Proceedings of the 2003 Winter Simulation Conference
S. Chick, P. J. Sánchez, D. Ferrin, and D. J. Morrice, eds.

INPUT MODEL UNCERTAINTY: WHY DO WE CARE AND WHAT SHOULD WE DO ABOUT IT?

Shane G. Henderson

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

ABSTRACT

An input model is a collection of distributions together with any associated parameters that are used as primitive inputs in a simulation model. Input model uncertainty arises when one is not completely certain what distributions and/or parameters to use. This tutorial attempts to provide a sense of why one should consider input uncertainty and what methods can be used to deal with it.

1 INTRODUCTION

Consider the following artificial examples of decision problems where simulation can play a role. (Any resemblance to real people is entirely coincidental.)

Example 1: Bruce Lee runs a bakery that is open from 6am till 3pm every day. During that time customers arrive according to a Poisson process at rate Λ. The rate Λ varies from day to day in an i.i.d. fashion, and on any given day is gamma distributed with parameters $\alpha > 0$ and $\beta > 0$, so that the density of Λ at $x > 0$ is proportional to $x^{\alpha-1}e^{-x/\beta}$. A single staff member can serve a customer in an amount of time that is exponentially distributed with mean μ^{-1}. Service times are independent of one another and of the arrival process, and successive days are independent of one another. Bruce wants to decide how many staff are needed to serve customers so that over the long run at least 90% of customers wait 1 minute or less in line before being served.

Example 2: Steve Russell runs a wine store that is open from 11am till 8pm every day. During that time customers arrive according to a Poisson process at rate λ. The rate λ is fixed, but not known with certainty. However, the uncertainty is well modelled by assuming that λ is gamma distributed with parameters α and β (the same values as at Bruce Lee's bakery). A single staff member can serve a customer in an amount of time that is exponentially distributed with mean μ^{-1}(again the same value as at Bruce Lee's bakery). Service times are independent of one another and of the arrival process, and successive days are independent of one another. Steve wants to decide how many staff are needed to serve customers so that over the long run at least 90% of customers wait 1 minute or less in line before being served.

The structure of these two systems, being multiserver queues, is the same. Furthermore, customers arrive according to a Poisson process on any given day, and have the same service time distribution. The difference lies in the uncertainty associated with the arrival rate of the Poisson process. In Example 1 this uncertainty takes the form of a varying arrival rate, where the arrival rate varies *in a known fashion*. In Example 2 the arrival rate is the same from day to day, but *we do not know the exact value*.

Are these two problems the same? In other words, can we analyze the systems using identical performance measures and interval-estimation procedures?

I believe that the answer is no.

To understand why, let us consider the calculation of the long-run fraction of customers who wait for 1 minute or less. Suppose that we observe one of the stores for ℓ days. Let N_i and S_i be the number of customers that arrive to the store, and the number of customers that reach service in 1 minute or less, on day i respectively, $i = 1, \ldots, \ell$. Then the fraction of customers that wait for 1 minute or less over the ℓ days is

$$\frac{\sum_{i=1}^{\ell} S_i}{\sum_{i=1}^{\ell} N_i}.$$

As ℓ gets large, this ratio converges to

$$\frac{E S_1}{E N_1},$$

as can be seen by dividing both the numerator and denominator by ℓ and applying the strong law of large numbers to both.

Now let us specialize to Example 1. Conditioning on the arrival rate Λ and using a standard result for Poisson processes we find that

$$EN_1 = EE[N_1|\Lambda] = E[9\Lambda] = 9\alpha\beta,$$

where the 9 comes from the fact that the shop is open for 9 hours. There are several ways to compute or approximate ES_1. One could apply queueing theory, but we will instead use simulation. Specifically, one can imagine simulating the bakery operations for a large number of days ℓ. On day i we first generate a realization Λ_i of Λ from its distribution, and then simulate a multiserver queue with arrival rate Λ_i for the remainder of the day. The simulated random variables $(S_1, \ldots, S_\ell)$ are then i.i.d. random variables with finite variance, and so we can construct a confidence interval for ES_1 in the usual fashion. We see that we can proceed in exactly the fashion that we are used to in conducting simulation experiments. Interestingly, the situation is not as clear cut for Example 2.

Consider how we can compute EN_1 for Example 2. In this case there is no need to condition on λ since it is a deterministic quantity. From a standard result for Poisson processes, $EN_1 = 9\lambda$. But what is this value? We do not know for sure because we do not know the value of λ. We have the same problem with computing ES_1. We could use simulation to estimate it for any fixed value of λ, but what value of λ should be used? Should we pick a single value for λ? Or should we sample the value of λ prior to each day's operation as we did in Example 1? If we are to perform such sampling, then what should we report to the simulation user? A confidence interval as before? If so, how should such a confidence interval be interpreted?

The answers to these questions vary depending on who you ask, because this problem is a special case of the problem of input model uncertainty and there is no general agreement on how to proceed.

The general form of this problem may be phrased as follows. A simulation model relies on the specification of the distributions and associated parameters (these distributions could be multivariate) that serve as inputs to the model. Following the custom of several authors, we reserve the term *model uncertainty* to relate to the choice of a family of distributions (e.g., normal, exponential, Weibull), and *parameter uncertainty* to relate to the selection of parameters for those distributions (e.g., mean and variance for the normal distribution). The term *input model uncertainty* refers collectively to both problems.

This definition also encapsulates input models that are based on nonparametric methods such as empirical distribution functions. An empirical distribution function (or some smoothed version thereof) is a particular model choice. Parameter uncertainty then relates to the value of the distribution function at the observed points.

The problem is compounded by the fact that for any fixed input model, simulation can only report *estimates* of performance measures. In particular, the simulation is built from random variables that ensure that the resulting estimate is also random. This form of randomness is the one that we are very familiar with. It takes various names, depending on who you speak with, including statistical uncertainty, stochastic uncertainty, aleatory uncertainty and simulation uncertainty. This form of uncertainty can be contrasted with input model uncertainty as described above, which also has multiple names including structural uncertainty, subjective uncertainty, and epistemic uncertainty.

