Proceedings of the 1992 Winter Simulation Conference
ed. J. J. Swain, D. Goldsman, R. C. Crain, and J. R. Wilson

SIMULATION OUTPUT ANALYSIS

David Goldsman

School of Industrial and Systems Engineering
Georgia Institute of Technology
Atlanta, Georgia 30332, U.S.A.

ABSTRACT

This paper discusses the statistical analysis of output from computer simulations. In particular, we consider the problems of initialization bias (both how to identify it and what to do about it), steady-state point and confidence interval estimation, and selection among competing system designs.

1 INTRODUCTION

If the input processes driving a simulation are random variables, then the experimenter must also regard the output from the simulation as random. Thus, runs of the simulation do not directly yield the desired measures of system performance—the runs only give *estimates* of the performance measures. Since the estimators are themselves random variables, they are subject to sampling error. This sampling error must be taken into account when conducting a proper statistical analysis of the estimators; only then can the experimenter make valid inferences or decisions concerning the performance of the underlying system. Unfortunately, since simulations never produce independent and identically distributed (i.i.d.) normal output, "classical" statistical techniques do not apply to the analysis of simulation output. The purpose of this paper is to discuss practical methods to perform statistical analysis of output from discrete-event computer simulations. The paper follows the discussion in Banks, Carson, and Goldsman (1990).

In order to facilitate the discussion, we identify two types of simulations with respect to output analysis:

1. *Terminating (or transient) simulations.* Here, the nature of the problem explicitly defines the length of the simulation run. An example of a terminating simulation is that of a bank that closes at a specific time each day.

2. *Nonterminating (or steady-state) simulations.* In this case, we are interested in the long-run behavior of the system. If the simulation runs long enough, it presumedly approaches a "steady state," independent of the simulation's initial conditions. An example of a steady-state simulation is that of a continuously running production line for which the experimenter is interested in some long-run performance measure.

Techniques to analyze output from terminating simulations are based on the method of independent replications; see §2. Additional problems arise in the case of steady-state simulations. For instance, one must worry about the problem of starting the simulation—how should the simulation be initialized at time zero, and how long must the simulation be run before data representative of steady state can be collected? These initialization problems are discussed in §3. Then §4 presents techniques for point and confidence interval estimation for steady-state simulation performance parameters. §5 deals with the problem of comparing a number of competing systems. Finally, §6 presents conclusions and gives general references for the interested reader.

2 TERMINATING SIMULATIONS

We first investigate the case of terminating simulations. Suppose we simulate some system of interest over a finite time horizon and that we obtain discrete simulation output data $Y_1, Y_2, \ldots, Y_m$. The number of observations m can be a constant or a random variable. For example, the expermenter can specify *a priori* the number m of customer waiting times $Y_1, Y_2, \ldots, Y_m$ to be taken from a simple queueing simulation; or m could denote the random number of customers observed during a pre-specified time period $[0, T]$. Alternatively, we might observe continuous simulation output data $\{Y(t) | 0 \leq t \leq T\}$ over some pre-specified interval $[0, T]$; for instance, if we

are interested in estimating the time-averaged number of customers waiting in a certain queue during $[0, T]$, $Y(t)$ might represent the number of customers in the queue at time t.

For now, suppose that the goal of the experiment is to estimate the quantity

$$\theta \equiv \mathsf{E}[\overline{Y}_m],$$

where the sample mean is simply

$$\overline{Y}_m \equiv \frac{1}{m}\sum_{i=1}^{m} Y_i$$

in the discrete case, and

$$\overline{Y}_T \equiv \frac{1}{T}\int_0^T Y(t)\,dt$$

in the continuous case.

