from the random-number generator. We ensure the replicates are identically distributed by starting each simulation using the same initial conditions $\mathscr{C}$ and using the same dynamics to govern the evolution of the system.

Example 2 (continued) Assume the manufacturing system only accepts jobs between 9:00am and 5:00pm, and let X be the number of jobs that are completed in the first 4 hours of operation in a day. We ran 50 independent replications of the system, with each replication covering the first 4 hours of operation in a day, and let X_i denote the number of jobs completed in the first 4 hours of operation in the ith replication. Our simulations yielded the values $X_1 = 23$, $X_2 = 16$, $X_3 = 19$, $\ldots$, $X_{50} = 24$. Thus, by (4) our point estimate of $\mu = E[X]$ is

$$\bar{X}(50) = \frac{1}{50}[23 + 16 + 19 + \cdots + 24] = 20.7.$$

We argued before that the sample mean $\bar{X}(n)$ is a reasonable estimator for the true mean μ, but how close is $\bar{X}(n)$ to μ? The central limit theorem (CLT) provides an answer. Specifically, let σ^2 denote the *variance* of X, i.e., $\sigma^2 = \mathrm{Var}(X) = E[(X-\mu)^2] = E[X^2] - \mu^2$, which we assume is positive and finite. We sometimes also refer to the *standard deviation* of X, which is $\sigma = \sqrt{\sigma^2}$. Let $N(0,1)$ denote a *standard normal* random variable (with mean 0 and variance 1), whose density function $f(x) = (1/\sqrt{2\pi})\exp(-x^2/2)$ is the familiar "bell-shaped curve" shown in Figure 3. The CLT asserts that for n large,

$$\frac{\sqrt{n}}{\sigma}(\bar{X}(n) - \mu) \stackrel{\mathscr{D}}{\approx} N(0,1), \tag{5}$$

where $\stackrel{\mathscr{D}}{\approx}$ means "has approximately the same distribution as." The approximation in (5) is usually reasonable for $n \geq 30$, and it becomes exact as $n \to \infty$.

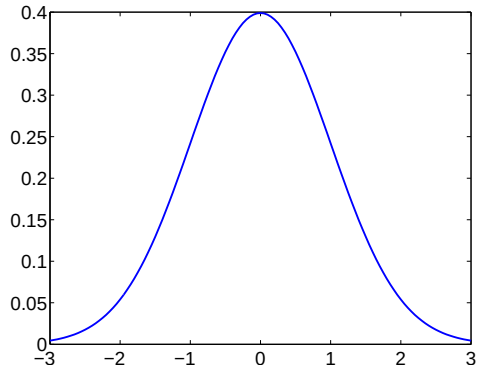

Figure 3: Density function of a standard normal random variable $N(0,1)$.

To illustrate the meaning of (5), we ran the following set of experiments. In each experiment we collected n i.i.d. samples $X_1, X_2, \ldots, X_n$ from an exponential distribution with mean $\mu = 1$ (so $\sigma = 1$); i.e., the density function of each X_i is $g(x) = e^{-x}$ for $x \geq 0$. From the n samples, we computed $Z = (\sqrt{n}/\sigma)(\bar{X}(n) - \mu)$, which is the left-hand side of (5). We then repeated this 10^3 independent times, each time computing Z from the n samples. We then plotted the 10^3 values of Z in a histogram. We repeated this whole process four different times, first with $n = 1$, then $n = 3$, $n = 10$ and $n = 50$. Figure 4 gives the resulting histograms. For $n = 1$, note that $\bar{X}(1)$ is the sample average of just a single exponential random variable, so the histogram for $n = 1$ looks like the density function of an exponential random variable. However, as n increases, the histogram starts resembling the bell-shaped curve in Figure 3, which is the density of the standard normal. Thus, we see the CLT taking effect for large sample sizes n.

We now use the CLT in (5) to derive a *confidence interval* for μ. First define a *confidence level* $1-\alpha$ with $0 < \alpha < 1$; typically, one chooses $\alpha = 0.1$, 0.05 or 0.01, so $1-\alpha = 0.9$, 0.95 or 0.99. Then, we look up in a normal table the constant $z \equiv z_{1-\alpha/2}$ for which $P\{N(0,1) \leq z\} = 1 - \alpha/2$. The *critical point* z is the value on the horizontal axis in Figure 3 such that the area under the curve to left of z is exactly $1-\alpha/2$. Virtually any introductory statistics book provides a normal table; also see Table T.1 of Law (2007) or Table A.3 of Banks et al. (2005). For example, $z = 1.65$ when $\alpha = 0.1$, $z = 1.96$ when $\alpha = 0.05$, and $z = 2.58$ when $\alpha = 0.01$. By the symmetry around 0 of the standard normal density function, $P\{N(0,1) > z\} = P\{N(0,1) < -z\} = \alpha/2$, so for large sample sizes n,

$$\begin{aligned} 1-\alpha &= P\{-z \leq N(0,1) \leq z\} \\ &\approx P\left\{-z \leq \frac{\sqrt{n}}{\sigma}(\bar{X}(n) - \mu) \leq z\right\} \end{aligned} \tag{6}$$