It is worth noting that while we are discussing this problem in the context of stochastic simulation, the problem is not unique to this field. For example, even if one were to apply queueing formulae to approximate the performance measure for Example 2 above, one must still deal with the issue of what to report, and how to decide when Steve Russell has enough staff members. With deterministic formulae one no longer has to deal with simulation uncertainty, but one still has to deal with input model uncertainty.

As a second example, the field of risk analysis has grappled with this issue for some time; see Helton (1996), Helton (1997), Helton and Davis (2003) and Oberkampf et al. (2003) for entry points to that literature, and below for the approach described in Helton (1996). Oberkampf et al. (2003) describe a wide variety of methods for dealing with input model uncertainty that draw from such fields as interval analysis, fuzzy set theory, possibility theory, evidence (Dempster-Shafer) theory, and imprecise probability theory. These methods are not included in this survey because I believe that the methods that *are* included are more appropriate for addressing input model uncertainty in simulation.

As a third example, there is now an area known as *robust optimization* that deals with optimization problems with constraints, where the parameters of the optimization problem (not the decision variables) are assumed to lie in an ellipsoid L say. These methods require that any decision be feasible with respect to the constraints for *any* choice of the parameters in L. Assuming the problem is of "minimize" type, they then minimize the maximum possible value of the objective, where the inner maximum is over the values of the parameters. Robust optimization methods are therefore quite conservative in their approach. Nevertheless, for many problems one does not see much of a deterioration in the optimal value that is reported, and the recommended solutions are far more robust to perturbations in the parameters than is the case for a solution generated assuming that the parameters take a single value. Much of the work in this area is devoted to developing efficient solution algorithms. See Ben-Tal and Nemirovski (1998), Ben-Tal and Nemirovski (2000) for details and examples.

Consider now the two questions in the title of this paper. First, why do we care? The answer to this question is well understood, and is discussed below. Second, what should we do about it? The answer to this question is less clear, and many answers have been proposed. Any method for dealing with input uncertainty must satisfy at least the following requirements.

- **Transparency** – The method should be understood by users.
- **Validity** – The method should be based on a firm statistical foundation that experts agree is reasonable.
- **Implementability** – The method should ideally be able to be applied to a range of problems without any need for expert intervention in each application.
- **Efficiency** – The method should not require an unduly large amount of computing time.

This paper surveys the methods that have been suggested for dealing with input uncertainty and is organized as follows. In §2 we answer the question of why we care. §3 establishes a framework that allows a concrete discussion of the various methods that have been proposed to deal with input uncertainty. In §4 we describe a standard method, that is standard in the sense that it has been well known and used for some time. Next, in §5 we survey some of the recently proposed methods. Some final remarks are offered in §6.

2 WHY DO WE CARE?

In this section we explore an example involving the M/M/1 queue. Our goal is to explain the motivation for explicitly addressing input model uncertainty. Our presentation is motivated by the example presented in Barton and Schruben (2001) and elaborated on in Barton et al. (2002), although we present the key ideas in a different manner. In particular, we work with parametric classes of distributions as opposed to empirical distribution functions, and consider the case where the simulation is "perfect", i.e., simulation error is 0. At the end of the section we discuss the example given in Barton and Schruben (2001) in a little more detail, partly to stress that the idealized assumptions of our example do not distort the key issues, although they do simplify some of the difficulties.

Example 3: Consider an M/M/1 queue with arrival rate λ_0 customers per hour and service rate μ_0 customers per hour. We assume that $\mu_0 = 10$ is known, but λ_0 is not. We take the (unknown) value of $\lambda_0 = 9$. We are interested in computing the expected steady-state sojourn time (time spent in queue and in service) in the system, which queueing theory gives as $f(\lambda_0)$, where

$$f(\lambda) = \begin{cases} (\mu_0 - \lambda)^{-1} & \text{if } \lambda < \mu_0, \\ \infty & \text{if } \lambda \geq \mu_0. \end{cases}$$

The case where $\lambda \geq \mu_0$ corresponds to an unstable system, so that in this case we take the performance to be ∞. The function f also depends on μ_0 but we ignore this dependence in what follows because our focus is on the unknown parameter λ_0. The true value of performance is $f(\lambda_0) = 1$.

In this example we do not use simulation but rather, for any value of λ, simply compute the function $f(\lambda)$. So here the function f takes the place of a simulation. One can view f as a zero-variance simulation, or the result from a simulation that runs for an infinite period of time. Even in this idealized setting the issue of input model uncertainty is nontrivial.

We assume that λ_0 must be estimated from interarrival time data. Suppose that we have $n \geq 1$ i.i.d. $\exp(\lambda_0)$ interarrival times $U_1, \ldots, U_n$. The maximum likelihood estimator of λ_0 from this data is $\hat{\lambda}_n = 1/\bar{U}_n$, the inverse of the sample mean. Hence, for any finite value of n, our estimate of λ_0 will not coincide with the true value of 9 with probability 1. The key question is whether this makes much difference.

Asymptotic theory ensures that for $n \geq 1$, the estimator $\hat{\lambda}_n$ is approximately normally distributed with mean λ_0 and variance λ_0^2/n. We will pretend that this distributional approximation is exact, and look at performance assuming this is the case.

Our estimator $\hat{\lambda}_n$ is normally distributed, so it has infinite right and left tails. Therefore, no matter how large n is, there is a positive probability that the queue is unstable. Furthermore, there is also a positive probability that $\hat{\lambda}_n$ is negative since we are assuming that it is *exactly* normally distributed! Of course, the case of a fitted negative arrival rate never arises in practice because of the form of our estimator of $\hat{\lambda}_n$. There is no real need to worry about either instability or negative arrival rates when n is large, since the chance of these events is then ridiculously small, as can be quantified by large deviations theory. We ignore the possibility of a negative arrival rate in what follows, but explicitly consider the possibility of an unstable queue.