By definition, $\overline{Y}_m$ is an unbiased estimator for θ; but a sound statistical analysis requires that we also provide an estimate of precision of $\overline{Y}_m$, e.g., an estimate of $\mathsf{Var}(\overline{Y}_m)$. Since the Y_i's are not necessarily i.i.d. random variables, it is may be the case that $\mathsf{Var}(\overline{Y}_m) \neq \mathsf{Var}(Y_i)/m$ for any i. Thus, one must be very careful when estimating $\mathsf{Var}(\overline{Y}_m)$. In particular, an experimenter should *not* use the familiar estimator $S^2/m \equiv \sum_{i=1}^{m}(Y_i - \overline{Y}_m)^2/[m(m-1)]$, which is likely to be *extremely biased* as an estimator of $\mathsf{Var}(\overline{Y}_m)$.

One can instead use the method of *independent replications* to estimate $\mathsf{Var}(\overline{Y}_m)$. This simple method proceeds by conducting b *independent* simulation runs (replications) of the system of interest, each replication consisting of m observations. It is easy to force the replications to be independent—just reinitialize each replication with a different pseudo-random number seed. Let us denote the sample mean from the ith replication, $i = 1, \ldots, b$, by

$$Z_i \equiv \frac{1}{m}\sum_{j=1}^{m} Y_{i,j},$$

where $Y_{i,j}$ is the jth observation from the ith replication, for $i = 1, \ldots, b$ and $j = 1, \ldots, m$. If each run is initialized under the same operating conditions (e.g., all queues empty and idle), then the sample replication means $Z_1, \ldots, Z_b$ are *i.i.d.* random variables, and the obvious estimator for $\mathsf{Var}(\overline{Y}_m) = \mathsf{Var}(Z_i)$ is

$$\hat{V}_R \equiv \frac{1}{b-1}\sum_{i=1}^{b}(Z_i - \overline{Z}_b)^2,$$

where

$$\overline{Z}_b \equiv \sum_{i=1}^{b} Z_i/b$$

is the grand sample mean of all the replicate means. Notice the resemblance of $\hat{V}_R$ to S^2/m. The difference is that $\hat{V}_R$ is usually quite a bit less biased for $\mathsf{Var}(\overline{Y}_m)$ than is S^2/m, since the replicate means used in $\hat{V}_R$ are i.i.d.

With the estimator $\hat{V}_R$ for $\mathsf{Var}(\overline{Y}_m) = \mathsf{Var}(Z_i)$ in hand, one can state that $\hat{V}_R/b$ is a reasonable variance estimator for $\overline{Z}_b$. If the number of observations per replication m happens to be large enough, we can go further. For if m is large enough, a central limit theorem allows us to assume that the replicate means are approximately i.i.d. *normal*. Then we immediately have an approximate $100(1-\alpha)\%$ two-sided confidence interval for θ,

$$\theta \in \overline{Z}_b \pm t_{b-1,\alpha/2}\sqrt{\hat{V}_R/b}\,, \qquad (1)$$

where $t_{d,\gamma}$ represents the $1-\gamma$ quantile of the t-distribution with d degrees of freedom.

For example, suppose that our goal is to estimate the expected average waiting time for the first 5000 cutomers in an elementary queueing system. We shall make five independent simulation replications of the system, where each run is initialized empty and idle and consists of 5000 customer waiting times. Further suppose that we calculate the following replicate means:

i	1	2	3	4	5
Z_i	3.2	4.3	5.1	4.2	4.6

Then we can easily calculate $\overline{Z}_5 = 4.28$ and $\hat{V}_R = 0.487$. Setting the level $\alpha = 0.05$, we find that $t_{4,0.025} = 2.78$, and (1) yields $[3.41, 5.15]$ as a 95% two-sided confidence interval for the expected average waiting time for the first 5000 customers.

The method of replications can be used to obtain variance estimates for statistics other than sample means; and then it can be used to obtain confidence intervals for quantities other than $\mathsf{E}[\overline{Y}_m]$. For instance, one might be interested in estimating certain quantiles. For discussions on such additional uses of independent replications, see any of the standard texts cited in §6.

