Proceedings of the 2001 Winter Simulation Conference
B. A. Peters, J. S. Smith, D. J. Medeiros, and M. W. Rohrer, eds.

ABC'S OF OUTPUT ANALYSIS

Susan M. Sanchez

Operations Research Department and
Graduate School of Business & Public Policy
Naval Postgraduate School
Monterey, CA 93943, U.S.A.

ABSTRACT

We present a brief overview of several of the basic output analysis techniques for evaluating stochastic dynamic simulations. This tutorial is intended for those with little previous exposure to the topic, for those in need of a refresher course, and especially for those who have never heard of output analysis. We discuss the reasons why simulation output analysis differs from that taught in basic statistics courses and point out how to avoid common pitfalls that may lead to erroneous results and faulty conclusions.

1 INTRODUCTION

The process of building, validating, verifying and using a simulation model for decision-making can be arduous. You've spent a great deal of time and effort in several distinctly different tasks: working with the decision-makers who will be the end users of the simulation results, determining what data to collect to create reasonable distributions for various model components, coding and verifying the simulation model, and then validating its behavior. After you've taken such care in these earlier stages of the simulation process, you owe it to yourself to analyze the output properly—if not, you've negated much of your effort. Fortunately, the output analysis stage is generally much less time-consuming than the earlier modeling and coding stages because the simulation model is now working for you. Matters are facilitated as simulation software companies continue to improve the output analysis capabilities of their packages. It can also be fascinating to discover the patterns and complexities of the simulation model's behavior under one (or more) scenarios. Output analysis will allow you and the end-user to effectively gain insights into the model's performance, and so lead to better decisions.

Before going further, there are two types of simulations that we will not be discussing in any detail. The first is the class of *deterministic simulation models*, in which no stochastic elements are involved. Deterministic simulations use fixed, non-random values to specify the model and particular variant of the system under investigation. Because there is no randomness, the output is also fixed for any specific set of inputs. Rerunning the simulation with the same input factors will give the same result, so output analysis is concerned with uncovering the fixed input/output relationship. The second is the class of *static simulation models*, where the analyst essentially uses random sampling over input distributions to perform numerical integration of a static system. Both of these modeling approaches are certainly legitimate uses of simulation, but fall outside the scope of this tutorial.

The world is full of uncertainty, and most (if not all) realistic simulation models will incorporate some randomness as well as some element of time elapsing. We therefore focus on *stochastic, dynamic simulation models* throughout the rest of this paper. Such models can be used to examine a diverse set of applications. For example, the simulation may have been designed to model the operation of a customer service center, traffic patterns over a particular location grid, hospital facilities utilization, waiting times for customers arriving at a service center, the number of cars passing through an intersection during a 5 minute period, the efficacy of various strategies in combat warfare, the impact of changes in layout and equipment on production throughput, and more.

Within the class of stochastic simulation models, one further distinction is necessary: simulations can be either *terminating* (sometimes called *finite*) or *nonterminating* in nature. Terminating simulations are those in which there is a natural event which specifies when the simulation is complete. Examples include events such as the time at which a satellite experiences catastrophic failure, the time at which a retail establishment finishes for the day, the completion of a construction project or the end of a fixed-term contract for supplying a good or service. Many times these termination events are stochastic, rather than deterministic. While a bank's door may state that the closing time is 6:00 p.m. on weekdays, if any customers arrive just before 6:00 their service will extend slightly beyond the official close of the day. Nonterminating simulations are those for which no natural terminating event exists. These could include the operation of a manufacturing facility if work in process remains on the shop floor. Even if the factory closes during the evening, we can treat its hours of operation as a long nonterminating simulation. Some nonterminating simulated systems exhibit *steady-state* behavior, which means that in the long run, the distribution of the output measure is independent of time. If the output has a fixed mean value and covariance structure, we say that it is *weakly stationary*.

Suppose that a stochastic, dynamic simulation model has been successfully developed and validated. Running this simulation model will generate a stream of output. This output might be indexed by time, e.g., the model's output might be the number of patients checked into the hospital at midnight on successive days. Alternatively, the output might be indexed by count, such as the service time for successive customers who depart from a system. In either case, you must decide how to generate and analyze the output.

We address this via the ABC's of output analysis. In Sections 2, 3 and 4 we describe some basic concepts that are important for simulators to understand in order to conduct output analysis properly. In Section 5 we briefly mention extensions related to these basics, as well as some other more advanced or more specialized output analysis techniques. Our goal is not to present full details of the methods, but to leave the reader with an appreciation for the topics. More complete discussions and additional references can be found in simulation texts, such as Banks et al. (2000), Bratley, Fox and Schrage (1987), Fishman (1978), Law and Kelton (2000), Nelson (1995), Thesen and Travis (1992), as well as in Alexopoulos and Seila (1998), Kelton (1997) or other papers cited in this tutorial.