Since $\hat{\lambda}_n$ is a random variable, the estimator $f(\hat{\lambda}_n)$ of the mean sojourn time is also a random variable. (In fact, it is an improper random variable since it takes the value ∞ with positive probability.) The randomness arises purely as a result of input model uncertainty. We can obtain its distribution by noting that for $x > 0$

$$\begin{aligned} P(f(\hat{\lambda}_n) \leq x) &= P((\mu_0 - \hat{\lambda}_n)^{-1} \leq x, \hat{\lambda}_n < \mu_0) \\ &= P(\mu_0 - \hat{\lambda}_n \geq x^{-1}, \hat{\lambda}_n < \mu_0) \\ &= P(\hat{\lambda}_n \leq \mu_0 - x^{-1}) \\ &= \Phi\left(\frac{\mu_0 - \lambda_0 - x^{-1}}{n^{-1/2}\lambda_0}\right), \end{aligned} \tag{1}$$

where Φ is the cumulative distribution function of a standard normal random variable. Here (1) follows since $\hat{\lambda}_n$ is normally distributed. Differentiating, we obtain the density of $f(\hat{\lambda}_n)^{-1}$ for $x > 0$ as

$$\frac{n^{1/2}}{\lambda_0 x^2}\phi\left(\frac{\mu_0 - \lambda_0 - x^{-1}}{n^{-1/2}\lambda_0}\right), \tag{2}$$

where ϕ is the density of a standard normal random variable. The density (2) is plotted for various values of n in Figure 1.

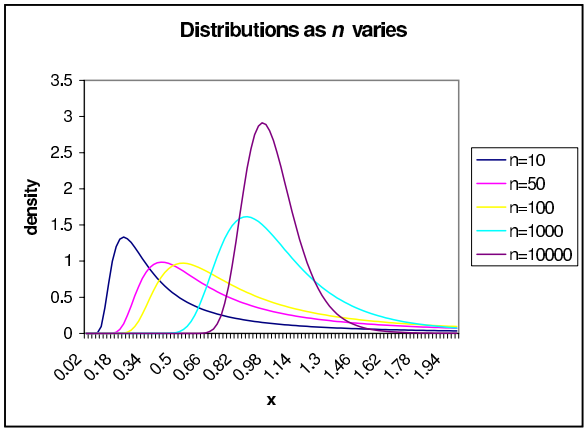

Figure 1: The Density of the Expected Steady-State Waiting Time for Various n

It is important to understand what these densities tell us. For any fixed value of n, the density gives a sense of what our "simulation" experiment could predict for the mean steady-state sojourn time. The height of the density, as always, gives a sense of how likely a given value is to occur. Some observations are in order.

1. As n increases, the densities shift to the right and concentrate around 1, indicating that for large n we are likely to predict a value very close to the correct value of 1. This is as expected because as n increases $\hat{\lambda}_n \to \lambda$ with probability 1.
2. The densities are not heavily concentrated, even for moderately large values of n. Therefore it is quite likely that we will predict values for the steady-state mean waiting time that are quite different from 1, simply because of our error in the estimate of the arrival rate. We need a *very* large value of n, i.e., a significant amount of data, to ensure high accuracy.
3. The densities are somewhat skewed, especially for small values of n, so that most of the probability concentrates around values significantly smaller than 1. So bias is most-likely significant, even for moderately large n.

So in this first example, we see that input model uncertainty can have a significant impact on performance predictions. One needs a very large number of observations to ensure high accuracy.

Of course, this example is contrived in several ways. First, the system we studied is a heavily-loaded M/M/1 queue. Performance measures for such queues are highly sensitive to input parameters. Hence, this example is perhaps an extreme example of sensitivity to input parameters. Second, the system has no bound on capacity. This is often a feature of *models* but not of real systems themselves. For example, call centers have a finite number of trunk lines, and emergency rooms in hospitals can redirect patients to other parts of the hospital, or to other hospitals. Nevertheless, for any capacitated queue, similar phenomena arise. Barton and Schruben (2001) explicitly deal with a capacitated queue in their example, and yet they observe similar behaviour to that shown above. Third, we assumed that interarrival times were indeed exponentially distributed. In general we may suspect that this is the case via our understanding of a process, but still not be absolutely sure. Zouaoui and Wilson (2001a) and others call such uncertainty *model* uncertainty as opposed to *parameter* uncertainty. Even when we ignore model uncertainty we see nontrivial behaviour. Finally, we assumed a "zero variance" simulation. In practice we do not have this luxury and must explicitly deal with the fact that simulation estimates of performance measures are subject to simulation uncertainty.

As noted above, Barton and Schruben (2001) consider a similar example in order to demonstrate the difficulties associated with ignoring input model uncertainty. They look at a single-server queue with capacity for 10 customers, where customers arriving to a full system are lost. Instead of using parametric distributions as we did, they instead use smoothed empirical distribution functions to estimate both the interarrival and service time distributions from data. They conduct a finite-length simulation run in order to estimate the expected steady-state sojourn time over customers that actually enter the queue. As the simulation runlength increases, the confidence intervals reduce in width. For an infinite simulation runlength the confidence interval widths would converge to 0, giving an exact result, just as we assumed in the above example. They observe that the coverage of the confidence intervals, given by the fraction of their confidence intervals that cover the true value of the performance measure, deteriorates as the simulation runlength increases. As they point out, this is to be expected because any one of their experiments first samples an interarrival and service time distribution. The sampled distributions differ from the true distributions, and so the simulation experiment estimates the performance associated with the wrong system.

Typically, simulation is used to provide insight so that a decision can be made. In Example 2 above the decision

is how many staff to hire to ensure satisfactory customer service. The main reason that we care about input model uncertainty is that it may lead us to an incorrect decision. If we underestimate the true arrival rate in Example 2, then we will likely not hire enough staff members to ensure satisfactory waiting times. If we overestimate the true arrival rate then we will provide better service than we expected (not such a bad thing), but at the financial cost of hiring too many staff. While these issues may not be so critical in a bakery or wine store, they are of more concern when one is dealing with an emergency service call center, for example.

3 FRAMEWORK

Let us consider a fairly general framework that will allow a concrete discussion of the issues. This framework is essentially that proposed in Cheng (1994) with a small extension to allow for model uncertainty, and another slight modification of notation to reduce reliance on the Greek alphabet!

We wish to compute $\alpha = f(m_0, \theta_0)$, where the function f depends on two variables. The first variable m indexes a (potentially uncountably infinite) class of models. The second variable $\theta \in \Re^p$ represents a finite-dimensional set of parameters. For a given model m, one may not need all p parameters since various models require more or less parameters. For example, the normal distribution has 2 parameters while the exponential has only 1.