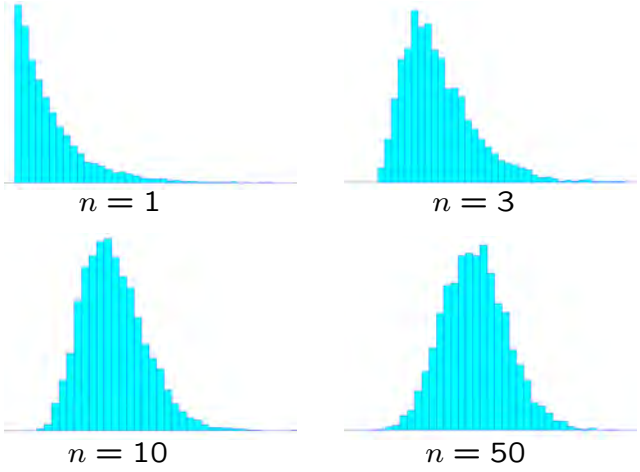

Figure 4: The histograms are constructed from 10^3 independent values of $Z = (\sqrt{n}/\sigma)(\bar{X}(n) - \mu)$ for different values of n, where $X_1, X_2, \ldots, X_n$ are i.i.d. exponential with mean $\mu = 1$.

$$= P\left\{ -\frac{z\sigma}{\sqrt{n}} \le \bar{X}(n) - \mu \le \frac{z\sigma}{\sqrt{n}} \right\}$$
$$= P\left\{ \mu \in \left[\bar{X}(n) \pm \frac{z\sigma}{\sqrt{n}} \right] \right\}, \quad (7)$$

where the approximation in (6) follows for large n from the CLT in (5). Thus, (7) implies that when n is large, the interval

$$\left[\bar{X}(n) - \frac{z\sigma}{\sqrt{n}},\ \bar{X}(n) + \frac{z\sigma}{\sqrt{n}} \right] \quad (8)$$

has roughly probability $1-\alpha$ of containing the true mean μ, and we call (8) an approximate $100(1-\alpha)\%$ *confidence interval* for μ.

One problem with the confidence interval in (8) is that the value of σ is typically unknown, so we need to estimate it. Since $X_1, X_2, \ldots, X_n$ are i.i.d., we can use classical statistics to do this. Let

$$S^2(n) = \frac{1}{n-1} \sum_{i=1}^{n} (X_i - \bar{X}(n))^2, \quad (9)$$

which is the *sample variance* of $X_1, \ldots, X_n$, and is an estimator of σ^2. The *sample standard deviation* $S(n) = \sqrt{S^2(n)}$ is an estimator of σ. The following variant of the CLT in (5) then holds:

$$\frac{\sqrt{n}}{S(n)} (\bar{X}(n) - \mu) \overset{\mathcal{D}}{\approx} N(0,1) \quad (10)$$

for large sample sizes n, where we replaced σ in (5) with $S(n)$. Similarly, replacing σ in (8) with $S(n)$ yields

$$\left[\bar{X}(n) - \frac{zS(n)}{\sqrt{n}},\ \bar{X}(n) + \frac{zS(n)}{\sqrt{n}} \right] \quad (11)$$

as an approximate $100(1-\alpha)\%$ confidence interval for μ when n is large; i.e.,

$$P\left\{ \mu \in \left[\bar{X}(n) - \frac{zS(n)}{\sqrt{n}},\ \bar{X}(n) + \frac{zS(n)}{\sqrt{n}} \right] \right\} \approx 1-\alpha. \quad (12)$$

Because of the extra approximation from using $S(n)$ to estimate σ, we now require a larger sample size n, say $n \ge 50$, than before in (5) and (8). We summarize our discussion in the following:

Procedure to construct confidence intervals for transient measure μ

1. Specify a confidence level $1-\alpha$ with $0 < \alpha < 1$ and a sample size n that is large. Also, look up in a normal table the value of z such that $P\{N(0,1) \le z\} = 1 - \alpha/2$. Typically, one chooses $\alpha = 0.1$, 0.05 or 0.01, and one should choose $n \ge 50$.
2. Generate n i.i.d. replicates $X_1, X_2, \ldots, X_n$ of X.
3. Using the n data points $X_1, X_2, \ldots, X_n$, calculate the sample mean $\bar{X}(n)$ using (4) and the sample variance $S^2(n)$ using (9).
4. Use (11) to construct an approximate $100(1-\alpha)\%$ confidence interval for μ.

