Proceedings of the 2002 Winter Simulation Conference
E. Yücesan, C.-H. Chen, J. L. Snowdon, and J. M. Charnes, eds.

PANEL ON CURRENT ISSUES IN SIMULATION INPUT MODELING

Russell R. Barton

The Smeal College of Business Administration
The Pennsylvania State University
University Park, PA 16802, U.S.A.

Russell C. H. Cheng

Department of Mathematics
University of Southampton
Southampton, Hampshire SO17 1BJ, U.K.

Stephen E. Chick

Technology Management Area
INSEAD
77305 Fontainebleau Cedex, FRANCE

Shane G. Henderson

School of Operations Research and Industrial Engineering
Cornell University
Ithaca, NY 14853, U.S.A.

Averill M. Law

Averill Law & Associates
2601 N. Campbell Ave., Suite 110
Tucson, AZ 85719, U.S.A.

Lawrence M. Leemis

Mathematics Department
The College of William & Mary
Williamsburg, VA 23187-8795, U.S.A.

Bruce W. Schmeiser

School of Industrial Engineering
Purdue University
West Lafayette, IN 47907-1287, U.S.A.

Lee W. Schruben

Dept. of Industrial Engineering and Operations Research
University of California, Berkeley
Berkeley, CA 94720-1777, U.S.A.

James R. Wilson (Moderator)

Department of Industrial Engineering
North Carolina State University
Raleigh, NC 27695-7906, U.S.A.

ABSTRACT

In recent years, substantial progress has been made in the development of powerful new approaches to modeling and generation of the stochastic input processes driving simulation models. In this panel discussion, we examine some of the central issues and unresolved problems associated with each of these approaches to simulation input modeling.

1 RUSSELL R. BARTON

I would like to encourage additional research in the use of input resampling methods for the analysis of simulation output when the input distributions are based on empirical data. I'll focus on simulation driven by empirical distributions, rather than fitted parametric distributions, but I believe that this resampling approach is also important when fitting parametric distributions to empirical data. My comments are intended to initiate discussion, and are organized around three topics:

- Why two-step bootstrap resampling should be conducted when computing confidence intervals for parameters characterizing simulation output, and proper conduct of the two-step method;
- What is wrong with the two-step bootstrap method; and
- Why focus on empirical input distributions rather than fitted parametric distributions.

1.1 Why Two-Step Bootstrap Resampling?

There is some confusion about the use of bootstrap resampling of input distributions, in terms of how the simulation runs are conducted, and how the results are analyzed. I'll summarize the approach that Lee Schruben and I have been using, and I refer the reader to Barton and Schruben (2001) for details.

First, the bootstrap resampling should not be done *within* a simulation run, but rather between runs. Resampling within a single run has no impact on the simulation: the resulting cdf for the input values matches the original empirical cdf. This may explain Russell Cheng's comment (Cheng 1994):

> Given that sampling of the smoothed empirical cdf takes place in the second step, …, it is not clear that the first step is necessary or even helpful to carry out.

Resampling *between* runs does make a difference, however. Each simulation run is conducted with different input distributions. This component of variation is added to the normal run-to-run variation that occurs due to finite run length. This combination of two sources of variability affects the validity of the bootstrap approach, as I discuss in the next section. In our experiments, the variation due to changes in the input distributions overwhelmed run-to-run variation due to finite run lengths. The implication is that, without bootstrap resampling, the simulationist sees artificially small run-to-run variation, and constructs overly optimistic confidence intervals for output parameters. This is why two-step bootstrap resampling should be used: to capture the uncertainty in the predicted system performance due to finiteness of the empirical data used to determine the input distributions.

There is a second issue in the use of the two-step resampling method: intervals are calculated from empirical percentiles of the output statistic, not t-based intervals using the across-run standard deviation. The reason is that the bootstrap resampling provides an estimate of the distribution function of the statistic one would expect from repeated runs with different input samples of the same size. Increasing the number of replications (that is, bootstrap resamples) provides a better estimate of the *same* distribution, rather than a tighter distribution for the statistic.

1.2 What Is Wrong with the Two-Step Bootstrap

Cheng (1994) describes the two sources of error that arise when using input distributions that are fitted to empirical data. He identifies the first as bias error, due to the finiteness of the empirical sample, and the second as variance error, due to the finiteness of the simulation run length. He proposes a parametric bootstrap approach to estimate both components of variance across simulation runs.

These two sources of variation complicate the two-step bootstrap approaches that are described in Barton and Schruben (1993) and Barton and Schruben (2001). A necessary condition for bootstrap estimates of the distribution of the output statistic to converge to the true distribution (as sample size and the number of bootstrap replications go to infinity) is that the output statistic is a smooth function of the sample data. In some cases less restrictive assumptions are possible but the statistic still must still be a deterministic function of the sample input data. For the two-step bootstrap strategy, the finite length of each simulation run means that the statistic is a stochastic function of the input sample. Of course, the condition of a deterministic function need only hold approximately, since calculations of any bootstrap statistic using digital computers leaves a result that might be thought to have a random perturbation at the level of the machine precision.

For the queuing examples presented in our two previous papers, simulation run lengths had 4000 customers or more with input sample sizes of 500 or less. For these cases, the run-to-run variation from finite input sample size far exceeded the variation from finite run length. The resulting coverages for the bootstrap intervals were indistinguishable from the nominal 90%. Figure 1 shows that it is possible to generate poor coverage with the two-step bootstrap approach. The graph shows the nominal coverage of 90% intervals based on bootstrap and uniform bootstrap methods. These methods were applied to input distribution sample sizes of 500 for an $M/M/1/10$ queue with $\rho = 0.7$, with simulation run lengths varying from 50 to 8000 customers. The dashed lines indicate approximate 95% confidence intervals on the coverage. Coverage for both methods is indistinguishable from the nominal 90% for run lengths with 2000 or more customers (after warm-up).

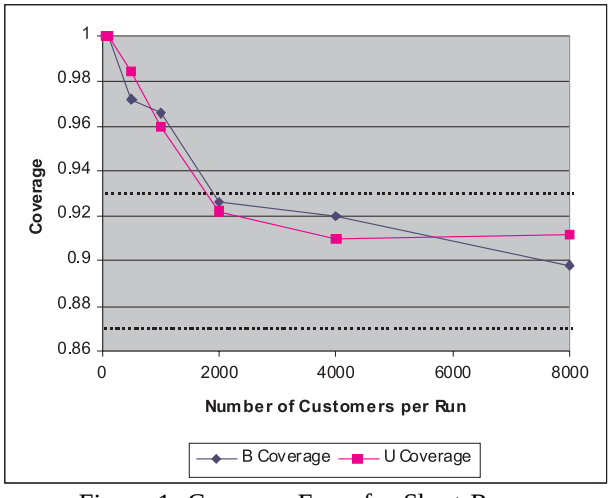

Figure 1: Coverage Error for Short Runs

To use two-step bootstrap resampling, one must ensure that the variation due to finite run length is relatively small compared with the variation due to the finite input sample size. Otherwise, the intervals may be overly conservative. This is easy to check by making replications using the same input samples, and comparing this variation with repeated runs using bootstrap resamples of the input distributions.

1.3 Why Focus on Empirical Distributions

Perhaps the most common approach to input modeling in simulation is to fit parametric models based on sample data. There have been many papers on this topic in previous Winter Simulation Conferences. See for example last year's tutorial by Leemis (2001) and the references therein.

Nonetheless, while this is the most common approach, I believe that it is high risk, for several reasons.

1. There is rarely a theoretical justification for a particular distribution. Many simulation responses are sensitive to the tails of the input distributions, yet these tails are precisely where parametric models often fail to capture reality.
2. Testing for or selecting a correct model in such circumstances might appear to be futile: as George Box said, "All models are wrong, some are useful."
3. A parametric distribution gives the simulationist an artificial sense of the well-definedness of the simulation. While the model is well-defined, it need not be well-connected to reality.

Empirical distributions are easy to use with most simulation packages. Using them makes the limited fidelity of the model more transparent, particularly if two-step bootstrap resampling is used to characterize uncertainty in the output statistics.

1.4 Acknowledgments

Many of the ideas in this statement came from discussions with Lee Schruben. Important questions and clarifications have been brought to light by discussions with many regular WSC participants, including Russell Cheng, Steve Chick, Dave Goldsman, Shane Henderson, David Kelton, Barry Nelson, Bruce Schmeiser, Bob Sargent, Jim Wilson, and others.

2 RUSSELL C. H. CHENG

Law and Kelton (2000) give a good introduction to basic input modelling. The stance adopted here is that input modelling is inextricably linked with input data. There are three main steps in handling input data: (1) Data Gathering, (2) Data Inspection, and (3) Data Analysis. Input modelling can be equated with the last of these, if we take the view that proper understanding of data requires its characterisation by a statistical or stochastic model.