3 INITIALIZATION PROBLEMS

A difficult problem, particularly in steady-state output analysis, concerns the treatment of *initialization bias*. Before any simulation can be run, the experimenter must provide initial values for all of the simulation's state variables. Since the experimenter may not know what initial values are appropriate for the state variables, these values are usually chosen somewhat arbitrarily; for example, we might decide that

it is easiest and most convenient to initialize a queue as empty and idle. Such a choice of initial conditions can have a significant but unrecognized impact on the outcome of the simulation run. Thus, the initialization bias problem can lead to analysis errors.

Schruben and Goldsman (1985) and Banks, Carson, and Goldsman (1990) give several examples of problems concerning simulation initialization.

- Visual detection of initialization effects is sometimes quite difficult. This is particularly true in the case of stochastic processes having a high intrinsic variance (such as queueing systems).
- How should the simulation be initialized? Suppose that a machine shop closes at a certain time each day, even if there are jobs waiting in line to be served; these unserved jobs must be served when the shop opens the next day. Thus, a simulation model of the shop should start each day with a demand that depends on the number of jobs remaining from the previous day.
- Initialization can cause estimators of steady-state parameters to be biased and to have high mean squared error.
- Initialization can cause confidence intervals for steady-state parameters to have poor coverage.

Since initialization bias raises such serious concerns as those outlined above, one must ask how to detect and deal with this type of bias? We first survey a number of methods to detect the presence of initialization bias.

1. *Attempt to detect the bias visually* by scanning a raw realization of the simulated process. Unfortunately, such a visual analysis can easily miss the bias that happens to be present. Further, in a simulation study with a large number of runs, a methodical scan of voluminous data can become quite tedious. Some suggestions that might help to make visual analysis a bit more efficacious include transforming the data (e.g., by taking logs or square roots), smoothing the data (see Sargent 1979 or Welch 1981, 1983), averaging data across several independent simulation runs, and constructing so-called CUSUM plots (see Schruben 1982).

2. *Conduct statistical tests for initialization bias.* For instance, Kelton and Law (1983) propose an intuitively appealing sequential procedure to detect initialization bias. Schruben (1982), Schruben, Singh, and Tierney (1983), and Goldsman, Schruben, and Swain (1990) give a series of tests that check to see whether the initial portion of the simulation output seems to contain more variation than latter portions.

If initialization bias is detected, one may want to ameliorate its effects. Two simple methods for dealing with the bias are suggested in the simulation literature.

1. *Truncate the output.* It is common practice to allow simulations to "warm up" before data are retained for analysis. This is referred to as *output truncation.* The experimenter hopes that the remaining data are representative of the steady-state system under study. Since it is such a simple concept, output truncation is probably the most popular method for dealing with initialization bias. Indeed, many simulation languages have built-in truncation functions (e.g., the RESET command in GPSS). The obvious problem with truncation lies in finding a good truncation point. If the output is truncated "too early," significant initialization bias might still exist in the remaining data. If the output is truncated too late, then "good" observations might be wasted. A number of truncation rules have been proposed as described in the surveys of Gafarian, Ancker, and Morisaku (1978), and Wilson and Pritsker (1978a, 1978b). Unfortunately, these rules do not seem to perform well in practice. A good suggestion is to conduct several independent runs of the simulation, average the observations across these runs, and then visually choose a truncation point based on the averaged run. Welch (1983) describes a visual/graphical approach. A more sophisticated truncation rule is described by Chance and Schruben (1992).

2. *Make a very long run.* One might try to overwhelm the effects of initialization bias by conducting a very long run. This method of bias control has some good points. Obviously, it is simple to carry out. Further, Fishman (1978) and other authors show that under certain conditions, an untruncated stochastic process yields point estimators for the steady-state mean having lower mean squared errors than the analogous estimators from the truncated data. However, a severe problem militates against the use of one long run as a bias control technique: This method can be very wasteful with observations; for some systems a prohibitively excessive run length might be required in order to render negligible any initialization effects.