The quantities m_0 and θ_0 represent the "true" model and its associated parameters. The notion of a "true" model and set of parameters is contrary to a Bayesian philosophy. When we come to Bayesian methods we will modify the discussion accordingly. The function f gives, for each m and θ, the exact value of the desired performance measure that would be obtained from a simulation with zero variance.

A simulation model is available that can be used to estimate $f(m, \theta)$. Specifically the simulation model can be used to generate i.i.d. samples $(X_i(m, \theta) : i \geq 1)$, where the samples are unbiased ($EX_i(m, \theta) = f(m, \theta)$) and have finite variance $\sigma^2(m, \theta)$.

This structure fits well with most terminating simulations where the goal is to compute an expected value. There are, however, simulation problems that do not fit this framework very well. An important class of problems that is excluded is steady-state simulation where initialization bias and choice of runlength play an important role. As another example, the problem of computing a quantile of the distribution of $X(m_0, \theta_0)$ does not fit our structure. For the problem of quantile estimation one can still view the performance measure as a function f of m and θ, but the function is no longer given by $f(m, \theta) = EX_1(m, \theta)$.

Suppose that one selects model $\hat{m}$ and associated parameters $\hat{\theta}$. These selections may be based on data or otherwise, and may be random or nonrandom. One then fixes $\hat{m}$ and $\hat{\theta}$ and then conditional on these values, obtains an i.i.d. sample of size ℓ, $(X_i(\hat{m}, \hat{\theta}) : 1 \leq i \leq \ell)$. One then estimates $f(m_0, \theta_0)$ by the sample average

$$\hat{\alpha} = \frac{1}{\ell} \sum_{i=1}^{\ell} X_i(\hat{m}, \hat{\theta}).$$

One way to measure the quality of $\hat{\alpha}$ is through its mean squared error

$$E(\hat{\alpha} - \alpha)^2 = \text{var}(\hat{\alpha}) + \text{bias}(\hat{\alpha})^2.$$

The bias is given by

$$Ef(\hat{m}, \hat{\theta}) - f(m_0, \theta_0). \tag{3}$$

The variance can be further broken down using the conditional variance formula as in Cheng (1994) to give

$$\begin{aligned} \text{var}(\hat{\alpha}) &= E\text{var}(\hat{\alpha}|\hat{m}, \hat{\theta}) + \text{var}E(\hat{\alpha}|\hat{m}, \hat{\theta}) \\ &= \frac{E\sigma^2(\hat{m}, \hat{\theta})}{\ell} + \text{var}\, f(\hat{m}, \hat{\theta}). \end{aligned} \tag{4}$$

We label the two terms in (4) expected simulation variance and input model variance respectively. Thus, the mean squared error of $\hat{\alpha}$ has three components: squared bias, expected simulation variance and input model variance.

In Example 3 there was only a single choice of model m_0. We assumed a zero-variance simulation so that $\sigma^2(m_0, \theta)$ is identically 0 for any θ. Input model variance is simply the variance associated with the density that was plotted. Bias is identified by comparing the mean of the density with the true value 1.

4 THE "STANDARD" APPROACH

We have seen that input model uncertainty can have a dramatic impact on performance predictions, and therefore on the decisions one makes based on the results of the simulation study. So what should one do about it? In this section we describe one of the currently used methods for dealing with input model uncertainty.

Bruce Schmeiser discusses input model uncertainty in Barton et al. (2002). He draws a clear distinction between model error (as can be quantified by bias as in (3)) and simulation error (quantified by simulation variance, i.e., the first term in (4)), but does not discuss input model variance. His main point is that so long as the simulation user understands that the error bounds reported by a simulation are with respect to simulation error alone, "all is well." To paraphrase his position, both bias and input model variance are irrelevant so long as one understands that simulation is

only a tool for analyzing the system with the selected input model and parameters.

Schmeiser also explicitly recognizes that it is important to obtain a sense of the effect of modeling error. He then argues that these two problems should be treated separately. This position is a reasonable one, and one that is essentially the status quo. There are some difficulties with this overall framework though, and we describe some of them below.

A tool that is often used to explore the impact of input model uncertainty is *sensitivity analysis*. A sensitivity analysis (e.g., Kleijnen 1994, Kleijnen 1996) is performed by varying the input distributions and parameters in some manner, and observing the changes in the output. This is often done in a somewhat haphazard way, although there are benefits to formalizing the approach using design of experiments and/or regression approaches. See, e.g., Chapter 12 of Law and Kelton 2000 for an accessible introduction to these techniques.

A standard, and often recommended, approach to quantifying the effect of parameter uncertainty is to use a 2^k factorial design. In this approach there are k (say) different parameters that are to be adjusted. The goal is to determine which parameters or parameter combinations have a significant effect on the output. If such parameters can be identified, then we can decide whether to collect more data to help improve the accuracy of estimates of these input parameters or not. Suppose we restrict attention to 2 possible values (high and low) for each parameter. Then there are 2^k possible parameter settings that could be considered. One then runs a simulation experiment at each of these parameter settings, and uses the results to determine which parameter combinations have a significant impact on the output performance measure.

Perhaps the greatest problem with this approach is that for large numbers of parameters k, a factorial design can require a tremendous amount of computation, i.e., the approach is not *efficient*. One might then use a fractional-factorial design, or screening methods to reduce the dimensionality of the problem (Kleijnen 1998). However, at this stage it starts to become necessary to have expert guidance on how to proceed, so that we run into difficulties with *implementability*. There are also other issues such as the selection of the high and low levels of each parameter.

These and several other issues are discussed in Kleijnen (1994), Kleijnen (1996), Kleijnen (1998). Kleijnen also briefly mentions uncertainty analysis, primarily in a setting like that of Example 3, where there is no simulation error, but also in the general case where simulation error is present. Uncertainty analysis involves randomly sampling the input parameters before each simulation run, but then holding the parameters constant during the run. Kleijnen (1994) also gives some references to early work that implements such uncertainty analysis, predominantly in the case where models are deterministic so that simulation error is not present.

5 RECENTLY PROPOSED METHODS

We now turn to some of the more-recently proposed methods for dealing with input model uncertainty in the presence of simulation error.