Much of the panel discussion will no doubt focus on methodological issues involved in (3), as these tend to have the greatest academic appeal. However, it is worth emphasising that no amount of academic cleverness can make up for basic inadequacies in the quality of the practical, real-life data that fuels and informs the input modelling process. So, before becoming engrossed in technical niceties it is as well to acknowledge that, arguably, far and away the most important part of input modelling is (1) and that this is largely out of our—the analysts'—hands. It is dependent on the front-line people who gather and collate the data. Even if the importance of (1) is agreed, it might be deemed to fall—thankfully—outside the remit of this panel session! However (2), the Data Inspection phase does come within our remit, and requires some comment, as this phase should culminate in the formulation of possible input models that are the object of fuller study in (3).

My experience is that a typical case study involves much effort in manipulating and organising of data, frequently from incoherent and suspect sources, in order to shape it into some semblance of credibility and usability. This can consume a huge part of a project's time. In Data Inspection, two aspects need consideration and evaluation, each with its own subheadings:

1. Data Quality:
   (a) Reliability
   (b) Stability
2. Data Complexity:
   (a) Type (discrete / continuous, quantitative / qualitative)
   (b) Size/Quantity of Data
   (c) Probabilistic Nature (including dependence)

'Rubbish in, rubbish out' is a well-known catch-phrase that captures the key difficulty of Data Reliability. It often requires input of expert knowledge to resolve. Unexplained variation is usually ascribed to randomness simply because we do not know its cause. However unexpected blips in data can often be explained away with additional information. For instance, an at first sight mysterious dip in reported TB incidence in an African country was due simply probably to poor data collection due to the civil war that raged over the same period as the dip.

By Data Stability I mean the validity of the data outside the period of its collection. Stock market forecasts based on data gathered up to, but not including, the recent bursting of the dot.com bubble, look misguided. In many simulation studies time is better spent evaluating data quality rather than on the simulation modelling itself.

In preparing this position statement, I was led to surveying the summer projects carried out by the students of our Master programme in OR, for a wide range of corporate clients. All involve handling of real data. Many of the projects involved simulation; and simulation might well have been used in those cases where it was not. The variety of data was sobering. Here is a small selection from the last year or two: HIV epidemics, car windscreen replacement, soft fruit distribution, package holiday sales, air-traffic controller performance, hospital bed-occupancy, mail orders, call centres, human growth curves, environmental risk statistics for earthworms, racing car performance, lost luggage returns, horse-race betting, LNG production, car worker motivation, bank account statistics, war-game studies. What stands out from this list is the sheer variety and complexity of the data. However it is possible to impose some order. A simple classification is to observe that data has a form that is typical of the sector from which it arises. In OR type studies important sectors are: (i) Manufacturing and Production, (ii) Services, (iii) Leisure, (iv) Environmental and Life Sciences, and (v) Physical Sciences. An amusing exercise is to try to pigeon-hole the MSc project list according to sector. The focus of discrete event simulation has perhaps been in (i). However the character and nature of data in other sectors is sometimes refreshingly different. There is some scope for formalising this problem of unravelling Data Complexity as one of Data Classification.

I highlight four further specific problems of Data Complexity, in no particular order.

The first is to note that recent advances adopt a much more sophisticated approach to input modelling. Advanced techniques, that allow for data dependence, are reviewed by Nelson and Yamnitsky (1998) and by Schmeiser (1999). More specific techniques for generating correlated data are described by Deler and Nelson (2001) and Ghosh and Henderson (2001).

The second involves non-quantitative data. Several of the MSc projects, including air-traffic controller performance, war games and car worker motivation, involved data concerned with human factors. These were hard to quantify and make reliable. In the war game example, it was readily acknowledged that morale and esprit-de-corps were key factors but almost impossible to quantify. In another instance, a colleague asked participants at a recent workshop on Forecasting if they made use of subjective forecasting techniques. All but one of the group of about 40 indicated that they did. Alas for Box-Jenkins! Bayesian inference is a powerful formal method for injecting human opinion into an analysis. This area has been the subject of much study by statisticians of the Bayesian persuasion, but has not yet received the wider recognition that it now deserves.

The third problem is a specific instance of a Data Distribution problem. It might be termed the Mixture Problem and it occurs, for example, in queueing situations. In the MSc list it occurs in the hospital-bed, call centre, mail order, and air-traffic controller examples. Often data is not homogeneous, but is a combination of samples from several different populations. Correct input modelling will require identification of the separate subpopulations. This is a difficult problem technically as it is non-standard. There has been much recent interest from both classical and Bayesian statisticians concerning this problem (See for example Cheng, 1998).

The fourth problem concerns Data Size. One seems to lurch from problems where there are literally only a couple of dozen data points, to those where gigabytes of data are all too readily available. Bayesian techniques seem good for the former situation and Resampling techniques for the latter. I will return to this problem after discussion of Data Analysis techniques. A point to note is that often, when large data sets are involved, the problem is to try to identify small subsets of key factors. Classification and Regression Tree (CART) and Neural Network techniques are two methods that are of interest in this regard but where the methodology is not all that advanced.

We turn now to Data Analysis. The purpose of this is twofold. The first aim is to better understand the structure of the data. This should be made explicit by fitting a formal statistical or stochastic model of the data. Secondly, once such an input model is fitted, it can then be used for generating input data in its own right, if this is required in the simulation. The quality of fit of the input model to the data needs to be evaluated. However this is not enough. The real need is to assess how the quality of fit of the input model affects the quality of the simulation itself. This can conveniently be treated as a validation problem. (Kleijnen et al. 2001)

Two interesting methodologies seem especially appropriate to Data Analysis: Bayesian Inference and Bootstrap Resampling.

Bayesian Inference is an excellent approach for bringing together prior information and new data (see Chick 2000), which also resolves questions of sensitivity analysis in a neat way (see Zouaoui and Wilson 2001, for example). There are two scenarios that can be highlighted. The first is the situation where input streams are dependent on parameters, denoted by the vector θ, whose values are uncertain. In the Bayesian approach, this uncertainty is captured by stipulating a prior distribution, $\pi(\theta)$, for θ. Let us denote by $r(\theta)$, the result obtained from a simulation run made at parameter value θ. Parameter uncertainty can then be accounted for by making runs at different θ values. Denote the set of θ values used as $\{\theta_i, i = 1, 2, \ldots, n\}$. If the θ_i are a random sample drawn from $\pi(\theta)$, then $\{r(\theta), i = 1, 2, \ldots, n\}$ is a random sample whose EDF estimates the (prior) CDF of r. This EDF can then be used for inference about r in the usual way.

If there is real data x, whose distribution $p(x|\theta)$ depends on θ, then this can be incorporated to construct a posterior distribution $p(\theta|x)$ using Bayes' theorem:

$$p(\theta|x) = l(x|\theta)\pi(\theta) \Big/ \int l(x|\theta)\pi(\theta)\, d\theta,$$

where $l(x|\theta)$ is the likelihood (i.e. $p(x|\theta)$ treated as a function of θ). Markov Chain Monte Carlo (MCMC) is currently the popular method for carrying out this construction. However I would advocate Importance Sampling as being intuitively easier to understand than MCMC and as being just as easy, if not easier, to apply in many cases.

Note that the data, x, can appear at two points. The simpler situation is where x is just (additional) data generated by the input process alone. For instance, in a queue problem, x might be more customer arrival rate data. In this case the Bayesian procedure is applied solely at the input modelling stage, with the posterior $p(\theta|x)$ obtained *before the simulation runs are done.*

The second more interesting case is where x represents output that the *simulation itself is attempting to reproduce.* In this case the likelihood will characterise the distributional relationship between x and r and so involve both; so it is not just $l(x|\theta)$, but $l(x, r|\theta)$. The simulation runs thus form part of the sampling process used to determine the posterior distribution. If runs are expensive this obviously can be a serious problem. This second situation occurs in many interesting forecasting or predictive situations and is an area of current research interest. A very interesting variant that there is no space to discuss concerns certain simulation optimization problems where the Bayesian approach can handle situations that are impossible to contemplate using a classical formulation.

Bootstrap Resampling is a general approach (Cheng 2001; Barton and Schruben 2001) that really should be better recognised in simulation given its ease of implementation and generality of application. One reason for this lack of recognition is an often voiced suspicion that one appears to be getting something for nothing. This stems from a misunderstanding. Bootstrap resampling *does not improve an estimate*—nothing can do that if an efficient estimator is being used—but is simply a means of estimating the *variability*, more generally the *distribution* of the estimator. Thus it is clearly a useful method for handling questions involving goodness-of-fit, validation, and sensitivity analysis.

In considering bootstrap methods an interesting question is whether one should bootstrap from the original data or from the fitted model. When large real data sets are available then these might reasonably be taken to be a good proxy of the population, and bootstrapping from the data is the natural thing to do. However when the original data set is small in size then parametric bootstrapping seems far more preferable.