Example 2 (continued) We previously computed the sample mean of our 50 data points $23, 16, 19, \ldots, 24$ as $\bar{X}(50) = 20.7$. Using (9), we get the sample variance to be

$$S^2(50) = \frac{1}{49}\left[(23-20.7)^2 + (16-20.7)^2 + \cdots + (24-20.7)^2 \right] = 8.9,$$

so an approximate 95% confidence interval for $\mu = E[X]$ is

$$\left[20.7 \pm \frac{1.96 \times \sqrt{8.9}}{\sqrt{50}} \right] = [20.7 \pm 0.8].$$

An interpretation of the approximate $100(1-\alpha)\%$ confidence interval for μ in (11) is that we are highly confident (i.e., approximately $100(1-\alpha)\%$ confident) that the true mean μ lies in the interval (11). Thus, a confidence interval provides a form of error bounds for our estimator $\bar{X}(n)$ of

μ. The *half width* H_n of the confidence interval in (11) is

$$H_n = \frac{zS(n)}{\sqrt{n}}, \tag{13}$$

i.e., the confidence interval in (11) is $\bar{X}(n) \pm H_n$. It can be shown that $S(n) \approx \sigma$ for large n, so as the sample size n increases, the half width decreases at rate $1/\sqrt{n}$. In particular, this means that to obtain one additional significant figure of accuracy (i.e., increase accuracy by a factor of 10), we need to increase the sample size n by a factor of 100. Thus, the estimator $\bar{X}(n)$ converges to μ rather slowly.

If we construct the confidence interval (11) using the above steps, (12) implies the probability is approximately $1-\alpha$ that the interval will contain μ. In other words, if we repeat these steps m independent times, this will give us m different confidence intervals. Some of them will contain (cover) μ, and others will not. The theory says that approximately $(1-\alpha)m$ of the m intervals should cover μ. For example, if we constructed $m = 1000$ independent 95% confidence intervals, we would expect that about 950 of them would contain μ, while about 50 would not. In practice, though, this does not always happen. The approximation in (12) only becomes exact as the sample size $n \to \infty$, so the coverage is only approximately $1-\alpha$ for large but finite n. The true value of the probability on the left-hand side of (12) is known as the *coverage*.

It would be nice to know when the approximation in (12) is good, and when it is not. It turns out that the quality of the CLT approximation in (10) is largely influenced by the value of the *skewness* of the random variable X that we are sampling, where the skewness is defined as $\gamma = E[(X-\mu)^3]/\sigma^3$ with μ the mean and σ the standard deviation of X. The skewness γ is a measure of the symmetry of the density (or probability mass) function of the random variable X. If the density of X is symmetric about its mean, then the skewness $\gamma = 0$. Asymmetrical densities typically lead to nonzero skewness. There are deep mathematical results showing that smaller skewness γ leads to a better CLT approximation in (10), which results in the approximation in (12) being more accurate. Thus, if the density of X is highly asymmetric (as is typical of queueing simulations), the CLT approximation is not so good, and the coverage of the confidence interval in (11) may be significantly less than its nominal value of $1-\alpha$. In fact, it is not unusual for confidence intervals that are supposed to have approximate 95% coverage to actually only have, say, 80% coverage. Law (2007), p. 236, provides more discussion on this issue.

3.1 Pre-specifying Confidence Interval Widths

In the previous section we discussed so-called *fixed-sample-size methods* for estimating a transient performance measure $\mu = E[X]$, where X represents the random performance of the system over some finite time horizon. These methods are so named because the sample size is fixed prior to running any simulations. However, before executing a simulation, we usually do not know how large the resulting half width (13) will be since we typically do not know the variance of X. In many situations, though, we would like to end up with an estimator with a small prespecified error ε, i.e., we want the $100(1-\alpha)\%$ confidence interval to be $\bar{X}(n) \pm \varepsilon$.

To achieve our goal of having a confidence interval with half width ε, we set the half-width H_n in (13) equal to ε and solve for n, yielding $n = (zS(n)/\varepsilon)^2$. This suggests that if we take n samples, where n is determined by (14), then the resulting confidence interval should have half width that is approximately ε. We can accomplish this using a two-stage procedure, where the first stage simulates n pilot runs to compute $S^2(n)$, and the second stage simulates

$$N = \left(\frac{zS(n)}{\varepsilon} \right)^2 \tag{14}$$

total replications.

Example 2 (continued) Suppose we want our estimate of μ to be within 0.5 of the correct value with confidence level 95%. Thus, we set $\varepsilon = 0.5$ in (14), and using the sample variance $S^2(50) = 8.9$ from our 50 previous replications gives

$$N = \frac{(1.96)^2 \times 8.9}{(0.5)^2} \approx 138.$$

Hence, we need a total of about 138 replications to obtain an estimate within 0.5 of the true value of μ (with 95% confidence).