The closest method to that described in the previous section was described in Freimer and Schruben (2002). Freimer and Schruben give two design of experiments methods for deciding how much data to collect for one or more parameters. Both of their methods iteratively search for an amount of data so that the difference in the results of the simulation experiment are statistically indistinguishable at extreme values of the parameter settings. The extreme values they select are the endpoints of confidence intervals for the parameters. In other words, they search for an amount of input data that is sufficient to ensure that simulation variance dominates both bias and input model variance. Their approach is certainly implementable (it requires only a few easily-understood inputs from the simulation user and the rest of the procedure is automated). Unfortunately, both of their approaches may require a large amount of computation. Furthermore, they use repeated hypothesis tests which, while not unreasonable and certainly common in the literature, is a potential source of concern. Finally, the amount of data reported as required is related to simulation variance, and not to an error tolerance prescribed by the user.

5.1 Delta-Method Approaches

Starting with Cheng (1994) and continuing with Cheng and Holland (1997), Cheng and Holland (1998), Cheng and Holland (2003), Cheng and Holland have developed a framework and several methods for dealing with input model uncertainty. The framework given by Cheng (1994) has been adopted by several authors including Zouaoui and Wilson (2001b), Zouaoui and Wilson (2001a). (This framework is slightly extended in §3.) The framework assumes that the model m_0 is specified with certainty, but that the parameters θ are not. The parameters are often assumed to be approximately normally distributed as is the case, under mild regularity conditions, when maximum likelihood is used to estimate θ_0. Cheng (1994), Cheng and Holland (1997) use the delta method (see, e.g., Henderson 2000, Henderson 2001) to determine the first-order terms in the combined simulation variance and input model variance (4). The bias (3) is not considered in these early papers. They give estimators for the first-order variance terms. They find that the estimators suffer when there are a large number of uncertain parameters. To deal with this problem they also

give a parametric bootstrapping approach that is described below with other bootstrapping methods.

Cheng and Holland (1998) introduce two new methods for estimating the combined simulation and input model variance, again ignoring bias. The first of these methods involves two stages, where the vector

$$g = \nabla_\theta f(m, \theta)|_{\theta_0}$$

is estimated in the first stage, and then in the second stage the bulk of the simulation effort is run at only 2 parameter settings that depend on the estimated g and the covariance matrix of the estimate $\hat{\theta}$ of θ_0. The second method does not require the estimation of g, so that the first stage of simulation is unnecessary. The second method requires that the simulation user knows the *sign* of the entries in g but not necessarily their absolute values. The method results in a conservative confidence interval procedure in the sense that the variance is overestimated. One might often expect the signs to be known for scale parameters of distributions, but for other parameters such as shape parameters it seems unlikely that this information would be available. All of these early methods ignore the bias (3) which, as we have seen, can be substantial, and this is perhaps their most serious disadvantage.

The second new method given in Cheng and Holland (1998) is considerably extended in Cheng and Holland (2003). In the later paper, the assumption that the sign of the entries of g is retained, and a welcome improvement is that simulation bias is explicitly considered. The authors show that their procedure yields a conservative confidence interval with only a small amount of computation relative to their previously-developed methods. This method is attractive in that its computational requirements are small relative to virtually all other currently-known methods, and it has a sound underlying theory. It has the disadvantages that it only applies to parameter uncertainty, and it requires the simulation user to know the signs of the components of g, which makes the method less implementable than it might otherwise be. One can imagine an extension of this method where the signs of the components of g are estimated as part of the procedure. Typically it is much easier to estimate the *sign* of a quantity than it is to estimate the actual *value*, so that such a procedure might be expected to work quite well in practice. It is not yet known how the performance of this method compares with the Bayesian approaches described shortly.

Ng and Chick (2001) discuss the issue of how to reduce input parameter uncertainty for simulations. They employ an approximation that is essentially the delta-method approximation introduced in Cheng (1994), although the interpretation is different since a Bayesian framework is employed. They use this approximation to decide what data to collect next to maximally reduce the variance of the simulation output.

5.2 Bayesian Methods

Starting with Chick (1997) there has been a fair amount of recent interest in Bayesian methods for simulation input analysis (Chick 1997, Chick 1999, Chick 2000, Chick 2001, Ng and Chick 2001, Chick and Ng 2002, Zouaoui and Wilson 2001b, Zouaoui and Wilson 2001a). The idea of applying Bayesian techniques to simulation analysis is not new, however, and earlier references can be found in Chick (1997). The overall philosophy behind these methods is to place a prior distribution on the input models and parameters of a simulation, update the prior distribution to a posterior distribution based on available data, and only then run a simulation experiment. The posterior distribution quantifies uncertainty in the input model m and parameters θ.

Chick (2001) recommends implementing a Bayesian model average (Draper 1995). The Bayesian model average (BMA) is simply $Ef(\hat{m}, \hat{\theta})$, where $(\hat{m}, \hat{\theta})$ follows the posterior distribution. In order to compute this expectation, Chick (2001) generates i.i.d. replicates of $X(\hat{m}, \hat{\theta})$ by first sampling a single m and θ from the posterior and then, based on those values, generating a single $X(m, \theta)$. This process is then repeated and the results are averaged. Zouaoui and Wilson (2001b), Zouaoui and Wilson (2001a) introduce what they call a "BMA-based simulation replication algorithm," which is essentially a version of the Bayesian model average where the user exercises control over how many simulation replications are performed at each sampled model and set of parameters. More specifically, Zouaoui and Wilson (2001b), Zouaoui and Wilson (2001a) generate *several* conditionally i.i.d. values $X_1(m, \theta), \ldots, X_k(m, \theta)$ for each pair (m, θ) that are sampled, and use various methods to decide how many such values to generate at each pair (m, θ).

Zouaoui and Wilson (2001b) focuses on the special case where m is known with certainty. Zouaoui and Wilson (2003b) and Zouaoui and Wilson (2003a) are extensions of the conference papers that include proofs and additional computational examples.

Chick and Ng (2002) look at the problem of identifying which input parameters have the greatest impact on the mean of the simulation output while *simultaneously* trying to obtain accurate estimates of those parameters. They use an entropy-based performance measure that is essentially the sum of a "model discrimination term" and a "parameter estimation" term.