In summary, the impact of input modelling on the quality of an overall simulation is often of crucial importance. The above comments are merely examples of areas and issues that I have been drawn to through my own research.

3 STEPHEN E. CHICK

Many simulations are run to study how simulation output depends on the inputs, be they design parameters or parameters of probability distributions that describe randomness (stochastic uncertainty) in an abstracted system. Many tools exist to analyze simulation output, either for a single set of inputs, or as a function of the inputs (e.g., Law and Kelton 2000; previous WSC proceedings; references therein).

Simulations are often required to do more than relate inputs to output—a model may be required to represent an existing or planned system. While design parameters are often easily related to simulation models (e.g., 5 real servers correspond to 5 simulated servers), input distributions and parameters pose a challenge. Why should we believe that actual service times have a Weibull distribution with shape parameter $\alpha = 5$ and scale parameter $\beta = 25$? Even if a simulation exactly determines the mean performance of the system as a function of the input parameters α, β, we still may be uncertain about the performance of the system. Why? In practice we usually don't know value of input parameters exactly, or even if the Weibull is the right distribution (structural uncertainty).

From this perspective, *input distribution selection* and *model validation* go hand in hand. Does a decision-maker find that a given distribution and parameter reasonably represent the randomness in a system? Including the decision-maker in the simulation process is a widely accepted key factor of successful simulation projects. And including the decision-maker makes input modeling a *subjective* process.

The classical approach commonly evaluates statistical methods, like the standard input selection method, by studying performance during repeated sampling as a function of known parameters. It is comforting to know how well a method works, on average, when applied repeatedly. Asymptotic methods (e.g. asymptotic normality results for MLEs) further inspire confidence in the standard mechanism. There are many benefits with the standard parameter estimation/goodness-of-fit approach.

But there are practical problems (e.g., Raftery 1995). And a decision maker often does not have the benefit of repeated sampling. There may be a few big simulation projects per year, not an infinite sequence of similar projects. A huge amount of data may not be available.

Bruno De Finetti (1990) asserts:

Probability doesn't exist.

In part he means that probability is a *subjective* statement about uncertainty, not a property defined only for conceptually infinite sequences of samples. An implication for simulation is to posit a joint probability model that describes both stochastic uncertainty and structural uncertainty. This includes formalizing statements like 'the arrival rate λ is unknown, but is likely around 10–12 per hour' with a probability distribution for the input parameter λ.

This is consistent with the subjective nature of input selection and simulation validation. Conditional probability and observations (field data and simulation observations) help infer the unknown input distributions and parameters of the modeled system. The posterior distribution of the inputs is then determined by Bayes' rule, given available data. The simulation output mean, the expectation taken over both stochastic and structural uncertainty, is determined by a *Bayes' model average* (e.g. Draper 1995; or Chick 2000 in a simulation context). The BMA samples input models and parameters from an appropriate posterior distribution. Simulation outputs are generated for each sampled input model/parameter combination.

The Bayesian approach is not new: Laplace and Bayes initiated idea streams centuries ago; De Finetti indicates that Hume and Berkeley influenced his thinking. But the ideas are not outdated and purely philosophical. They shed practical insights into many applications (e.g. Gilks et al. 1996). Young and Lenk (1998) even improve stock market portfolio allocation by accounting for input uncertainty.

The BMA is not the only way to handle input uncertainty, as the other panelists indicate. And rigid adherence to a BMA framework may be more trouble than it is worth (say, when a rough-cut point estimate/sensitivity analysis provides sufficient information to a decision maker). However the BMA approach does provide a coherent framework for thinking about uncertainty that makes direct links to other operations research tools, such as decision analysis. And it provides a way of extending the conversation about uncertainty analysis when a rough-cut analysis is inconclusive.

Many practical issues (integration to determine posterior distributions; generation of samples from those distributions) have been well studied. Nonuniform random variable generation and output analysis results from the discrete-event simulation literature play an important role. Markov Chain Monte Carlo (MCMC) methods play an increasingly important role. It is not possible to cite all the important work given space limitations, but the panel presentation will attempt to identify a number of important contributions.

But there are several issues to resolve. In addition to the usual simulation desire to more efficiently integrate and sample variates from posterior distributions, these include:

Prior distributions and sensitivity analysis. How can a modeler assess a 'useful' prior distribution for unknown input parameters, given that standard 'noninformative' distributions can have odd consequences in the absence of large amounts of data? Initial work with moment methods has shown some feasibility, but more work is required to handle a broader class of input distributions.

Graphical interfaces to assist in visualization of how changing the prior distributions of input parameters affects the distribution of outputs would be useful.

Experimental design methods for input selection may also help identify important parameters, and provide better information about performance than naive BMA input sampling.

Simulation software. Input selection, response modeling, and output analysis are linked by the BMA. Inputs are sampled from a distribution, simulations are run with those parameters, and the input/output combinations help describe the system response.

Most simulation software packages do not provide a simple interface to implement the BMA and to *visualize* how output uncertainty is influenced by both input uncertainty and stochastic variation. Can a closer link be created between input parameter selection software, the simulation engine, and output analysis tools, to help a modeler assess the relative importance of stochastic and structural uncertainty?

Inputs may be correlated. How does recent work on generating vectors of correlated variates apply in the BMA context?

Uncertainty reduction. How should resources be balanced, given a choice between running more replications, to reduce stochastic uncertainty, or collecting more field data, to reduce input parameter uncertainty, when the goal is to reduce an overall measure of uncertainty? Asymptotically optimal results exist for estimators of the output mean, and for estimating the distribution function of a conditional mean, under certain conditions. Can those conditions be relaxed? Can finite sample results be obtained?

Inverse problem. A decision-maker may be better able to specify distributions about system outputs than about inputs. For example, it may be easier for a manager to say that the mean weekly production is around 40–45 jobs, rather than to specify a joint distribution of the unknown parameters of service time distributions.

The inverse problem is to identify which set of probability measures on inputs that are compatible with a specified distribution on the outputs. In simple cases, e.g. an $M/M/1$ queue, a direct mapping from some inputs to outputs is known. But in general, how should information about likely outputs be used to help identify reasonable

input parameter values? Perhaps entropy methods are appropriate.

4 SHANE G. HENDERSON

The input modeler today is often spoiled. Large databases are now the norm rather than the exception. The availability of massive amounts of data suggests that the use of trace-driven simulations will become more prevalent in the near future. In trace-driven simulations, the recorded data, rather than observations generated from fitted distributions, is used to drive the simulation.

There are some important unresolved issues related to the use of trace-driven simulations. For example, the data may not have been correctly recorded. This issue arose in ambulance simulations in Auckland, New Zealand (Henderson and Mason 1999), where ambulance drivers were required to punch a button in the ambulance as they completed the various steps related to a call. If the ambulance drivers forgot to push the button at the appropriate time, then they often simply "caught up" by repeatedly pushing the button at a later time. This behaviour shows up as unrealistically short scene times, hospital transfers and so forth in a database. Theoretically sound methods for dealing with such problems are needed. It is also highly desirable to develop output analysis procedures for trace-driven simulations that can make some statistically sensible statement about the output. Some work on validation of trace-driven simulations (Kleijnen et al. 2001) is relevant, but not directly so. More work is needed.

Another area of input modelling that warrants further attention is the modelling of dependence in input random variables. One of the key difficulties in this setting is that the information required to specify the joint distribution of a set of random variables grows rapidly with the number of dependent random variables. Even in the finite dimensional case, where one is attempting to model the joint distribution of a finite number of random variables, we often resort to simply matching marginal distributions and correlations; see Cario and Nelson (1997), Ghosh and Henderson (2002), and Kurowicka and Cooke (2002) for example. It is highly desirable to be able to exercise more control over the joint distribution than existing methods allow without an explosion in data requirements. If one has a reasonable amount of data, then one method that is often neglected in this setting is that of kernel density estimation, which can provide an estimate of the joint distribution.

The infinite-dimensional problem is perhaps even more challenging. Here the goal is to generate a time series of observations (possibly vector-valued). Many of the existing methods for doing this (e.g., Cario and Nelson 1996; Melamed, Hill, and Goldsman 1992; Deler and Nelson 2001) are applicable in settings where the time series is short range dependent. Long range dependence seems to be considerably more difficult to capture with simple models, but strongly desired in, for example, telecommunications applications.

One final area that deserves a great deal more attention is the effect of parameter, or even model, uncertainty in simulation results. I am (almost) certain that this issue will be addressed by other panelists, and so will confine my remarks to my own planned research in this area. Suppose that one has a distribution, G say, on the input parameters θ for a simulation experiment. This distribution may arise through Bayesian analysis, asymptotic theory (asymptotic in the number of observations used to fit a distribution) or otherwise. The random variable of interest (e.g., throughput in a manufacturing setting), X say, depends on θ. If one assigns a fixed value θ_0 to θ, then in great generality we can view the output of the simulation experiment as an estimate of $E(X|\theta = \theta_0)$. Perhaps the primary object of interest is the *distribution* of $E(X|\theta)$. Lee and Glynn (1999) estimated the distribution function of this quantity, but perhaps the density of this quantity (assuming it exists) would be more helpful in building understanding. This is a subject of current research.