4 STEADY-STATE ANALYSIS

We shall henceforth assume that we are dealing with stationary (steady-state) simulation output, $Y_1, Y_2, \ldots, Y_n$. Suppose that our goal is to estimate some parameter of interest, perhaps the mean customer waiting time or the expected profit produced by a particular factory configuration. As in the case of terminating simulations, each run of the simulation will probably give a different point estimate for the parameter of interest. Thus, we should report the value of the point estimator along with a measure of its precision (variability). This measure of variability is of key importance, and can be in the form of a variance estimator or a confidence interval for the parameter of interest.

Over the last 30 years, a number of methodologies have been proposed in the literature for conducting steady-state output analysis. We will briefly examine the two most popular: batch means and independent replications. (As discussed earlier in §2, confidence intervals for *terminating* simulations usually evoke the method of independent replications.)

4.1 Batch Means

The method of batch means is frequently used to calculate estimators for $\mathsf{Var}(\overline{Y}_n)$ or confidence intervals for the steady-state process mean μ. The method divides one long simulation run $Y_1, Y_2, \ldots, Y_n$ into a number of contiguous *batches*. It then appeals to a central limit theorem to assume that the resulting sample means from each batch are approximately i.i.d. normal random variables. Specifically, suppose that we partition $Y_1, Y_2, \ldots, Y_n$ into b nonoverlapping, contiguous batches, each consisting of m Y_j's. (Assume for convenience that $n = bm$.) The ith batch consists of the random variables

$$Y_{(i-1)m+1}, Y_{(i-1)m+2}, \ldots, Y_{im},$$

$i = 1, 2, \ldots, b$. The ith batch mean is simply the sample mean of the m observations from batch i,

$$Z_i \equiv \frac{1}{m} \sum_{j=1}^{m} Y_{(i-1)m+j},$$

$i = 1, 2, \ldots, b$. Similar to the method of independent replications (as described in §2), we define the batch means estimator for the variance of each batch mean by

$$\hat{V}_B \equiv \frac{1}{b-1} \sum_{i=1}^{b} (Z_i - \overline{Z}_b)^2,$$

where

$$\overline{Y}_n = \overline{Z}_b \equiv \sum_{i=1}^{b} Z_i/b$$

is the grand sample mean of all the observations. If m is large enough, the batch means are approximately i.i.d. *normal*, and so (as in §2) we obtain an approximate $100(1-\alpha)\%$ two-sided confidence interval for μ.

$$\mu \in \overline{Z}_b \pm t_{b-1,\alpha/2} \sqrt{\hat{V}_B/b}.$$

This equation looks a great deal like (1). The difference is that batch means divides one long run into a number of batches, whereas independent replications uses a number of independent but shorter runs.

For illustrative purposes, we refer the reader to the example from §2 with the proviso that the Z_i's now be regarded as batch means (instead of replicate means); the same numerical manipulations now carry through the example.

Although the technique of batch means is intuitively appealing and easy to understand, problems can arise if the Y_j's are not stationary (e.g., if significant initialization bias is present), if the batch means are not normally distributed, or if the batch means are not independent. If any of these problems exist, poor confidence interval coverage may result (unbeknownst to the experimenter). To relieve the initialization bias problem, the user can simply truncate some of the output or make a long run as discussed in §3. Further, the lack of independence or normality of the batch means can be countered somewhat by increasing the batch size m. The reader is encouraged to consult the various textbooks cited herein or Schmeiser (1982) for more information concerning the method of batch means.