4 OUTPUT ANALYSIS FOR STEADY-STATE SIMULATIONS

We now discuss the estimation of steady-state performance measures. There are two cases to consider:

1. Discrete-time process: $\mathbf{Y} = (Y_i : i = 1, 2, \ldots)$ is an output process with an integer-valued time index, and $Y_i \xrightarrow{\mathcal{D}} Y$ as $i \to \infty$, where Y has the steady-state distribution.
2. Continuous-time process: $\mathbf{Y} = (Y(s) : s \geq 0)$ is an output process with a continuous-valued time index, and $Y(s) \xrightarrow{\mathcal{D}} Y$ as $s \to \infty$, where Y has the steady-state distribution.

Our goal is to estimate (and produce confidence intervals for) ν, where $\nu = E[Y]$. Under great generality, we can show that the time-average of the process converges to ν;

i.e.,

$$\frac{1}{m}\sum_{i=1}^{m} Y_i \to \nu \tag{15}$$

as $m \to \infty$ for discrete-time processes, and

$$\frac{1}{t}\int_0^t Y(s)\,ds \to \nu \tag{16}$$

as $t \to \infty$ for continuous-time processes.

We previously saw in Section 2.2 some examples of steady-state measures for a discrete-time process. For example, Y_i could be the flow time of the ith job in a manufacturing system, so ν represents the steady-state expected flow time of jobs. We now give an example of a continuous-time process.

Example 4 Suppose that the manufacturing system from before operates continuously and accepts jobs 24 hours a day, and let $Y(s)$ denote the number of jobs in the system at time s. We define the continuous-time stochastic process $\mathbf{Y} = (Y(s) : s \geq 0)$, and assume that $\mathbf{Y}$ has a steady state; i.e., $Y(s) \xrightarrow{\mathscr{D}} Y$ as $s \to \infty$. Then we may be interested in calculating $\nu = E[Y]$, which typically can be rewritten as in (16). Thus, ν in this case is the long-run time-average number of jobs in the system. Another possible steady-state measure is $\nu = P(Y \geq 10) = E[I(Y \geq 10)]$, which we can typically express as

$$\nu = \lim_{t\to\infty} \frac{1}{t}\int_0^t I(Y(s) \geq 10)\,ds,$$

so ν is the long-run proportion of time that at least 10 jobs are in the system.

4.1 The Difficulties of Output Analysis of Steady-State Simulations

We will concentrate on discrete-time processes (continuous-time processes can be handled in a similar manner). Our goal is to estimate and produce confidence intervals for the steady-state parameter ν. First, we examine how to produce a point estimator for ν. As we can see in (15), the parameter ν typically can be viewed as the long-run average level of Y_i. Thus, if we set

$$\bar{Y}(m) = \frac{1}{m}\sum_{i=1}^{m} Y_i,$$

then

$$\bar{Y}(m) \approx \nu$$

for large sample sizes m. In other words, running a "long" simulation (i.e., taking m large) will result in an estimator $\bar{Y}(m)$ that is "close" to ν. Hence, the problem of producing a point estimator for ν is easily solved. However, the construction of a confidence interval for ν is more delicate, as we shall see.

For virtually all reasonably behaved systems possessing a unique steady state, one can show that a central limit theorem for $\bar{Y}(m)$ is valid; i.e., there exists a constant $\bar{\sigma}$ such that

$$\frac{\sqrt{m}}{\bar{\sigma}}(\bar{Y}(m) - \nu) \overset{\mathscr{D}}{\approx} N(0,1) \tag{17}$$

for m sufficiently large.

Definition 2 *The parameter $\bar{\sigma}^2$ is called the* time-average variance constant *of the steady-state simulation.*

Unfortunately, it is not so straightforward to use the CLT in (17) to construct a confidence interval for ν. The problem stems from the fact that it is a non-trivial matter to estimate $\bar{\sigma}$ (or equivalently $\bar{\sigma}^2$). The sample variance $S^2(n)$ in (9) used to estimate σ^2 in the transient-simulation setting is only valid for i.i.d. data. In steady-state simulations, $Y_1, Y_2, \ldots$ are typically not i.i.d. For example, as noted in Section 1, successive flow times of jobs in our example of a manufacturing system are often dependent and not identically distributed. Thus, we cannot use (9) applied to the $Y_1, Y_2, \ldots$ to estimate $\bar{\sigma}^2$.

4.2 Method of Multiple Replications

The *method of multiple replications* offers one escape from this difficulty of estimating $\bar{\sigma}$. Suppose that rather than simulating one long replicate of length m, we simulate r independent and identically distributed replications, each of length $k = m/r$. We should choose r small, say $10 \leq r \leq 30$, so that the length k of each replication is large. We need k large since we are interested in the long-run behavior of the process $\mathbf{Y}$. We achieve independence of the replications by using non-overlapping streams of random numbers for the different replications. We obtain identically distributed replications by starting each with the same initial conditions and using the same system dynamics to generate each replication. From each replication we get an estimate of ν by averaging the observations within the replication. Because the r estimates across the replications are independent, we can form a sample variance of the r values. This is the basic idea underlying the method of multiple replications.