5 AVERILL M. LAW: THE DEVELOPMENT OF A COMMERCIAL DISTRIBUTION-FITTING SOFTWARE PACKAGE

In this talk we discuss the development of a commercial distribution-fitting software package for simulation practitioners and also for analysts in other application areas (actuarial science, agriculture, economics, reliability engineering, risk analysis, etc.) The first version of our software was developed by an undergraduate student and myself at the University of Wisconsin in 1978 and was called Explore. We had read a large number of papers on distribution fitting and our philosophy in developing Explore was to include virtually every relevant statistical feature. (At the time we did not realize that such an extensive number of features was beyond the statistical background and interest of the typical simulation practitioner.)

Based on the success of Explore in teaching a Master's degree course in simulation, we decided to commercialize the software in 1981. The software hit the market in 1983 and was called UniFit (R) (*uni*variate *fit*ting software), with the first customer being the U.S. Air Force. (An unfortunate aspect of the name UniFit was that some people missed the first "i" in UniFit and asked for information on the "unfit" statistical package.) UniFit met with only limited commercial success due to its technical nature and due to the fact that it was oriented toward mainframe computers—personal computers and a standard graphical interface were not widely available at that time.

In 1985 UniFit was converted to run on PCs under DOS and improved graphics were introduced. However,

UniFit's popularity increased by only a moderate amount. Apparently, many people still did not realize the critical role that probability and statistics plays in a successful simulation study, and/or they did not have the statistical background required to use UniFit.

UniFit did not change significantly between 1985 and 1992. We then came to the amazing realization that when it comes to distribution-fitting software, "less is actually more." We decided to develop an automated procedure for fitting distributions to a data set that was easy and fast to use, but did not sacrifice technical correctness. The methodology that we developed automated the following steps:

- Selecting a set of candidate theoretical probability distributions (gamma, beta, normal, etc.) that is consistent with the range of the data set being analyzed (i.e., nonnegative, bounded, or unbounded);
- Estimating the parameters of each candidate distribution using a statistically sound method such as maximum likelihood;
- Ranking the fitted distributions using one or more heuristics (e.g., the Kolmogorov-Smirnov test statistic) to determine which distribution provides the best representation for the data set; and
- Evaluating the best-fitting distribution to see if it is good enough in an *absolute* sense to actually use in a simulation model. (For perhaps one third of all data sets, no theoretical distribution provides a good representation. This is often because the data set is a mixture of two or more heterogeneous populations or because the data have been significantly rounded.)

The ranking and evaluation algorithm was developed as follows. We had 15 heuristics that were thought to have some ability to discriminate between a good-fitting and bad-fitting distribution. To determine which of these heuristics were actually the best, a random sample of size n was generated from a known "parent" distribution, and each of the 15 heuristics was applied to see if it could, in fact, choose the correct distribution. This was repeated for 200 independent samples, giving an estimated probability that each heuristic would pick the parent distribution for the specified sample size. The whole process was repeated for 175 parent-distribution/sample-size pairs, resulting in several heuristics that appeared to be superior. These heuristics were combined to give the overall algorithm for ranking the fitted distributions. The 35,000 generated data sets were also used to develop the rule for providing an absolute evaluation for a fitted distribution.

At the same time that automated fitting was introduced into UniFit, we also added the ability to provide an explicit representation of a selected distribution for a large number of commercial simulation-software products. The addition of automated fitting and of simulation-software representations made the distribution-fitting process much easier for an analyst and resulted in a considerable increase in popularity for UniFit.

In 1995 our distribution-fitting package was modified so that it would run under Windows and the name of the software was changed to ExpertFit (R).

Version 2 of ExpertFit was released in 1999 and featured a "distribution viewer" and a batch-mode capability. The distribution viewer allows an analyst to see characteristics of a distribution without entering any data. By using a slider bar for each parameter, one can interactively and quickly change the distribution being viewed. Batch mode allows an analyst to enter and analyze a large number of data sets in a matter of seconds—it was developed under a contract with Accenture (then Andersen Consulting).

In 2000 we, once again, tried to make our software easier to use. However, we did not want to "dumb down" the software at the expense of sacrificing technical correctness, as we had seen some software vendors do. We therefore introduced into Version 3 of ExpertFit two modes of operation: Standard and Advanced. *Standard Mode* is sufficient for 95 percent of all data analyses and is much easier to use than Version 2 of ExpertFit. It focuses the user on those features that are really important at a particular point in an analysis. *Advanced Mode* contains numerous additional features for the sophisticated user and is similar in comprehensiveness to Version 2, but it is easier to use. A user can switch from one mode to another at any time during an analysis.

We set out to improve the methodology that ExpertFit uses to fit a distribution to a data set in early 2002. For most of the probability distributions available in ExpertFit, we were able to develop new and improved methods for estimating the parameters of a distribution, and this capability was released as Version 4 of ExpertFit. We tested this new distribution-fitting methodology on 69 sets of real-world data and found that it produced better-fitting distributions for 84 percent of the data sets tested, as compared to the methodology used in Version 3. Note that the Anderson-Darling statistic—a powerful measure of goodness-of-fit—was used to compare the quality of fit for the distributions produced by Versions 3 and 4 of ExpertFit. We also introduced in Version 4 new capabilities for batch mode—this work was funded by a contract from Oak Ridge National Lab.

We have always had two major goals in the development of ExpertFit. The first is to provide the most comprehensive and technically correct set of features available in a simulation input-modeling package. In this regard, ExpertFit makes available 40 different probability distributions, 30 high-quality plots, 4 technically correct goodness-of-fit tests, and support for 26 different simulation-software packages. There are also modules available to help an analyst choose reasonable distributions in the absence of data, including

modeling machines that are subject to random breakdowns. Many of the features that we have added to ExpertFit were as a result of requests made by our clients.

We have also strived to make ExpertFit easy to use. The availability of Standard and Advanced Modes of operation allows an analyst to configure ExpertFit to their particular background and to their application. ExpertFit has a comprehensive amount of documentation that includes the following:

- 450 pages of context-sensitive help for all menus and results tables/graphs;
- Online feature index and tutorials on goodness-of-fit tests, available distributions, etc.; and
- User's Guide with 8 complete examples.

In academia, of which I was member for 17 years, it is critical for algorithms and test procedures to be carefully documented in the literature, so that they can be independently checked for technical correctness. However, in the world of commercial software the practice is often quite different, since it is common for one software company to copy the original ideas of another company. For this reason, many of the algorithms used by ExpertFit for fitting, ranking, and evaluating distributions, are proprietary and do not appear in the literature. However, we spend a considerable amount of time and effort verifying the efficacy and technical correctness of each and every methodology used in ExpertFit.

6 LAWRENCE M. LEEMIS

My remarks concerning input modeling will address specific areas within input modeling, as opposed to generalities. I will discuss (1) the transfer of new models and associated algorithms for input modeling to commercial input modeling software, (2) opportunities for the input modeling community to partner with hardware vendors in the analysis of data collected automatically, and (3) future research topics in input modeling.

6.1 Technology Transfer

There are generally three types of discrete-event simulation input modeling software available presently: standalone input modeling software available from vendors, input modeling software associated with a particular simulation modeling language, and freeware on websites posted by academics and practitioners. The first two types of software typically focus on the estimation of the parameters in standard univariate distributions (e.g., Weibull) fit by maximum likelihood or matching moments. The freeware generally fills in the gaps that are left by the commercial software.

There are relatively elementary stochastic models (e.g., parametric estimation for nonhomogeneous Poisson processes) that are not presently considered by the existing commercial software, yet are important enough that they should be easily accessible to modelers. How can these models be incorporated into input modeling software? I will suggest three ways:

1. **Competition**. Once one of these models is implemented in one of the commercial packages and gains popularity, other packages will follow suit. Healthy competition of this type, for example, has resulted in increasing capability for the Maple and Mathematica computer algebra systems.
2. **Winter Simulation Conference round table session**. The developers of the algorithms for more sophisticated input modeling techniques should meet with the commercial software developers at the Winter Simulation Conference to encourage the implementation of the new modeling techniques into existing packages.
3. **Demand**. Once the simulation modeling languages incorporate a more sophisticated input model into their packages, it will only be a matter of time before the software vendors will include fitting software for this particular input model in their commercial software. Unfortunately, this pull-type approach has put us in our present state where most of the commercial input modeling software is limited to univariate data fitting.

6.2 Automated Data Collection

Recently developed hardware for automated data collection provides the opportunity for the simulation input modeling community to assume a leadership role in the analysis of the data collected in this fashion. I will cite two examples of hardware from the nonsimulation community in the paragraphs below.