2 THE A'S: PREPARING FOR ANALYSIS

2.1 Application-Appropriate Output Measures

One pitfall that may arise in analyzing simulation output is a lack of a clear understanding of what question is being asked. Presumably, the end-users' needs interests were considered as the simulation model was initially developed. However, it is not uncommon for a model to be developed and built for one purpose and then subsequently expanded or used to address another question of interest. While it may sound simple, make sure *before you go any farther* that you're using output measures which are appropriate for answering the questions at hand!

Appropriateness means more than one thing. First, do you have the 'right' output measure? Consider a bank manager, who has commissioned a simulation study because she is concerned about customer waiting times when new services are offered. Possible quantities of interest include:

- expected customer time in system,
- expected customer waiting time (prior to service),
- probability that waiting time exceeds 10 minutes,
- variance of customer waiting time,
- variance of customer service time,
- probability of 1 to 3 minute service times or
- probability of 15 or more customers in line,

to name a few. Each of these measures is appropriate for answering a question, but the questions differ. For example, if the manager is really interested in the number of 'unhappy' customers who must wait longer than a specified amount of time to receive service, then good information on the expected service time — or even the expected waiting time — will not provide her with the information she needs. If you check to make sure that the decision-maker understands the implications of using several possible performance measures, before arriving at an agreement of which one(s) to use, you may avoid a great deal of hassle in the future.

Thus, the first (and most important) issue in selecting an output measure is insuring that it answers the right question. If there are noticeable constraints on computing time or budget, you may also wish to consider whether or not you're collecting the output measure directly. For example, the customer time in the system is equal to the sum of the time spent awaiting service and the time spent receiving service. If your interest is primarily in waiting time, it will be more efficient to report and evaluate waiting time directly than to estimate the expected total time, the expected service time, and use these to estimate the expected waiting time. This is often less of an issue now than in the past, since successive generations of CPUs keep reducing computing time requirements geometrically. However, since decision-makers have responded to the increased computing power by demanding insights for increasingly complex systems, and since answers to important questions are often needed 'yesterday,' the problem is not likely to go away completely. We simulators should be thankful: as we're better able to support effective decision-making, our jobs may become more interesting and more secure!

2.2 Autocorrelation Awareness

One qualitative difference between generating output via simulation and collecting data in traditional statistical sampling applications (e.g., surveys, agricultural experiments) is the high degree of serial autocorrelation that is typically seen in simulation output streams. Queueing systems — a common class of problems modeled via discrete-event simulation — are notorious for exhibiting this type of behavior. For example, consider a fast food restaurant with a single drive-thru window. If one car must wait a long time before receiving their order because they joined the end of a long queue, then it is likely that cars arriving just before or just after this car will also experience longer waits. Conversely, if a car arrives and the driver immediately places an order, then it is likely that the next car arriving will experience at most a short delay. While this relationship is not deterministic, it will reveal itself as a series of positively autocorrelated data: cars arriving in close proximity to one another are more likely to exhibit similar waiting times than those arriving far apart. (Negatively correlated output streams sometimes occur, but far less frequently than positively correlated output.)

The net impact of correlation in simulation output is that you need to generate *a whole lot* of information in order to get a reasonable picture of the system behavior. You cannot treat successive output values as independent observations—if you do, particularly for short output streams, you're likely to vastly underestimate the system variance and, perhaps, provide a biased estimate of the system mean. This can lead to unpleasant surprises when the system is implemented.

2.3 Averages and Aggregation

You wouldn't feel comfortable predicting the outcome of an election after polling one prospective voter, so you shouldn't feel comfortable reporting one number from a simulation as "the answer." This is true even if that number is itself a summary obtained from a large sample, such as the average waiting time of the bank's first 100 (or even 1,000) customers, or the total number of customers arriving during the day. As we show in Section 4, the right way to summarize simulation output involves appropriately conveying information about both the center and the spread of the output measure's distribution. This typically means constructing interval estimates, rather than simply point estimates, of the underlying 'true' performance.

Despite the fact that a single averaged or aggregated value will not suffice for purposes of simulation output analysis, averages and aggregates still play important roles as steps along the way. So, while the waiting times of successive customers may be highly correlated, the average waiting times from one day to the next should be independent. If the aggregation or averaging involves a large initial sample, then it is more likely that the distribution of the resulting summary measure will be normally distributed. If you examine the right output measure, and deal with data summaries that look independent and perhaps even normally distributed, then (as you'll shortly see), the rest of the analysis won't be difficult.