Suppose that we have run r i.i.d. replications, each having run length k, and we arrange the output from all the simulations as follows:

	Simulation Output			
replication 1	$Y_{1,1}$	$Y_{1,2}$	$\cdots$	$Y_{1,k}$
replication 2	$Y_{2,1}$	$Y_{2,2}$	$\cdots$	$Y_{2,k}$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
replication r	$Y_{r,1}$	$Y_{r,2}$	$\cdots$	$Y_{r,k}$

The entries in the first row are the k observations from the first replication, the entries in the second row are the k observations from the second replication, and so on. Now let X'_j be the average of the entries in the jth row:

Rep	Simulation Output				Sample mean within rep
1	$Y_{1,1}$	$Y_{1,2}$	$\cdots$	$Y_{1,k}$	X'_1
2	$Y_{2,1}$	$Y_{2,2}$	$\cdots$	$Y_{2,k}$	X'_2
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
r	$Y_{r,1}$	$Y_{r,2}$	$\cdots$	$Y_{r,k}$	X'_r

(18)

where for each replication $j = 1,2,\ldots,r$,

$$X'_j = \frac{1}{k}\sum_{i=1}^{k} Y_{j,i}.$$

Thus, X'_1 is the average of the observations in the first row of (18), X'_2 is the average of the observations in the second row of (18), and so on. Since we ran i.i.d. replications, $X'_1, X'_2, \ldots, X'_r$ are i.i.d. observations with $E(X'_j) \approx \nu$ for each $j = 1,2,\ldots,r$, if k is sufficiently large, by virtue of (15). So we can use classical statistics to form a point estimator and confidence interval using the observations $X'_1, X'_2, \ldots, X'_r$. Specifically, let

$$\bar{X}'(r) = \frac{1}{r}\sum_{j=1}^{r} X'_j$$

and

$$S'^2(r) = \frac{1}{r-1}\sum_{j=1}^{r} (X'_j - \bar{X}'(r))^2$$

be the sample mean and sample variance, respectively, of the X'_j. Then, an approximate $100(1-\alpha)\%$ confidence interval for ν is given by

$$\left[\bar{X}'(r) - \frac{tS'(r)}{\sqrt{r}},\ \bar{X}'(r) + \frac{tS'(r)}{\sqrt{r}}\right], \tag{19}$$

where $t \equiv t_{r-1,1-\alpha/2}$ is chosen such that $P\{T_{r-1} \le t\} = 1-\alpha/2$ and T_{r-1} is a Student-t random variable with $r-1$ degrees of freedom. Virtually all introductory statistics books provide t-tables giving values of t for various α and degrees of freedom; also see Table T.1 of Law (2007) or Table A.5 of Banks et al. (2005). (Here we use the critical point from a t-distribution rather than a standard normal distribution because the number r of replications is often small.)

Example 1 (continued) We ran $r = 10$ independent replications, each consisting of $k = 1000$ simulated jobs, to estimate ν, the steady-state expected flow time of jobs. In each replication we averaged the flow times of all 1000 jobs in the replication. The resulting averages in the 10 replications were $X'_1 = 40.1$, $X'_2 = 87.8$, $X'_3 = 54.2$, $\ldots$, $X'_{10} = 59.8$. The average of the 10 values is $\bar{X}'(10) = 62.3$, which is our point estimate of ν. We also used (19) to compute an approximate 95% confidence interval for ν as follows. Since we simulated $r = 10$ independent replications, we use the 0.975 critical point from a t-distribution with $r-1 = 9$ degrees of freedom, so $t = 2.26$, which we obtained from a t-table. The sample variance of the 10 values is $S'^2(10) = 15.5$, so our confidence interval is

$$\left[62.3 \pm \frac{2.26 \times \sqrt{15.5}}{\sqrt{10}}\right] = [62.3 \pm 11.1].$$

A problem with the method of multiple replications is that, while the technique permits simple estimation of the variance, the multiple-replicate estimator $\bar{X}'(r)$ can be significantly contaminated by the presence of *initialization bias*. Specifically, the law of large numbers guarantees that for each replication j,

$$X'_j = \frac{1}{k}\sum_{i=1}^{k} Y_{j,i} \approx \nu$$

for large k. However, since each replicate is usually started with initial conditions $\mathscr{C}$ that are atypical of the steady state (e.g., queueing simulations are often started with no customers present), it often follows that for any finite k,

$$E\left[\frac{1}{k}\sum_{i=1}^{k} Y_{j,i}\right] \neq \nu.$$

Thus, we conclude that if the number of replicates r is large relative to the run length k of each replication, then the estimator $\bar{X}'(r)$ may be significantly biased by the initial conditions.