As documented at <www.factoryware.com>, the *FactoryPulse System* is a rapid deployment productivity analysis software package. This system is used to analyze cycle time, down time, capacity, utilization, etc. for a discrete-manufacturing system. The system sets up in less than 30 minutes and contains its own statistical analysis package that can be used to analyze data that is collected on the system of interest. Although the analysis software contains graphical measures (e.g., pie charts), none of the typical simulation input modeling concerns are addressed in the data analysis portion of the system.

A second hardware example is the JAMAR hand-held counting board and associated PETRA (Professional Engineers Traffic Reporting and Analysis) software produced by JAMAR Technologies, Inc. The counting board con-

tains buttons keyed to turning movement (left-hand turn, straight, right-hand turn) associated with a two-way intersection. This hardware allows the collection of time and turning movement data that would be impossible to collect with a stopwatch and clipboard. Although the analysis software is capable of producing empirical CDFs associated with various measures of performance on an intersection, none of the typical input modeling tools (e.g., parametric distributions) are included.

The input modeling community could provide a valuable link between hardware of this type and the simulation modeling community at large.

6.3 Future Research

I believe that there are still fundamentally important, but unaddressed problems in input modeling. I will give two examples.

The first example considers the analysis of service-time data for a server capable of processing multiple jobs simultaneously. Examples of such servers include web-site servers and chefs. As the jobs arrive to the server, the service time will increase once a particular threshold (which may not be given to the data analyst) is reached. The problem is further complicated if the jobs are of varying size, as in the case of the web-site server. It is complicated even further if the jobs need to be sequenced so that a subset of the jobs need to be completed at approximately the same time. The proper analysis of the record of such a server in order to construct a discrete-event simulation input model is a non-trivial task which has not yet been addressed in the literature.

The second example comes from a conversation that I had with Steve Chick at last year's Winter Simulation Conference. Traditional simulation output analysis assumes that the simulation model is "correct" in the sense that the stochastic input model being used accurately depicts the random elements of the system of interest. In practice, this is virtually *never* the case. Consider the simulation of an $M/M/1$ queue using an estimated arrival rate of $\lambda = 1$ customer per minute and an estimated service rate of $\mu = 10/9$ customers per minute. Assume that the exponential assumptions are appropriate. One would certainly have much less faith in the output analysis of such a model if $n = 10$ interarrival times and $m = 12$ service times were collected than if $n = 1000$ interarrival times and $m = 1200$ service times were collected.

This explicit recognition of sampling error in the input models and how it can be incorporated in output analysis is an emerging research topic. I am presently working on this topic, assessing the sampling variability effects in input modeling in discrete-event simulation, with two William & Mary students, Rob McGregor and Matt Duggan. I will end with some preliminary results associated with this research.

Assume that the true system of interest is an $M/M/1$ queue with arrival rate λ, service rate μ, and traffic intensity $\rho = \frac{\lambda}{\mu}$. For the discussion here, assume the arbitrary values of $\lambda = 1$ and $\mu = \frac{10}{9}$ for the two rate parameters. The first question that can be addressed here is whether the expected steady-state queue length is infinite when λ and μ are estimated from data. When $\rho \geq 1$, the steady-state expected queue length is infinite.

Assume that we sample n exponential(λ) interarrival times, $X_1, X_2, \ldots, X_n$. The estimated mean interarrival time from the n sampled times is $\overline{X}$, and $1/\overline{X}$ is the estimated arrival rate. The distribution of $\overline{X}$ is Erlang.

Similarly, assume that we sample m exponential(μ) service times, $Y_1, Y_2, \ldots, Y_m$. The estimated mean service time is $\overline{Y}$ and the estimated service rate is $1/\overline{Y}$. The distribution of $\overline{Y}$ is also Erlang.

When the interarrival rate exceeds the service rate, the expected queue length will not be finite since jobs are arriving faster than they are being serviced. It is useful to know what the probability is of this occurring, e.g. $P(\overline{X} < \overline{Y}) = \int_0^\infty \int_{\overline{x}}^\infty f_{\overline{x}}(\overline{x}) f_{\overline{y}}(\overline{y})\, d\overline{y}\, d\overline{x}$.

In order to see how collecting more service times or interarrival times will affect the probability of the queue eventually growing without bound, the probabilities for values of n and m adjacent to $n = 12$ and $m = 10$ (two arbitrary sample sizes) are shown in Table 1.

Table 1: Probability of Steady-State Infinite Expected Queue Length

	$m = 9$	$m = 10$	$m = 11$
$n = 11$	0.4025	0.4031	0.4035
$n = 12$	0.3983	0.3987	0.3990
$n = 13$	0.3946	0.3949	0.3950

A seemingly counter-intuitive notion in Table 1 is that $P(\overline{X} < \overline{Y})$ increases in m. We checked this result for larger values of m and found that the probability does begin to decrease eventually. As expected, the probabilities decrease as n increases.

The distribution of the delay times is another output statistic that can be considered. We begin by looking at the delay time of the third customer, D_3. Kelton (1985) has computed the expected delay times of the k^{th} customer for a standard $M/M/1$ queue with fixed λ and μ values. Using Kelton's formula, $E[D_3]$, the expected delay of the third customer is

$$\frac{\lambda}{\mu^2}\left[\frac{1 + 4(\frac{\lambda}{\mu}) + 2(\frac{\lambda}{\mu})^2}{(\frac{\lambda}{\mu} + 1)^3}\right].$$

For the values chosen, $\lambda = 1$ and $\mu = \frac{10}{9}$, using the equation above results in $E[D_3] \cong 0.7345$, which was double-checked by 5,000,000 simulation replications times

with an average delay of the third customer being 0.7344, confirming the validity of the equation and the implementation. In addition, Kelton's formula was checked for several other values of ρ.

In place of λ in the equation, we use the distribution of $1/\overline{X}$ and in place of μ in the equation, we use the distribution of $1/\overline{Y}$. However, the problem proved to be too difficult to be solved analytically, so the distribution of the third delay has not been determined.

Since the analytic solution is not tractable, we simulated the third delay time with the estimated parameters $1/\overline{X}$ and $1/\overline{Y}$ rather than the fixed values for λ and μ. The values for $\overline{X}$ and $\overline{Y}$ are computed for $n = 12$ and $m = 10$ on each replication. The simulation is then run using $\overline{X}$ and $\overline{Y}$, and the delay of the third customer is calculated. For 1,000,000 simulation replications, the average delay of the third customer for the estimated parameters was 0.7853. This was compared to the average delay with the fixed parameters being 0.7345, which is about a 6.9% increase. The simulation also showed that there are slightly more zero delay times with the estimated parameters.

7 BRUCE W. SCHMEISER

At the 1991 Winter Simulation Conference in Phoenix I first heard an ingenious idea of how to deal with practitioner uncertainty about the model. During an informal conversation, Lee Schruben discussed the idea of expanding the simulation model to include practitioner uncertainty. For example, if a homogeneous arrival rate λ were not known with certainty, then the rate could be obtained by sampling from a distribution. The distribution would have a large variance if practitioner uncertainty were great and zero variance if practitioner certainty exists. I forget the original version of the idea: perhaps each replication would use one randomly generated value of λ. Since 1991 various versions of the idea have been suggested by Lee and others.

The concept that I wish to discuss is the general attempt to give the practitioner a sense of how practitioner uncertainty about the model affects the point estimator and its standard error, with the interpretation that the standard error now reflects both sampling error and modeling error. Although measuring the effect of practitioner uncertainty about the model is certainly an important issue, I think that using a single measure for these two sources of errors is not a good idea.

In this discussion, which elaborates on a point made in Schmeiser (2001), I assume that the purpose of the simulation experiment is to estimate an unknown performance measure θ_0 from an unknown true model M_0. Further, I assume that other sources of error are negligible: Random-number and random-variate generators are perfect, numbers are represented on a computer as real numbers, the simulation code has no bugs, and point estimators have zero bias (including from an initial transient).

A standard simulation experiment designed to estimate θ_0 then has two sources of error. First, rather than using model M_0, it might use model M with associated performance measure $\theta \neq \theta_0$, which we define to be modeling error. Second, the point estimator $\widehat{\theta}$ has sampling error, which we measure by its variance (or its square root, standard error). Typically the variance is $O(n^{-1})$, where n is a measure of computational effort; we assume only that the variance goes to zero as n goes to infinity.

In the standard simulation experiment, we are using $\widehat{\theta}$, which has expected value θ, to estimate θ_0. Consider the usual decomposition of mean squared error (mse) into squared bias and variance. Modeling error leads to bias $\theta - \theta_0$. Sampling error leads to variance $\mathrm{V}(\widehat{\theta})$. If we are to combine measures of sampling error and modeling error, then mse would be an acceptable single measure. Asymptotically for large values of n, mse goes to squared bias (as the variance goes to zero). Because θ_0 is unknown, the modeling-error bias is unknown, and therefore the quality of the point estimator $\widehat{\theta}$ is measured by standard error alone.