The Bayesian framework is an elegant one that enables a clean answer to many vexing questions. There are, however, several issues that deserve further attention. Perhaps the key issue is that of computational efficiency. It can be quite difficult to compute the posterior distribution in general,

so that one often has to resort to computational devices like Markov chain Monte Carlo methods or importance sampling. Chick (2001) provides an overview, and Zouaoui and Wilson (2001b), Zouaoui and Wilson (2001a) also discuss the issue. One of the key problems with either of these techniques is the need to tailor the methods to each application, which reduces the implementability of the Bayesian approach. Some users also object to the need to specify prior distributions for the data, although it is my personal view that this is actually a strength of the approach.

An important point is how one should interpret the output of the BMA algorithms given in these papers. The standard BMA estimates $Ef(\hat{m}, \hat{\theta})$, as does the method described in Zouaoui and Wilson (2001a), but this can be substantially different from $f(m_0, \theta_0)$ due to the bias (3). This objection makes sense, of course, only when one adopts the frequentist perspective that there is a single correct choice (m_0, θ_0) of the model and parameters. Perhaps a more robust interval-estimation method is the one described in Section 4.2.2 of Zouaoui and Wilson (2001b) based on quantiles of the posterior distribution of the simulation output, which is essentially a percentile confidence interval. This method bears a close resemblance to certain interval-estimation procedures used with resampling methods.

5.3 Resampling Methods

We do not review the key ideas of resampling here. For an introduction and overview of resampling methods in simulation, see Cheng (2000), Cheng (2001).

Barton and Schruben (1993) propose two resampling methods for accounting for input uncertainty. They use empirical distribution functions (EDFs) to model the distribution functions of independent input random variables. They perform a number of macro replications, where each macro replication consists of first sampling the input EDFs from a family of such distribution functions, and then performing a simulation experiment using the sampled empirical distribution functions. This approach is similar to a Bayesian model average, in that parameters (EDFs) are first sampled, and then a simulation run is conducted. Their two methods of resampling are standard bootstrap resampling and a method that they call uniform resampling.

Barton and Schruben (2001) provide an update on this approach. They also describe the construction of interval estimates of performance measures. They recommend the use of percentile confidence intervals for these two methods. They also introduce a method they call *direct resampling* where input data are partitioned into subsamples. Each subsample is then used to fit an EDF and a simulation experiment is performed. They conclude that with sufficient data one should use direct resampling together with t confidence intervals, i.e., confidence intervals of the form $A \pm tH$, where A is a point estimate, H is an estimate of the standard error of A, and t is a constant related to the desired confidence level. Given the bias that is evident in Figure 1 even for large values of n, the direct-resampling approach does not seem advisable, unless one is sure that the problem under study is not subject to such bias. Given the bias issue, the use of percentile confidence intervals together with one of the other resampling methods seems much more advisable.

One potential difficulty with the use of percentile confidence intervals is that they were originally recommended for use with bootstrapping methods in the absence of simulation uncertainty. Unfortunately, when simulation uncertainty is present, the percentile confidence intervals are based on a convolution of input model uncertainty *and* simulation uncertainty, rather than on input model uncertainty alone. This same problem is apparent in Section 3 of Cheng and Holland (2003), where a certain bootstrap method is reviewed, and in the interval estimation procedure mentioned at the end of Section 4.2.1 of Zouaoui and Wilson (2001b). There is not currently any obvious way to separate these two forms of uncertainty using existing resampling methods. It seems reasonable to expect that as long as simulation uncertainty is "small" relative to input model uncertainty, this convolution issue will not cause any major problems. However, more work is needed to understand exactly how such intervals behave.

In Barton et al. (2002), Barton reviews the resampling procedures advanced in Barton and Schruben (1993), Barton and Schruben (2001). In the same paper, Schruben discusses a variety of possible extensions.

Further discussion of resampling methods for input model uncertainty can be found in Cheng (1994) and Cheng and Holland (1997). The emphasis in these papers is on variance rather than bias.

5.4 Induced Distribution Methods

Recall that we assume that the estimates $\hat{m}$ and $\hat{\theta}$ of m_0 and θ_0 follow a certain distribution. This distribution can arise through expert solicitation as in Helton (1997), through frequentist techniques like maximum likelihood as discussed in Cheng (1994), Cheng and Holland (1997), through Bayesian formalisms as in Chick (2001), implicitly through a resampling scheme, or otherwise. Once this distribution is specified, we view $\hat{m}$ and $\hat{\theta}$ as random objects. The distribution of $f(\hat{m}, \hat{\theta})$ gives the distribution of the desired performance measure induced purely by input model uncertainty. Note that this distribution excludes any simulation uncertainty. This distribution is therefore a compact representation of the effect of input model uncertainty. This is, in fact, the distribution associated with the densities that were computed in Example 3, where we looked at a problem with no simulation uncertainty.

Andradóttir and Glynn (2003) describe how to estimate the mean $Ef(\hat{m},\hat{\theta})$, which is the same value as the BMA. Their framework is more general than the one here in that it is explicitly designed to incorporate features of steady-state simulations like bias, and they do not restrict attention to functions of the form $f(\hat{m},\hat{\theta}) = E[X_1(\hat{m},\hat{\theta})|\hat{m},\hat{\theta}]$ as we do. They perform several macro replications, where each macro replication first selects values for $\hat{m}$ and $\hat{\theta}$, and then devotes a varying amount of computational effort to a simulation at those input model settings. They show how to split effort between sampling values for $\hat{m}$ and $\hat{\theta}$ and simulating at those settings so as to minimize the mean squared error of an estimator of $Ef(\hat{m},\hat{\theta})$. They also show that if certain numerical integration schemes that are superior to Monte Carlo in low-dimensional problems are used, then one can improve the rate of convergence of the mean squared error to 0. Here, dimension refers to the combined dimensions of $\hat{m}$ and $\hat{\theta}$.