4.2 Independent Replications

Of the various difficulties encountered when using the method of batch means, it can be argued that the possibility of correlation among the batch means is the most deleterious. This problem is explicitly avoided when one uses the method of independent replications, described in the context of terminating simulations back in §2. The replicate means are independent by their construction; but this method is not without pitfalls. Since *each* of the b independent replications can contain initialization bias, initialization bias presents more of a problem when using independent replication than when using batch means. Further, similar to batch means, we cannot guarantee the normality of the replicate means (although this is not usually regarded as a serious problem). Of course,

these problems wash out as the replicate size m becomes large. Nevertheless, for moderate m, Law and Kelton (1984) regard the possibility of initialization bias in each of the b replications as serious enough to recommend the use of the method of batch means over that of independent replications.

4.3 Other Methods

We mention several other popular methods for obtaining variance estimators for the sample mean and confidence intervals for the steady-state process mean μ.

Spectral Estimation. The batch means method attempts to obtain a good estimate for $\mathsf{Var}(\overline{Y}_n)$. The spectral method estimates this quantity (as well as the analogous confidence intervals for μ) in a completely different manner. In particular, this approach works in the so-called *frequency domain*, whereas batch means uses the so-called *time domain*. Spectral estimation is sometimes a bit complicated to conduct, but it works well enough to suggest that the reader peruse some of relevant references, e.g., Heidelberger and Welch (1981, 1983). Meketon and Schmeiser (1984) describe the technique of *overlapping batch means*, which links together the batch means and spectral methods.

Regeneration. It is well known that the output from many simulations can be broken into i.i.d. blocks or groups, i.e., the simulation "starts from scratch" probabilistically at certain *regeneration* points. An easy example of this phenomenon occurs with the M/M/1 queue's waiting time process (which periodically returns to zero); the blocks whose endpoints are defined by waiting times equal to zero are i.i.d. The method of regeneration takes advantage of this i.i.d. structure and, under mild conditions, produces excellent estimators for $\mathsf{Var}(\overline{Y}_n)$ and confidence intervals for μ. The method is easy to implement and effectively eliminates any problems caused by initialization bias. On the other hand, it is sometimes difficult to define reasonable regeneration points, and it is sometimes the case that *extremely* long simulation runs are required in order to obtain a sufficient number of i.i.d. blocks. For more detailed discussions, the reader should see Crane and Iglehart (1975) and Crane and Lemoine (1977).

Standardized Time Series. One often uses the central limit theorem to standardize i.i.d. random variables into an (asymptotically) standard normal random variable. Schruben (1983) extends this idea by using a *process* central limit theorem to standardize a stationary simulation process into a so-called *Brownian bridge* process. (This Brownian bridge process is a Gaussian process which we can regard as a generalization of the limiting standard normal random variable from the i.i.d. case.) Properties of Brownian bridges are then used to calculate some nice estimators for $\mathsf{Var}(\overline{Y}_n)$ and confidence intervals for μ. The standardized time series method is very easy to apply and has some asymptotic advantages over batch means (as described in Goldsman and Schruben 1984); however, long simulations may be required in order for the necessary asymptotics to kick in (see Sargent, Kang, and Goldsman 1992). The interested reader should also see Glynn and Iglehart (1990) for the theoretical development of the method.

5 COMPARISON OF SYSTEMS

An important problem in simulation output analysis is that of comparing competing systems or alternative system configurations. Simulation is uniquely equipped to help the experimenter conduct such analysis. In this section, we survey three techniques for the problem of selecting the best of a number of systems.

5.1 Common Random Numbers

The idea here is to use exactly the same pseudo-random numbers in exactly the same ways for corresponding runs of each of the competing systems. By subjecting all the alternative systems to identical (or nearly identical) experimental conditions, we hope that it will be easier to distinguish which systems are best even though the respective estimators are subject to sampling error.