All is well as long as the practitioner understands that the error is with respect to θ and not θ_0; that is, modeling error is not a consideration in the standard simulation experiment. To the extent that simulation is only a method to analyze the model M, bias due to modeling error is irrelevant. Modeling error is important, though, so interest in measuring its effect is understandable.

The appeal of including practitioner model uncertainty in M is the hope that the increase in standard error will be (approximately) equal to the unknown bias. My primary concern is that I see no reason for such an equality to occur. Consider the following two points.

First, the equality needs to occur for all run lengths n, so any reasonable version needs to reflect the fundamental difference that sampling error decreases with additional simulation sampling while additional sampling has no impact on modeling error.

Second, a practitioner's uncertainty may not reflect reality. A practitioner could be certain, but wrong; a practitioner could be correct, but uncertain.

Another concern is that a simulation experiment that includes model uncertainty could have substantially more bias than the simpler experiment based on a fixed model. Suppose that the model is a steady-state $M/M/1$ queue and that a bivariate distribution is placed on the arrival rate λ and service rate μ. Care must be taken to ensure that the resulting performance measure θ is not infinite, in which case the associated modeling bias is infinite. Even if θ is finite, the bias easily can be larger than that of the standard experiment.

My conclusion is that model error should be treated separately from sampling error. Good methods for esti-

mating standard errors exist. Similarly, good methods for estimating sensitivity to modeling errors exist. Maintaining the distinction between these two sources of error simplifies and clarifies both experimentation and interpretation of results.

8 LEE W. SCHRUBEN

8.1 Introduction

I will present arguments for the following propositions:

1. "Real world" data is not important to the success of many simulation studies.
2. Fitting distributions to data for driving simulations is fundamentally unsound.
3. Beta variates are the only scalar random variables we need for most practical simulation input modeling.
4. Resampling nonhomogeneous renewal processes is a promising way to model time dependency.

Simulation input is defined narrowly here as the data generated by stochastic models that describe the behavior of the *external environment* of the system being simulated. The conventional approach to input modeling involves collecting and validating real world data, statistically selecting probability distributions that fit the data and then generating input random variables from these distributions to model the system environment.

The importance placed on simulation input modeling is reflected by the considerable amount of space devoted to this topic in simulation textbooks as well as the large number of journal articles and the proliferation of distribution fitting software. Indeed, input modeling typically consumes much of the resources and time in a simulation project.

I would like to consider the propositions that input modeling is really not all that important to the success of a simulation study and that the value of "real world" data is highly overrated.

8.2 Reality?

To place the value of real world data in perspective it might be helpful to remember the following *Five Dastardly D's of Data* (Schruben and Schruben 2001). Data can be

1. Distorted: Data might be defined inconsistently or poorly communicated. Even if communicated correctly, the values of observations may have been mistranslated, scaled, or simply recorded incorrectly.
2. Dated: The data may be from a system that has or soon will be changed. For example: factory data may have been gathered for an older production process or using last year's product mix and demands.
3. Deleted: Observations may be missing from the data set. This might be because the data was collected over a finite time interval during which important but rare events simply did not occur. For example: medical trial data might be censored by patients dropping out of the study for various and unknown reasons.
4. Dependent: Data may be summarized (i.e., only daily averages are reported) removing critical cycles or other trends. Data might be collected under unusual conditions or at systematic times, creating or concealing relationships within and among the sets of observations. Fitting a data set to a scalar probability distribution removes dependencies, maybe the most useful information in the data set.
5. Deceptive: Any of the above four data problems might be intentional.

Even if we have a very large set of valid system data, a fundamental problem with using real system data remains: once the data is observed, it is *old*. Most simulation studies are motivated by a desire to learn what would happen if we were to *change* a system. By modeling the input to our experiments using data from the old system, we implicitly assume that the environment in which the system lives is stationary over the time between data collection and the end of the study; this is probably not the case.

We are also assuming that the system we are studying is closed to its environment. For example, if we were to improve customer service that we would not attract more customers. If we assume that changing the system will have no significant impact on its environment, we have to ask ourselves: why are we considering changing it? Our so-called, "what if?" experiments are more aptly "as if!" experiments.

I am not suggesting that real system data is useless. One justification given for collecting is to verify the simulation code. Here I would prefer driving the simulation with the original data traces (preserving dependencies and reducing the variance between real and simulated performance) to fitting scalar distributions to the data.

A very good reason for using real system data is to estimate the statistical performance of the current system. This sounds silly since we usually can directly observe the performance of the current system. However, the goal of many simulation studies is to improve an existing system, and doing nothing is the default decision. A case needs to be made for any proposed changes. Accurately estimating the effects of proposed changes must be relative to the real performance of the current system. The central question is

not whether or not the proposed change is an improvement but rather whether is it *enough* of an improvement to justify the costs of the changes. Accurately estimating the difference between the simulated and the actual performance of the current system is an imperfect way to calibrate a simulation, but it is better than nothing. How to do this calibration is an interesting topic for research.

Probably the best reason for encouraging the collection of real world data is to force simulation practitioners to observe the systems they are modeling. Collecting and analyzing data gives them something to do while learning how the system works.

8.3 Reality Check?

I feel the best approach for estimating the current performance of a real system is to construct percentile confidence intervals using the two-step empirical resampling methods advocated in Russ Barton's contribution to this panel. This approach retains some of the shortcomings of fitting distributions to data (such as ignoring dependencies and nonstationarity), but it does account for the uncertainty of not knowing the true input distribution or its parameters.

One of these methods for generating independent scalar random variables, a Bayesian bootstrap, has a great deal of appeal since it uses only the information in the data. No distributions are actually fit to the data.

Consider some important real world random variable, X. Rather than estimating its unknown distribution function, $F_X(x)$ (and thereafter treating this as a known function), we can implicitly randomize this function by observing real values of its horizontal coordinates and generating its vertical coordinates from the appropriate distribution.

The random variable X is observable in the real system and we know that the distribution of $Y = F_X(X)$ is iid uniform. We can implicitly sample points on F by observing n ordered values of X as its horizontal coordinates, $x_{[1]}, x_{[2]}, \ldots, x_{[n]}$. Then for *every* new simulation run, we generate a *new* set of n vertical coordinates for F, $y_{[1]}, y_{[2]}, \ldots, y_{[n]}$ as n uniform order statistics. To simulate a new value of X we generate an iid uniformly distributed pseudorandom number, U, and interpolate the inverse of F between $x_{[i]}$ and $x_{[i+1]}$ where U falls between $y_{[i]}$ and $y_{[i+1]}$ (search starting at $i = \lfloor nU \rfloor$). Since we are resampling, percentile confidence intervals should be used.

The coverage of the confidence intervals for this approach as well as for other resampling approaches dramatically dominates that for conventional methodology when the simulation runs are long enough so that the error due to a finite real world sample is less important than the error due to a finite simulation run length. This is usually the case since simulated observations are much cheaper than real world data.

8.4 Beta Testing

For simulating proposed systems, it is unreasonable to assume that the environment from which we collected our real world data will not change. Here the question is: how might the environment react to our changing the system to make the change worthwhile … or to break it? This question is answered by treating simulation input as another set of factors in our simulation experimental design. We perform sensitivity analysis on the input to try to find out where our proposed system will best operate (considering both issues of optimality and robustness).

For sensitivity analysis, beta variates are the only ones we need for most simulation studies. The four parameters of a beta distribution (including location and range) offer plenty of flexibility for input sensitivity analysis as shown in Figure 2 from Schruben and Schruben (2001). This distribution also has the advantage of having finite support.

8.5 Nonhomogeneous Renewal Processes

A model that may be useful for sensitivity analysis to nonstationary simulation environments is what I will call a nonhomogeneous renewal process. This is the obvious generalization of a nonhomogeneous Poisson Process (NHPP).

The conventional approach for modeling a NHPP, analogous to fitting distribution functions to iid data, is to estimate the cumulative intensity function and treat it as known throughout the simulation study. This approach, of course, does not account for the uncertainty in the cumulative intensity function.

The idea is to resample the process without explicitly estimating the cumulative intensity function. We first sample the total number of events independently for every simulated cycle using a resampling scheme like that in Barton and Schruben (2001). Aggregated cycle data (e.g. total daily demand) is likely to be readily available whereas detailed event times probably are not. Then, conditioned on the number of observations in a simulated cycle, we have exactly the same situation as we did for the Bayesian bootstrap. The cumulative intensity function maps a nonstationary Poisson process along time on the horizontal axis into a stationary Poisson process along the vertical axis with a rate of 1. The horizontal coordinates of the randomized cumulative intensity function are the observed real event times in (superimposed) cycles. The vertical coordinates are the order statistics from the appropriate uniform distribution—relying on the fact that given the total number of events in a homogeneous Poisson process cycle the actual event times have a joint uniform distribution.