Recall that in the setup of Section 3, $f(\hat{m},\hat{\theta})$ is actually a conditional expectation, $f(\hat{m},\hat{\theta}) = E[X(\hat{m},\hat{\theta})|\hat{m},\hat{\theta}]$. Lee (1998) described how to efficiently compute the distribution function of $f(\hat{m},\hat{\theta})$ in the case where $(\hat{m},\hat{\theta})$ has a discrete distribution, and more generally. Lee and Glynn (1999) extended the results of Lee (1998) in the discrete-distribution case. The discrete distribution case is not of as much interest as the general case in our discussion since parameter uncertainty is often captured through continuous distributions.

Steckley and Henderson (2003) use kernel density estimation methods to estimate the density of $f(\hat{m},\hat{\theta})$ (when it exists). Under a variety of conditions they establish that the rate of convergence of the estimated density to the true density in terms of mean integrated squared error is of the order $c^{-4/7}$, where c is the computational budget. This rate is slower than that associated with kernel density estimation ($c^{-4/5}$) in the i.i.d. setting. The difference is due to the fact that one needs to control the simulation uncertainty as well as input model uncertainty.

Helton (1996) (see also Helton 1997) summarizes work done by a research group at Sandia National Laboratories on risk assessments for nuclear waste disposal. He estimates, for any given m and θ, the complementary cumulative distribution function (CCDF) of $X_1(m,\theta)$, $\bar{F}(x;m,\theta) = P(X_1(m,\theta) > x)$. When $\hat{m}$ and $\hat{\theta}$ are recognized as random variables, one then obtains a family of CCDFs. To manage the multidimensional nature of these results, he then focuses on a fixed value $x = R$ at which the CCDFs are evaluated, and looks at the distribution of these values induced by the input model uncertainty in $\hat{m}$ and $\hat{\theta}$. Latin hypercube sampling is used over the distribution of $(\hat{m},\hat{\theta})$ and Monte Carlo sampling is used for each fixed m and θ to obtain the results. The use of Latin hypercube sampling over the distribution of $(\hat{m},\hat{\theta})$ is similar in philosophy to the use of numerical integration techniques in Andradóttir and Glynn (2003). In general, numerical integration techniques should work well in low-dimensional problems, but can be expected to perform less well when the dimension of $(\hat{m},\hat{\theta})$ is high.

6 CONCLUSIONS

One can quite reasonably argue that there is no need to develop methods that capture input model uncertainty and simulation uncertainty in the same framework. So long as the simulation user is aware of potential model errors due to input model uncertainty, interprets the simulation output accordingly, and conducts sensitivity and/or uncertainty analyses all is well. The problem is that the typical simulation user is not particularly proficient in statistics, and so is unlikely to be aware of appropriate sensitivity and/or uncertainty analyses. This suggests the need for a transparent, statistically valid, implementable and efficient method for understanding input model uncertainty.

A feature of virtually all of the methods designed to capture input model uncertainty is additional computation over and above that required when a single model and set of parameters is chosen, i.e., the standard approach is followed. This computation almost invariably takes the form of repeated macro replications where $(\hat{m},\hat{\theta})$ are sampled, and then one or more simulation runs are performed at the sampled values. Of course, in the standard method, one usually needs to perform a careful sensitivity and/or uncertainty analysis. Once one factors in the additional computational effort required to perform such an analysis, it is no longer clear that the methods outlined above are computationally more demanding than the standard approach.

The benefit of these methods is a more appropriate representation of the uncertainty in predictions of performance measures than in the standard approach. The extent to which these methods are implementable, i.e., can avoid the need for expert intervention and thus be automated, is one of the chief factors that will determine whether they will be adopted in the mainstream.

Perhaps an even more important factor in determining whether these methods will be adopted is transparency. These methods need to be understood by users. Education of users about the issues and methods available is one key requirement. Another is ensuring that the results of these kinds of analyses can be put into a digestible form. Even confidence intervals are not as widely accepted as we might prefer, and if we are to report a confidence interval that estimates a deterministic quantity, the user must understand the interpretation of the deterministic quantity. This transparency requirement is all the more challenging when we realize that most simulation models are designed to estimate a large number of performance measures. Our methods should be able to easily handle such complexity.

At the current point in time there is no clear "winner" among the methods outlined in this paper. All have advantages and disadvantages relative to the other methods. It may very well be that many of the methods can be successfully applied to a single problem, and the choice of method may come down to a matter of taste. Nevertheless, many of the methods are still in an early stage of development, so any conclusions about the dominance of one method over another are probably somewhat premature.

ACKNOWLEDGMENTS

The support of the National Science Foundation through grants DMI-0224884 and DMI-0230528 is gratefully acknowledged.

REFERENCES

Andradóttir, S., and P. W. Glynn. 2003. Computing Bayesian means using simulation. Submitted for publication.

Barton, R. R., R. C. H. Cheng, S. E. Chick, S. G. Henderson, A. M. Law, L. M. Leemis, B. W. Schmeiser, L. W. Schruben, and J. R. Wilson. 2002. Panel on current issues in simulation input modeling. In *Proceedings of the 2002 Winter Simulation Conference*, ed. E. Yücesan, C.-H. Chen, J. L. Snowdon, and J. M. Charnes, 353–369. Piscataway, NJ: IEEE.

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, E. C. Russell, and W. E. Biles, 503–508. Piscataway, NJ: IEEE.

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, NJ: IEEE.

Ben-Tal, A., and A. Nemirovski. 1998. Robust convex optimization. *Mathematics of Operations Research* 23:769–805.

Ben-Tal, A., and A. Nemirovski. 2000. Robust solutions of linear programming problems contaminated with uncertain data. *Mathematical Programming, Series A* 88:411–424.

Cheng, R. C. H. 1994. Selecting input models. In *Proceedings of the 1994 Winter Simulation Conference*, ed. J. D. Tew, S. Manivannan, D. A. Sadowski, and A. F. Seila, 184–191. Piscataway, NJ: IEEE.

Cheng, R. C. H. 2000. Analysis of simulation output by resampling. *International Journal of Simulation: Systems, Science & Technology* 1:51–58.

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, NJ: IEEE.

Cheng, R. C. H., and W. Holland. 1997. Sensitivity of computer simulation experiments to errors in input data. *Journal of Statistical Computation and Simulation* 57:219–241.