For instance, consider the case in which we wish to compare two queueing systems, A and B, on the basis of their expected customer transit times, θ_A and θ_B, respectively. The system with the smaller θ-value is said to be the better system. We have at our disposal estimators $\hat{\theta}_A$ and $\hat{\theta}_B$ for θ_A and θ_B, respectively. Obviously, we will declare A as the better system if $\hat{\theta}_A - \hat{\theta}_B < 0$. But if $\hat{\theta}_A$ and $\hat{\theta}_B$ are simulated independently, then

$$\mathsf{Var}(\hat{\theta}_A - \hat{\theta}_B) = \mathsf{Var}(\hat{\theta}_A) + \mathsf{Var}(\hat{\theta}_B)$$

might be quite large; in such a case, our declaration might lack conviction. If we could somehow reduce $\mathsf{Var}(\hat{\theta}_A - \hat{\theta}_B)$, then our declaration would be a much more confident one. By using common random numbers (and having a little luck), we can sometimes induce a strong positive correlation between $\hat{\theta}_A$ and $\hat{\theta}_B$;

then

$$\begin{aligned} &\mathsf{Var}(\hat{\theta}_A - \hat{\theta}_B) \\ &\quad = \mathsf{Var}(\hat{\theta}_A) + \mathsf{Var}(\hat{\theta}_B) - 2\mathsf{Cov}(\hat{\theta}_A, \hat{\theta}_B) \\ &\quad < \mathsf{Var}(\hat{\theta}_A) + \mathsf{Var}(\hat{\theta}_B), \end{aligned}$$

and one obtains a savings in variance.

5.2 Antithetic Random Numbers

Similarly, if one can induce *negative* correlation between two unbiased estimators, $\hat{\theta}_1$ and $\hat{\theta}_2$, for some parameter θ, then the unbiased estimator $(\hat{\theta}_1 + \hat{\theta}_2)/2$ might have very low variance.

The standard texts (see §6) all give advice on how to run the simulations of the competing systems so as to induce positive or negative correlation between them. If conducted properly, common random numbers and antithetic random numbers can lead to tremendous variance reductions; see the discussion in Nelson (1992).

5.3 Ranking and Selection

Ranking and selection methods form another class of techniques used to compare alternative systems. Here, an experimenter might be interested in selecting the "best" one of a number of competing processes. Further, the experimenter might wish to correctly select the best process with a certain high probability, especially if the best process is significantly better than its competitors. Ranking and selection methods are simple to use, fairly general, and intuitively appealing. There is significant literature on the subject; the reader could start by looking at the relevant discussion in Law and Kelton (1991) before graduating to the more mathematical references cited there.

6 CONCLUSIONS

Proper output analysis is one of the most important aspects of any simulation study. Since simulation output is never i.i.d. normal, the experimenter must be very careful when attempting to analyze such data; indeed, the purpose of this tutorial has been to alert the experimenter to some of the issues and techniques relevant to conducting valid analysis.

There are many interesting aspects of output analysis that we have not discussed in this paper, e.g., multivariate parameter estimation, sequential methods, and special variance reduction techniques. More advanced papers on simulation output analysis can be found elsewhere in this *Proceedings*, e.g., Hood and Welch (1992), Nelson (1992), and Seila (1992). There are also a number of excellent text books available that devote substantial discussion to the subject, e.g., Fishman (1978), Banks and Carson (1984), Bratley, Fox, and Schrage (1987), and Law and Kelton (1991).

ACKNOWLEDGMENTS

David Goldsman's work was supported by National Science Foundation Grant No. DDM-90-12020.

REFERENCES

Banks, J., and J. S. Carson, II. 1984. *Discrete-event system simulation.* Englewood Cliffs, New Jersey: Prentice-Hall.

Banks, J., J. S. Carson, II, and D. Goldsman. 1990. Computer simulation. In *Handbook of statistical methods for engineers and scientists*, ed. H. M. Wadsworth, 12.1–12.36. New York: McGraw-Hill.

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

Chance, F., and L. W. Schruben. 1992. Establishing a truncation point in simulation output. Technical Report, School of Operations Research and Industrial Engineering, Cornell University, Ithaca, New York.

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

Crane, M. A., and A. J. Lemoine. 1977. *An introduction to the regenerative method for simulation analysis.* Berlin: Springer-Verlag.