Now consider generalizing this to other nonhomogeneous stochastic processes. Assume the existence of a function that maps nonstationary (real) time into stationary time like the cumulative intensity function does for the

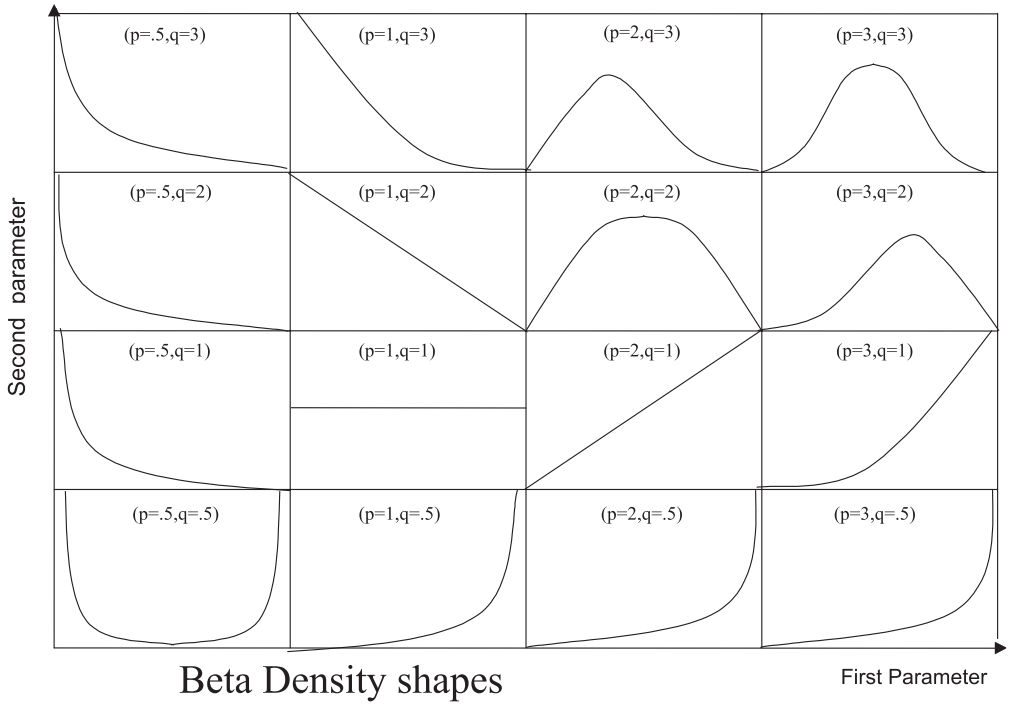

Figure 2: Beta "Shape Matrix"

NHPP. If we knew this function, then we could generate a stationary renewal process along the vertical (stationary time) axis and map the nonstationary event times along the horizontal (real time) axis via the inverse of this function. If the stationary point process (conditioned on the number of events in the simulated cycle) has events distributed like uniform random variables we have the NHPP. Since we are dealing with order statistics and the uniform distribution is a special case of the beta distribution, using a nonhomogeneous beta process suggests itself as an obvious generalization for sensitivity analysis.

Finally, for small samples, it may be worthwhile to randomize the nonstationary empirical cumulative intensity function proposed in Leemis (1991). The goal would be to randomly resample a *new* empirical cumulative intensity function for *every* simulated cycle. This might partially account for the uncertainty in not knowing this function. Ideally we would simulate using randomly resampled functions like in Figure 7 in Arkin and Leemis (2000) rather than using the estimated function like in Figure 8 for every simulated cycle.

8.6 Acknowledgments

I have benefited from many conversations on this topic with the other members of this panel, in particular Russ Barton, as well as with Jack Kleijnen, George Fishman, Peter Lewis, Bob Sargent, Bennett Fox and the late Alan Pritsker. Research support from the Semiconductor Research Corporation and the National Science Foundation is greatly appreciated.

REFERENCES

Arkin, B. L., and L. M. Leemis. 2000. Nonparametric estimation of the cumulative intensity function for a nonhomogeneous Poisson process from overlapping realizations. *Management Science* 46 (7): 989–998.

Barton, R. R., and L. W. Schruben. 1993. Uniform and bootstrap resampling of empirical distributions. In *Proceedings of the 1993 Winter Simulation Conference*, ed. G. W. Evans, M. Mollaghasemi, W. E. Biles, and E. C. Russell, 503–508. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.

Barton, R. R., and L. W. Schruben. 2001. Resampling methods for input modeling. In *Proceedings of the 2001 Winter Simulation Conference*, ed. B. A. Peters, J. S. Smith, D. J. Medeiros and M. W. Rohrer, 372–378. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers. Available via <www.informs-cs.org> [full address is <http://www.informs-cs.org/wsc01papers/048.PDF>; accessed July 14, 2002].

Cario, M. C., and B. L. Nelson. 1996. Autoregressive to anything: time-series input processes for simulation. *Operations Research Letters* 19:51–58.

Cario, M. C., and B. L. Nelson. 1997. Modeling and generating random vectors with arbitrary marginal dis-

tributions and correlation matrix. Technical Report, Department of Industrial Engineering and Management Sciences, Northwestern University, Evanston, Illinois.

Cheng, R. C. H. 1994. Selecting input models. In *Proceedings of the 1994 Winter Simulation Conference*, ed. J. D. Tew, M. S. Manivannan, D. A. Sadowski and A. F. Seila, 184–191. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.

Cheng, R. C. H. 1998. Bayesian model selection when the number of components is unknown. In *Proceedings of the 1998 Winter Simulation Conference*, ed. D. J. Medeiros, E. F. Watson, J. S. Carson and M. S. Manivannan, 653–659. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers. Available via <www.informs-cs.org> [full address is <http://www.informs-cs.org/wsc98papers/087.PDF>; accessed July 14, 2002].

Cheng, R. C. H. 2001. Analysis of simulation experiments by bootstrap resampling. In *Proceedings of the 2001 Winter Simulation Conference*, ed. B. A. Peters, J. S. Smith, D. J. Medeiros, and M. W. Rohrer, 179–186. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers. Available via <www.informs-cs.org> [full address is <http://www.informs-cs.org/wsc01papers/021.PDF>; accessed July 14, 2002].

Chick, S. E. 2000. Bayesian methods for simulation. In *Proceedings of the 2000 Winter Simulation Conference*, ed. J. Joines, R. Barton, P. Fishwick, and K. Kang, 109–118. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers. Available via <www.informs-cs.org> [full address is <http://www.informs-cs.org/wsc00papers/018.PDF>; accessed July 14, 2002].

De Finetti, B. 1990. *Theory of probability*. volume 2. Chichester: John Wiley & Sons.

Deler, B., and B. L. Nelson. 2001. Modeling and generating multivariate time series with arbitrary marginals and autocorrelation structures. In *Proceedings of the 2001 Winter Simulation Conference*, ed. B. A. Peters, J. S. Smith, D. J. Medeiros, and M. W. Rohrer, 275–282. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers. Available via <www.informs-cs.org> [full address is <http://www.informs-cs.org/wsc01papers/034.PDF>; accessed July 14, 2002].

Draper, D. 1995. Assessment and propagation of model uncertainty (with discussion). *Journal of the Royal Statistical Society, Series B* 57 (1): 45–97.

Ghosh, S., and S. G. Henderson. 2001. Chessboard distributions. In *Proceedings of the 2001 Winter Simulation Conference*, ed. B. A. Peters, J. S. Smith, D. J. Medeiros, and M. W. Rohrer, 385–393. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers. Available via <www.informs-cs.org> [full address is <http://www.informs-cs.org/wsc01papers/050.PDF>; accessed July 14, 2002].

Ghosh, S., and S. G. Henderson. 2002. Chessboard distributions and random vectors with specified marginals and covariance matrix. *Operations Research* to appear.

Gilks, W. R., S. Richardson, and D. J. Spiegelhalter. 1996. *Markov chain Monte Carlo in practice*. London: Chapman and Hall.

Henderson, S. G., and A. Mason. 1999. Estimating ambulance requirements in Auckland, New Zealand. In *Proceedings of the 1999 Winter Simulation Conference*, ed. P. A. Farrington, H. B. Nembhard, J. Evans, D. Sturrock, 1670–1674. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers. Available via <www.informs-cs.org> [full address is <http://www.informs-cs.org/wsc99papers/241.PDF>; accessed July 14, 2002].

Kelton, W. D. 1985. The transient behavior of the $M/M/s$ queue, with implications for steady-state simulation. *Operations Research* 33:378–395.

Kleijnen, J. P. C., R. C. H. Cheng, and B. Bettonvil. 2001. Validation of trace-driven simulation models: Bootstrap tests. *Management Science* 47:1533–1538.

Kurowicka, D., and R. M. Cooke. 2002. The vine-copula method for representing high dimensional dependent distributions. In *Proceedings of the 2002 Winter Simulation Conference*, ed. E. Yücesan, C. H. Chen, J. M. Charnes, J. L. Snowdon. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.