3 THE B'S: BREAD-AND-BUTTER TECHNIQUES

3.1 Bias Removal

Often, queueing system simulations begin from a state which is easy to visualize and convenient to program. For example, consider the so-called 'empty and idle state' for a hospital: there are no patients, no outstanding laboratory or diagnostic tests to be conducted, no broken equipment, but a full complement of hospital staff stand ready to perform their duties. As we start running the simulation, we generate entities and activities: hospital staff schedules, patient arrivals, patient medical care needs, equipment and supplies arrive or are utilized, and so forth. These in turn interact within the simulation, creating bottlenecks, scheduling conflicts, routing and capacity problems, and a host of other changes in the system state. Eventually the impacts of the unrealistic initial conditions wash out. We say that the system has 'warmed up' and the hospital operates under its steady-state distribution.

Initialization bias refers to the fact that if most (or all) of the output stream is generated during the warm-up period, then averages or other summary measures of these data may dramatically overestimate or underestimate the steady-state performance. One way to counteract initialization bias is to start the system under steady-state conditions. Unfortunately, you may not know what these conditions are until after you've run the simulation and done some output analysis, so convenience may dictate that a simple (albeit unrealistic) starting state be used. Initialization bias problems can still be avoided any data obtained during the warm-up period is deleted prior to further analysis. Determining the length of the warm-up period is not a science, but several graphical and numerical methods have been proposed and tested.

The main idea: you only want 'good data' that accurately represents the performance of the system. This means — once again — you must be sure that your analysis matches the question of interest. If you are studying the operation of a bank, with working hours 9:00 a.m. to 6:00 p.m., then you have a terminating simulation for which *all* of the data are useful. If you want to know the average number of customers served during a day, it would be wrong to throw out data at the beginning of the day because the bank started out empty. On the other hand, if our interest is in steady-state utilizations within the hospital, then you should discard the initial transient or warm-up period because empty-and-idle conditions are completely unrealistic assumptions.

3.2 Basic Replications

Perhaps the simplest output analysis technique to explain is one in which the simulation is treated (almost) as any other basic experimental unit for statistical sampling purposes. If you have independent observations of some output measure, then you will be able to use standard statistical methods to generate confidence intervals for its expected value. Consider first a terminating simulation, such as a single day of operations at a bank, where initialization bias is not an issue. The basic replication method consists of getting independent output streams by making several runs with different random number seeds. Output from a single run can then be averaged or aggregated to yield a *single* output value, such as the mean waiting time or the total number of customers served during that run. Note that if the output of interest is the time until termination, or the number of events (such as sales) before termination, then the run's output is already in the form of a single number.

For nonterminating simulations this technique is often called the replication/deletion method, because each replication's warm-up period must be deleted before the summary output value for that replication is computed. In practice, it is easier to implement the replication/deletion method if the same truncation point is used for all replications. It is also easier to explain if round numbers are used: managers may readily accept a statement such as 'from each run, we eliminated the first $1,000$ observations (or 100 simulated hours of output)' if you explain the initialization bias problem. However, they may become suspicious and believe you're manipulating the results if you make a statement like 'we eliminated the first 933 observations (or 102.81 hours of output).'

3.3 Batch Means

For nonterminating simulations, another common approach used to achieve near-independence between summary output values is the method of batch means. This essentially takes the output stream and chops it up into batches of equal size. Then a single summary output measure—often the mean—is computed for each batch. If the batch size is sufficiently large, then the batch means will be approximately independent of one another.

There are several methods that one can use to determine a batch size, though for a moderately busy queueing system it's not unreasonable to have around 1,000 departures per batch. If you've already calculated the length of the warm-up period, then this may give you a conservative estimate of the necessary batch size. In practice, many analysts choose a large batch size, perhaps a convenient round number, and then delete the first batch or batches from consideration to alleviate initialization bias. The pre-specified batch size is used unless it appears (from graphical or statistical analysis) to be problematically small.

For nonterminating simulations, the savings in total run length can be substantial if you use batch means instead of using the basic replication/deletion method. This is particularly true if the warm-up period w is long. For example, suppose $w = 3,000$ and you want 20 (approximately) independent groups of data made up of $1,000$ observations each. If you used basic replication/deletion, you'd need to generate a total of $n = 20(3,000 + 1,000) = 80,000$ observations, and you'd end up throwing 75% of these away. In contrast, if you used batch means you'd need to generate only $n = 3,000 + 20(1,000) = 23,000$ observations and you'd only discard 13% of the data.