Fishman, G. S. 1978. *Principles of discrete event simulation.* New York: John Wiley & Sons.

Gafarian, A. V., C. J. Ancker, Jr., and T. Morisaku. 1978. Evaluation of commonly used rules for detecting 'steady state' in computer simulation. *Naval Research Logistics Quarterly* 25:511–529.

Glynn, P. W., and D. L. Iglehart. 1990. Simulation output analysis using standardized time series. *Mathematics of Operations Research* 15:1–16.

Goldsman, D., and L. W. Schruben. 1984. Asymptotic properties of some confidence interval estimators for simulation output. *Management Science* 30:1217–1225.

Goldsman, D., L. W. Schruben, and J. J. Swain. 1990. Tests for transient means in simulated time series. Technical Report, School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, Georgia.

Heidelberger, P., and P. D. Welch. 1981. A spectral method for confidence interval generation and run

length control in simulations. *Communications of the ACM* 24:233–245.

Heidelberger, P., and P. D. Welch. 1983. Simulation run length control in the presence of an intial transient. *Operations Research* 31:1109–1144.

Hood, S. J., and P. D. Welch. 1992. Experimental design issues in simulation with examples from semiconductor manufacturing. In *Proceedings of the 1992 Winter Simulation Conference* (this volume).

Kelton, W. D., and A. M. Law. 1983. A new approach for dealing with the startup problem in discrete event simulation. *Naval Research Logistics Quarterly* 30:641–658.

Law, A. M., and W. D. Kelton. 1984. Confidence intervals for steady-state simulations, I: a survey of fixed sample size procedures. *Operations Research* 32:1221–1239.

Law, A. M., and W. D. Kelton. 1991. *Simulation modeling and analysis*, 2d Ed. New York: McGraw-Hill.

Meketon, M. S., and B. W. Schmeiser. 1984. Overlapping batch means: Something for nothing? In *Proceedings of the 1984 Winter Simulation Conference*, 227–230.

Nelson, B. L. 1992. Designing efficient simulation experiments. In *Proceedings of the 1992 Winter Simulation Conference* (this volume).

Sargent, R. G. 1979. An introduction to statistical analysis of simulation output data. In *Proceedings of the 1979 AGARD Symp.* Paris, France.

Sargent, R. G., K. Kang, and D. Goldsman. 1992. An investigation of finite-sample behavior of confidence interval estimators. To appear in *Operations Research.*

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

Schruben, L. W. 1982. Detecting initialization bias in simulation output. *Operations Research* 30:569–590.

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

Schruben, L. W., and D. Goldsman. 1985. Initialization effects in computer simulation experiments. Technical Report #594, School of Operations Research and Industrial Engineering, Cornell University, Ithaca, New York.

Schruben, L. W., H. Singh, and L. Tierney. 1983. Optimal tests for initialization bias in simulation output. *Operations Research* 31:1167–1178.

Seila, A. F. 1992. Advanced Output Analysis Tutorial. In *Proceedings of the 1992 Winter Simulation Conference* (this volume).

Welch, P. D. 1981. On the problem of the initial transient in steady state simulations. IBM Watson Research Center, Yorktown Heights, New York.

Welch, P. D. 1983. The statistical analysis of simulation results. In *The computer performance modeling handbook*, ed. S. Lavenberg. Orlando, Florida: Academic Press.

Wilson, J. R., and A. A. B. Pritsker. 1978a. A survey of research on the simulation startup problem. *Simulation* 31:55–58.

Wilson, J. R., and A. A. B. Pritsker. 1978b. Evaluation of startup policies in simulation experiments. *Simulation* 31:79–89.

AUTHOR BIOGRAPHY

DAVID GOLDSMAN is an Associate Professor in the School of Industrial and Systems Engineering at the Georgia Institute of Technology. His research interests include simulation output analysis and ranking and selection. He is Vice President of the TIMS College on Simulation.