A partial solution to this problem is to use *initial-data deletion*, which we now describe. Suppose that we somehow can determine the first c observations of the simulation are significantly "contaminated," i.e., not very representative of steady state. Also, suppose all observations after the first c are "clean," i.e., not significantly contaminated. Then in each replication, we delete the first c observations when calculating the sample mean of the replication. Specifically, we arrange the output of all the replications as follows:

Rep	Simulation Output						Sample mean of clean data
	"contaminated"			"clean"			
1	$Y_{1,1}$	$\cdots$	$Y_{1,c}$	$Y_{1,c+1}$	$\cdots$	$Y_{1,k}$	X_1
2	$Y_{2,1}$	$\cdots$	$Y_{2,c}$	$Y_{2,c+1}$	$\cdots$	$Y_{2,k}$	X_2
$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
r	$Y_{r,1}$	$\cdots$	$Y_{r,c}$	$Y_{r,c+1}$	$\cdots$	$Y_{r,k}$	X_r

where for each replication $j = 1, 2, \ldots, r$,

$$X_j = \frac{1}{k-c} \sum_{i=c+1}^{k} Y_{j,i}$$

is the sample mean of the (non-contaminated) observations $Y_{j,c+1}, Y_{j,c+2}, \ldots, Y_{j,k}$, in replication j. After simulating the r replications, compute

$$\bar{X}(r) = \frac{1}{r} \sum_{j=1}^{r} X_j$$

and

$$S^2(r) = \frac{1}{r-1} \sum_{j=1}^{r} (X_j - \bar{X}(r))^2,$$

which are the sample mean and sample variance, respectively, of the X_j. Then, an approximate $100(1-\alpha)\%$ confidence interval for ν is given by

$$\left[\bar{X}(r) - \frac{tS(r)}{\sqrt{r}},\ \bar{X}(r) + \frac{tS(r)}{\sqrt{r}} \right].$$

For more details on initial-data deletion, including some heuristics to determine c, see Section 9.5.1 of Law (2007).

Example 1 (continued) In each of the $r = 10$ replications consisting of 1000 jobs, we applied initial data deletion by deleting all data from the first 1000 minutes of simulated time, which corresponded to about the first 100 flow times being discarded. Then for each replication we averaged the flow times of the non-deleted jobs. The resulting averages in the 10 replications were $X_1 = 40.7$, $X_2 = 92.0$, $X_3 = 53.4$, $\ldots$, $X_{10} = 62.4$, from which we computed the sample mean and sample variance. This resulted in an approximate 95% confidence interval for ν as $[64.0 \pm 12.2]$.

4.3 Other Methods for Steady-State Simulations

One problem with initial-data deletion with the method of multiple replications is that in each of the r replications, we have to discard c observations. Thus, we are deleting a total of rc observations across all of the replications. If we instead use a *single-replicate algorithm* (i.e., one with $r = 1$), then we would only delete a total of c observations.

A number of single-replicate procedures have developed, including batch means (e.g., pp. 528–529 of Law 2007), spectral methods (e.g., Anderson 1994), standardized time series methods (Schruben 1983), and regenerative methods (Crane and Iglehart 1975, Shedler 1993). (It turns out that the regenerative method does not suffer from initialization bias, so it does not require initial-data deletion. However, it instead is afflicted by another type of bias from the fact that it estimates a ratio of means by a ratio of sample averages, and a nonlinear function of sample averages is in general a biased estimator of the same function of the true means.) Some of these techniques require more sophisticated mathematics to understand and can be more difficult to implement. For an overview of these other techniques, see Bratley, Fox, and Schrage (1987) or Law (2007).

There are several methods for constructing confidence intervals for steady-state measures having pre-specified width. For example, Nakayama (1994) presents two-stage procedures for obtaining fixed-width confidence intervals using standardized time series methods, and Glynn and Whitt (1992) consider several so-called sequential procedures.

5 ESTIMATING MULTIPLE PERFORMANCE MEASURES

Consider our previous example of a manufacturing system that only accepts new jobs between 9:00am and 5:00pm, and suppose that we want to compute

- μ_1, the expected number of jobs processed in a day;
- μ_2, the probability that the number of jobs processed in a day is at least 100;
- μ_3, the number of jobs in a day that have flow time greater than 80 minutes.

These are all transient performance measures, and suppose we use the same simulation to estimate all 3 measures by running n independent replications. Let $X_{1,i}$ denote the number of jobs processed in the ith replication (day). Let $X_{2,i}$ be 1 if at least 100 customers are served in the ith replication, and 0 otherwise. Let $X_{3,i}$ be the number of jobs in the ith replication that have a flow time greater than 80 minutes.

After running n replications, suppose we construct a 95% confidence interval for each μ_s, $s = 1, 2, 3$. Let I_s denote the 95% confidence interval for μ_s, so if we ran a sufficiently large number n of replications, then $P\{\mu_s \in I_s\} \approx 0.95$ for each $s = 1, 2, 3$. But what can we say about the *joint* coverage of the 3 confidence intervals; i.e., what is $P\{\mu_s \in I_s, \text{ for all } s = 1, 2, 3\}$?