Cheng, R. C. H., and W. Holland. 1998. Two-point methods for assessing variability in simulation output. *Journal of Statistical Computation and Simulation* 60:183–205.

Cheng, R. C. H., and W. Holland. 2003. Calculation of confidence intervals for simulation output. Submitted for publication.

Chick, S. E. 1997. Bayesian analysis for simulation input and output. In *Proceedings of the 1997 Winter Simulation Conference*, ed. S. Andradóttir, K. J. Healy, D. H. Withers, and B. L. Nelson, 253–260. Piscataway, NJ: IEEE.

Chick, S. E. 1999. Steps to implement Bayesian input distribution selection. In *Proceedings of the 1999 Winter Simulation Conference*, ed. P. A. Farrington, H. Black Nembhard, D. T. Sturrock, and G. W. Evans, 317–324. Piscataway, NJ: IEEE.

Chick, S. E. 2000. Bayesian methods for simulation. In *Proceedings of the 2000 Winter Simulation Conference*, ed. J. A. Joines, R. R. Barton, K. Kang, and P. A. Fishwick, 109–118. Piscataway, NJ: IEEE.

Chick, S. E. 2001. Input distribution selection for simulation experiments: accounting for input uncertainty. *Operations Research* 49:744–758.

Chick, S. E., and S. H. Ng. 2002. Joint criterion for factor identification and parameter estimation. In *Proceedings of the 2002 Winter Simulation Conference*, ed. E. Yücesan, C.-H. Chen, J. L. Snowdon, and J. M. Charnes, 400–406. Piscataway, NJ: IEEE.

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

Freimer, M., and L. Schruben. 2002. Collecting data and estimating parameters for input distributions. In *Proceedings of the 2002 Winter Simulation Conference*, ed. E. Yücesan, C.-H. Chen, J. L. Snowdon, and J. M. Charnes, 392–399. Piscataway, NJ: IEEE.

Helton, J. C. 1996. Computational structure of a performance assessment involving stochastic and subjective uncertainty. In *Proceedings of the 1996 Winter Simulation Conference*, ed. J. M. Charnes, D. J. Morrice, D. T. Brunner, and J. J. Swain, 239–247. Piscataway, NJ: IEEE.

Helton, J. C. 1997. Uncertainty and sensitivity analysis in the presence of stochastic and subjective uncertainty. *Journal of Statistical Computation and Simulation* 57:3–76.

Helton, J. C., and F. J. Davis. 2003. Latin hypercube sampling and the propagation of uncertainty in analyses

of complex systems. *Reliability Engineering & System Safety* 81:23–69.

Henderson, S. G. 2000. Mathematics for simulation. In *Proceedings of the 2000 Winter Simulation Conference*, ed. J. A. Joines, R. R. Barton, K. Kang, and P. A. Fishwick, 137–146. Piscataway NJ: IEEE.

Henderson, S. G. 2001. Mathematics for simulation. In *Proceedings of the 2001 Winter Simulation Conference*, ed. B. A. Peters, J. S. Smith, D. J. Medeiros, and M. W. Rohrer, 83–94. Piscataway NJ: IEEE.

Kleijnen, J. P. C. 1994. Sensitivity analysis versus uncertainty analysis: when to use what? In *Predictability and Nonlinear Modelling in Natural Sciences and Economics*, ed. J. Grasman and G. van Straten. Dordrecht: Kluwer Academic.

Kleijnen, J. P. C. 1996. Five-stage procedure for the evaluation of simulation models through statistical techniques. In *Proceedings of the 1996 Winter Simulation Conference*, ed. J. M. Charnes, D. J. Morrice, D. T. Brunner, and J. J. Swain, 248–254. Piscataway, NJ: IEEE.

Kleijnen, J. P. C. 1998. Experimental design for sensitivity analysis, optimization, and validation of simulation models. In *Handbook of Simulation: Principles, Methodology, Advances, Applications, and Practice*, ed. J. Banks. New York: Wiley.

Law, A. M., and W. D. Kelton. 2000. *Simulation Modeling and Analysis*. 3rd ed. New York: McGraw-Hill.

Lee, S. H. 1998. *Monte Carlo Computation of Conditional Expectation Quantiles*. Ph.D. thesis, Stanford University, Stanford, CA.

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. Black Nembhard, D. T. Sturrock, and G. W. Evans, 1654–1663. Piscataway, NJ: IEEE.

Ng, S. H., and S. E. Chick. 2001. Reducing input parameter uncertainty for simulations. In *Proceedings of the 2001 Winter Simulation Conference*, ed. B. A. Peters, J. S. Smith, D. J. Medeiros, and M. W. Rohrer, 364–371. Piscataway, NJ: IEEE.

Oberkampf, W. L., J. C. Helton, C. A. Joslyn, S. F. Wojtkiewicz, and S. Ferson. 2003. Challenge problems: uncertainty in system response given uncertain parameters. Available online via <http://www.sandia.gov/epistemic/eup_challenge.htm> [accessed July 6, 2003].

Steckley, S. G., and S. G. Henderson. 2003. A kernel approach to estimating the density of a conditional expectation. In *Proceedings of the 2003 Winter Simulation Conference*, ed. S. E. Chick, P. J. Sánchez, D. J. Morrice, and D. Ferrin, To appear. Piscataway, NJ: IEEE.

Zouaoui, F., and J. R. Wilson. 2001a. 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, NJ: IEEE.

Zouaoui, F., and J. R. Wilson. 2001b. Accounting for parameter uncertainty in simulation input modeling. In *Proceedings of the 2001 Winter Simulation Conference*, ed. B. A. Peters, J. S. Smith, D. J. Medeiros, and M. W. Rohrer, 354–363. Piscataway, NJ: IEEE.

Zouaoui, F., and J. R. Wilson. 2003a. Accounting for input model and parameter uncertainty in simulation. Submitted for publication.

Zouaoui, F., and J. R. Wilson. 2003b. Accounting for parameter uncertainty in simulation input modeling. *IIE Transactions*. To appear.

AUTHOR BIOGRAPHY

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. He likes cats but is allergic to them. His research interests include discrete-event simulation, queueing theory and scheduling problems. His e-mail address is <sgh9@cornell.edu>, and his web page URL is <www.orie.cornell.edu/~shane>.