Law, A. M., and W. D. Kelton. 2000. *Simulation modeling and analysis*. 3d ed. New York: McGraw-Hill.

Lee, S. H., and P. W. Glynn. 1999. Computing the distribution function of a conditional expectation via Monte Carlo: discrete conditioning spaces. In *Proceedings of the 1999 Winter Simulation Conference*, ed. P. A. Farrington, H. B. Nembhard, J. Evans, D. Sturrock, 1654–1663. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers. Available via <www.informs-cs.org> [full address is <http://www.informs-cs.org/wsc99papers/239.PDF>; accessed July 14, 2002].

Leemis, L. M. 1991. Nonparametric estimation of the cumulative intensity function for a nonhomogeneous Poisson process. *Management Science* 37 (7): 886–900.

Leemis, L. 2001. Input modeling techniques for discrete-event simulations. In *Proceedings of the 2001 Winter Simulation Conference*, ed. B. A. Peters, J. S. Smith, D. J. Medeiros and M. W. Rohrer, 62–73. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers. Available via <www.informs-cs.org> [full address is

<http://www.informs-cs.org/wsc01papers/009.PDF>; accessed July 14, 2002].

Melamed, B., J. R. Hill, and D. Goldsman. 1992. The TES methodology: Modeling empirical stationary time series. In *Proceedings of the 1992 Winter Simulation Conference*, ed. J. J. Swain, D. Goldsman, R. C. Crain, and J. R. Wilson, 135–144. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.

Nelson, B. L., and M. Yamnitsky. 1998. Input modeling tools for complex problems. In *Proceedings of the 1998 Winter Simulation Conference*, ed. D. J. Medeiros, E. F. Watson, J. S. Carson and M. S. Manivannan, 105–112. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers. Available via <www.informs-cs.org> [full address is <http://www.informs-cs.org/wsc98papers/014.PDF>; accessed July 14, 2002].

Raftery, A. E. 1995. Bayesian model selection in social research (with discussion by Andrew Gelman, Donald B. Rubin, and Robert M. Hauser, and a rejoinder). In *Sociological methodology 1995*, ed. P. V. Marsden. Cambridge, Massachusetts: Blackwells.

Schmeiser, B. W. 1999. Advanced input modeling for simulation experimentation. In *Proceedings of the 1999 Winter Simulation Conference*, ed. P. A. Farrington, N. B. Nembhard, D. T. Sturrock, and G. W. Evans, 110–115. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers. Available via <www.informs-cs.org> [full address is <http://www.informs-cs.org/wsc99papers/014.PDF>; accessed July 14, 2002].

Schmeiser, B. W. 2001. Some myths and common errors in simulation experiments. In *Proceedings of the Winter Simulation Conference*, ed. B. A. Peters, J. S. Smith, D. J. Medeiros and M. W. Rohrer, 39–46. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers. Available via <www.informs-cs.org> [full address is <http://www.informs-cs.org/wsc01papers/006.PDF>; accessed July 14, 2002].

Schruben, D., and L. W. Schruben. 2001. *Graphical simulation modeling using SIGMA*. 4th ed. Berkeley, California: Custom Simulations.

Young, M. R., and Lenk, P. J. 1998. Hierarchical Bayes methods for multifactor model estimation and portfolio selection. *Management Science* 44 (11): S111-S124.

Zouaoui, F., and J. R. Wilson. 2001. Accounting for input model and parameter uncertainty in simulation. In *Proceedings of the 2001 Winter Simulation Conference*, ed. B. A. Peters, J. S. Smith, D. J. Medeiros, and M. W. Rohrer, 290–299. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers. Available via <www.informs-cs.org> [full address is <http://www.informs-cs.org/wsc01papers/036.PDF>; accessed July 14, 2002].

AUTHOR BIOGRAPHIES

RUSSELL R. BARTON is a Professor and Associate Dean for Research and Ph.D./M.S. Programs in The Smeal College of Business Administration at The Pennsylvania State University. He has a B.S. degree in Electrical Engineering from Princeton and M.S. and Ph.D. degrees in Operations Research from Cornell. Before entering academia, he spent twelve years in industry. He is the Vice President/President-Elect for the INFORMS–College on Simulation. His research interests include applications of statistical and simulation methods to system design and to product design, manufacturing, and delivery. His email address is <rbarton@psu.edu>.

RUSSELL C. H. CHENG is Professor and Head of Operational Research at the University of Southampton. He has an M.A. and the Diploma in Mathematical Statistics from Cambridge University, England. He obtained his Ph.D. from Bath University. He is a former Chairman of the U.K. Simulation Society, a Fellow of the Royal Statistical Society, Member of the Operational Research Society. His research interests include: variance reduction methods and parametric estimation methods. He is Joint Editor of the *IMA Journal of Management Mathematics*. His e-mail and web addresses are <R.C.H.Cheng@maths.soton.ac.uk> and <www.maths.soton.ac.uk/staff/Cheng>.

STEPHEN E. CHICK is Associate Professor of Technology Management at INSEAD in Fontainebleau, France. Problems in manufacturing, operations and health care motivate his research in simulation, Bayesian statistics, decision analysis, and computational methods in statistics. He is an associate editor of *Management Science* and Secretary/Treasurer of the INFORMS–College on Simulation. He can be contacted at <stephen.chick@insead.edu>.

SHANE G. HENDERSON is an assistant professor in the School of Operations Research and Industrial Engineering at Cornell University. He has previously held positions in the Department of Industrial and Operations Engineering at The University of Michigan and the Department of Engineering Science at the University of Auckland. He is an associate editor for the *ACM Transactions on Modeling and Computer Simulation*, *Operations Research Letters*, and *Mathematics of Operations Research*, and the *Newsletter* editor for the INFORMS–College on Simulation. His research interests include discrete-event simulation, queueing theory, and scheduling problems. His e-mail address is <shane@orie.cornell.edu>, and his web page is <www.orie.cornell.edu/~shane>.

AVERILL M. LAW is President of Averill M. Law & Associates, a company specializing in simulation consulting, training, and software. He has been a simulation consultant to many organizations and has presented more than 345 simulation short courses in 17 countries. He has written or coauthored papers and books on simulation, operations research, and statistics, including the textbook *Simulation Modeling and Analysis*. He developed the ExpertFit distribution-fitting software and he wrote a regular column on simulation for *Industrial Engineering* magazine. Previously, he was a tenured faculty member at the University of Wisconsin-Madison and the University of Arizona. He has a Ph.D. in industrial engineering and operations research from the University of California at Berkeley. His E-mail address is <averill@ix.netcom.com>, and his Web site is <www.averill-law.com>.

LAWRENCE M. LEEMIS is a professor and chair of the Mathematics Department at The College of William & Mary. He received his B.S. and M.S. degrees in Mathematics and his Ph.D. in Industrial Engineering from Purdue University. He has also taught at Baylor University, The University of Oklahoma, and Purdue University. His consulting, short course, and research contract work includes contracts with AT&T, NASA/Langley Research Center, Delco Electronics, Department of Defense (Army, Navy), Air Logistic Command, ICASE, Komag, Federal Aviation Administration, Tinker Air Force Base, Woodmizer, Magnetic Peripherals, and Argonne National Laboratory. His research and teaching interests are in reliability and simulation. He is a member of ASA, IIE, and INFORMS. His email address is <leemis@math.wm.edu>, and his web address is <www.math.wm.edu/~leemis>.

BRUCE W. SCHMEISER is a professor in the School of Industrial Engineering at Purdue University. His interests lie in applied operations research, with emphasis in stochastic models, especially the probabilistic and statistical aspects of stochastic simulation. He is an active participant in the Winter Simulation Conference, including being Program Chair in 1983 and chairing the Board of Directors during 1988–1990.

LEE W. SCHRUBEN is Chancellor's Professor and Department Chairman of the Department of Industrial Engineering and Operations Research at the University of California at Berkeley. Before joining the Berkeley faculty, he was at Cornell University in the School of Operations Research and Industrial Engineering. His email address is <schruben@ieor.berkeley.edu>.

JAMES R. WILSON is Professor and Head of the Department of Industrial Engineering at North Carolina State University. He received a B.A. degree in mathematics from Rice University, and he received M.S. and Ph.D. degrees in industrial engineering from Purdue University. His current research interests are focused on the design and analysis of stochastic simulation experiments. He has served as a departmental editor of *Management Science* and as an area editor of *ACM Transactions on Modeling and Computer Simulation*. He has also held various offices in INFORMS–College on Simulation; and he currently serves as a corepresentative of that organization to the Board of Directors of the Winter Simulation Conference. He is a member of ASA, IIE, and INFORMS. His e-mail address is <jwilson@eos.ncsu.edu>, and his web page is <www.ie.ncsu.edu/jwilson>.