More generally, suppose that we are estimating q means μ_s, $s = 1, 2, \ldots, q$, and for each μ_s, we construct a $100(1-\alpha_s)\%$ confidence interval I_s. What can we say about $P\{\mu_s \in I_s, \text{ for all } s = 1, 2, \ldots, q\}$? In general, it is difficult to determine the joint confidence level, but Bonferroni's inequality provides a lower bound for this probability:

$$P\{\mu_s \in I_s, \text{ for all } s = 1, 2, \ldots, q\} \geq 1 - \sum_{s=1}^{q} \alpha_s.$$

Thus, in our previous example in which we had three 95% confidence intervals, the Bonferroni inequality implies the joint probability that all three confidence intervals contain their respective true means is at least 85%. Therefore, our joint confidence level for all three intervals is less than the confidence level for any single interval. If we want the joint confidence to be at least 95%, then we might set $\alpha_s = 0.01$ for each s. This would yield individual 99% confidence intervals, with the joint probability being at least 0.97. Thus, to have high confidence that all of our individual confidence intervals contain their respective means, we need to construct the individual confidence intervals with even higher confidence levels.

Often, one wants to compare different systems to see which one is the "best." For example, we may have 5 possible designs for a manufacturing system, and we want to determine which has the highest expected daily production. There is substantial literature on this topic, much of it in the areas of so-called *selection procedures* and *multiple-comparison procedures*. For an overview of these and other simulation-optimization methods, see Fu, Glover, and April (2005).

6 OTHER USEFUL METHODS

We now briefly discuss some other techniques that can be useful for simulations. *Variance-reduction techniques* (VRTs), which are also known as *efficiency-improvement techniques*, can lead to simulation estimators with smaller error (variance) by typically either collecting additional information from the simulation run(s) or changing or controlling the way in which the simulation is run. Some of the more widely used VRTs include the following:

- *Common random numbers* (e.g., see Section 11.2 of Law 2007) can improve simulations comparing two or more systems by running the simulations of the various systems using the same stream of (uniform) random numbers. In general this leads to fairer comparisons in the sense that all systems are subjected to the same random inputs. The goal is to induce positive correlation among the resulting estimators, which can be advantageous when estimating differences of performance measures between systems.
- *Antithetic variates* (e.g., see Section 11.3 of Law 2007) can improve results from simulating a single system by inducing negative correlations between pairs of replications.
- The method of *control variates* (e.g., see Section 11.4 of Law 2007) collects additional data during the simulation, where the mean of the extra collected data is known before running the simulation. For example, in a queueing simulation, one often knows the mean of a service-time distribution, and so one might additionally collect the random service times that are generated during the simulation. The data collected typically is correlated with the simulation output, and this correlation can be exploited to obtain an estimator with lower variance than the standard estimator.
- *Importance sampling* (Hammersley and Handscomb 1965, Glynn and Iglehart 1989) is often used in rare-event simulations, such as for analyzing buffer overflows in communication networks and system failures of fault-tolerant systems. In these settings, the event of interest, typically some kind of failure, occurs very rarely, and importance sampling changes the dynamics of the system to cause the event to occur more frequently. Unbiased estimators are recovered by multiplying by a correction factor known as the likelihood ratio. Heidelberger (1995) and Nicola, Shahabuddin, and Nakayama (2001) review importance-sampling methods for rare-event simulations of queueing and reliability systems.

Other VRTs include stratified sampling, conditional Monte Carlo, and splitting. These and other methods are described in Chapter 11 of Law (2007) and Chapter 2 of Bratley, Fox, and Schrage (1987).

One is often interested in estimating derivatives of performance measures with respect to system parameters. For example, in a reliability system, one may want to know how the mean time to system failure changes as a component's failure rate varies. This information can be useful in designing systems by identifying components on which to focus to improve overall performance. Also, derivative information can be used with some simulation-optimization methods (e.g., Andradóttir 1998). Techniques for estimating derivatives using simulation include perturbation analysis (Glasserman 1991, Ho and Cao 1991, Fu and Hu 1997) and the likelihood-ratio or score-function method (Reiman and Weiss 1989, Rubinstein 1989, Glynn 1990).

7 CONCLUSIONS

We have described some techniques for statistically analyzing the output from a simulation. It is important to keep in mind that the methods presented here are all *asymptotically* valid, so large run lengths are needed to ensure that valid inferences are drawn.

In addition to the references given throughout the paper, other resources covering simulation-output analysis include Banks (1998), Banks et al. (2005), Fishman (2001), Melamed and Rubinstein (1998), Ross (2002), and Henderson and Nelson (2006).

REFERENCES

Anderson, T. W. 1994. *The statistical analysis of time series*. New York: Wiley.

Andradóttir, S. 1998. Simulation optimization. In *Handbook of simulation: principles, methodology, advances, applications, and practice*, ed. J. Banks, 307–333. New York: John Wiley and Sons.

Banks, J. 1998. *Handbook of simulation: principles, methodology, advances, applications, and practice*. New York: John Wiley and Sons.

Banks, J., J. S. Carson, II, B. L. Nelson, and D. M. Nicol. 2005. *Discrete-event system simulation*. 4th ed. Upper Saddle River, New Jersey: Prentice-Hall, Inc.

Bratley, P., B. L. Fox, and L. E. Schrage. 1987. *A guide to simulation*. 2nd ed. New York: Springer-Verlag.

Crane, M., and D. L. Iglehart. 1975. Simulating stable stochastic systems, III: Regenerative processes and discrete-event simulations. *Operations Research* 23:33–45.

Fishman, G. S. 2001. *Discrete-event simulation: modeling, programming, and analysis*. New York: Springer-Verlag.

Fu, M., F. Glover, and J. April. 2005. Simulation optimization: a review, new developments, and applications. In *Proceedings of the 2005 Winter Simulation Conference*, ed. M. E. Kuhl, N. M. Steiger, F. B. Armstrong, and J. A. Joines, 83–95. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.

Fu, M., and J. Q. Hu. 1997. *Conditional Monte Carlo: gradient estimation and optimization applications*. Boston: Kluwer Academic Publishers.

Glasserman, P. 1991. *Gradient estimation via perturbation analysis*. Boston: Kluwer Academic Publishers.

Glynn, P. W. 1990. Likelihood ratio derivative estimators for stochastic systems. *Communications of the ACM* 33:75–84.

Glynn, P. W., and D. L. Iglehart. 1989. Importance sampling for stochastic simulations. *Management Science* 35:1367–1392.

Glynn, P. W., and W. Whitt. 1992. The asymptotic validity of sequential stopping rules in stochastic simulations. *Annals of Applied Probability* 2:180–198.

Hammersley, J. M., and D. C. Handscomb. 1964. *Monte Carlo methods*. London: Methuen.

Heidelberger, P. 1995. Fast simulation of rare events in queueing and reliability models. *ACM Transactions on Modeling and Computer Simulation* 5:43–85.

Henderson, S. G., and B. L. Nelson. 2006. *Handbooks in operations research and management science, volume 13: simulation*. London: Methuen.

Ho, Y. C., and X. R. Cao. 1991. *Discrete event dynamic systems and perturbation analysis*. Boston: Kluwer Academic Publishers.

Law, A. M. 2007. *Simulation modeling and analysis*. 4th ed. New York: McGraw-Hill.

Melamed, B., and R. Y. Rubinstein. 1998. *Modern simulation and modeling*. New York: John Wiley and Sons, Inc.

Nakayama, M. K. 1994. Two-stage stopping procedures based on standardized time series. *Management Science* 40:1189–1206.

Nakayama, M. K. 2006. Output analysis for simulations. In *Proceedings of the 2006 Winter Simulation Conference*, ed. L. F. Perrone, F. P. Wieland, J. Liu, B. G. Lawson, D. M. Nicol, and R. M. Fujimoto, 36–46, Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.

Nicola, V. F., P. Shahabuddin, and M. K. Nakayama. 2001. Techniques for fast simulation of models of highly dependable systems. *IEEE Transactions on Reliability* 50:246–264.

Reiman, M. I., and A. Weiss. 1989. Sensitivity analysis for simulations via likelihood ratios. *Operations Research* 37:830–844.

Ross, S. M. 2002. *Simulation*. 3rd ed. Boston: Academic Press.

Rubinstein, R. Y. 1989. Sensitivity analysis and performance extrapolation for computer simulation models. *Operations Research* 37:72–81.

Schmeiser, B. W. 1982. Batch size effects in the analysis of simulation output. *Operations Research* 30:556–568.

Schruben, L. W. 1983. Confidence interval estimation using standardized time series. *Operations Research* 31:1090–1108.

Shedler, G. S. 1993. *Regenerative stochastic simulation*. San Diego: Academic Press.

Stein, C. 1945. A two-sample test for a linear hypothesis whose power is independent of the variance. *Annals of Mathematical Statistics* 16:243–258.

Whitt, W. 1991. The efficiency of one long run versus independent replications in steady-state simulation. *Management Science* 37:645–666.

AUTHOR BIOGRAPHY

MARVIN K. NAKAYAMA is an associate professor in the Department of Computer Science at the New Jersey Institute of Technology. He received a Ph.D. in operations research from Stanford University. He won second prize in the 1992 George E. Nicholson Student Paper Competition sponsored by INFORMS and is a recipient of a CAREER Award from the National Science Foundation. He is the stochastic models area editor for *ACM Transactions on Modeling and Computer Simulation* and the simulation area editor for *INFORMS Journal on Computing*. His research interests include applied probability, statistics, simulation and modeling. His e-mail address is <marvin@njit.edu>, and his web page is <web.njit.edu/~marvin>.