4 THE C'S: CONVEYING THE RESULTS

4.1 Confidence

As mentioned earlier, point estimates are not useful for decision-making purposes. Suppose that after any necessary truncation, you have n summary values. Let's call these $\overline{Y}_1, \overline{Y}_2, \ldots, \overline{Y}_n$, although you should remember that these might be percentiles, or variances, or summary statistics other than sample averages. (These arise from n runs under the replication/deletion method, or n batches under the method of batch means.) Let S denote the standard deviation of these $\overline{Y}_i$, and let $t_{1-\alpha/2;n-1}$ denote the value from the t distribution corresponding to an upper-tail area of $\alpha/2$. Then a $100(1-\alpha)\%$ confidence interval for the true expected performance is

$$\frac{1}{n}\sum_{i=1}^{n}\overline{Y} \pm t_{1-\alpha/2;n-1}\frac{S}{\sqrt{n}}.$$

For this interval to be valid, the $\overline{Y}_i$'s should be essentially independent, and either normally distributed (perhaps because they are averages or aggregates of a large number of raw output values), or else n should be sufficiently large that the central limit theorem applies. Remember that the total data collection effort may be huge, even if the degrees of freedom are small. For example, if we have taken 5 batches of $15,000$ observations each, then we have only four degrees of freedom — not $14,999$ or $74,999$.

For a fixed total computational effort, there is a trade-off between the number of runs (or batches) and the run length (or batch size) required, even if initialization bias is not an issue. For illustration purposes, suppose we're dealing with batch means. If the batch size is large, then S, the standard deviation of the batch means, will be low because the $\overline{Y}_i$'s will be tend to be quite close to their expected value (and, serendipidously, more likely to be normally distributed). However, the small number of batches means that the denominator $\sqrt{n}$ will be small and the t-value will be larger, together acting to increase the width of the confidence interval. On the other hand, if many short runs are made, then the t-value shrinks to the normal distribution value and $\sqrt{n}$ is large, but at the same time S may be extremely high if little averaging occurs within the batch. The same trade-off holds conceptually if you use the basic replication or replication/deletion approach. Both long runs and many runs are desirable, but if you've got constraints on time or budget you can't achieve both.

As an alternative to formal statistical inference, some clever graphical displays can be used to describe simulation output. A well-constructed picture may easily be worth a thousand words if it reveals clear patterns that might go undetected if only standard numerical summaries were used. Several graphical techniques for describing simulation output are described in more detail by Grier (1992).

Animation has become increasingly popular, and many simulation software companies now have built-in animation capabilities in their packages. Animation can be useful for debugging code and identifying incomplete model specifications (such as forklift trucks running through each other in production facilities). It is also used for some other purposes, notably that of improving the buy-in of decision-makers on the model's logic, construction, and ultimate utility. However, it isn't worthwhile to get the decision-maker to 'believe in' your simulation model if you don't bother to use this model to obtain comprehensive results. A short time spent watching a visual animation of part of the system is no substitute for a valid statistical analysis: because of the autocorrelation and initialization issues—or random chance—you may be observing the system in highly unusual states. Human judgement is easily swayed by occurrences which may be visually striking, but have minimal real impact.

The usefulness of the confidence interval for decision-making purposes will, as in basic statistics, depend on its width and the level of confidence $100(1-\alpha)$. Even if you're using graphical displays as the primary method for conveying the results, rather than formal statistical inference, you should be fairly certain that you've captured the essential characteristics of the output. How can you achieve this confidence? As we describe in the next section, you can take explicit control of the simulation run conditions.

4.2 Control

What if you construct your confidence interval and find that it is narrower than some desired precision? What if your histograms or dot plots look essentially the same if you base them on only half of the output data? No problem — you may have wasted some computer CPU cycles, but your results should be useful to the decision-maker. However, if you spent a great deal of unneeded time collecting simulation output data, then you might want to look more carefully at control issues before beginning your next analysis.

On the other hand, perhaps your confidence intervals are too wide or your graphical displays are difficult to interpret. Then the decision-maker may not have the information they need to arrive at a good decision. For example, suppose the marketing department has shown that a new policy of "on time or half price" will be profitable only if fewer than 1% of production orders are not filled by their due date. A simulation model of the manufacturing facility, including forecasts of (random) customer demand, is created. If a confidence interval for the expected proportion of late orders is $[0.003, 0.004]$, then the simulation results show that the new policy is profitable. If the confidence interval for the expected proportion of late orders is $[0.013, 0.018]$, then the new policy appears unprofitable. But if the confidence interval is $[0.003, 0.018]$ then the decision-maker does not have sufficient evidence to make a judgment on the profitability issues. This interval is too wide to address the problem at hand. You can 'fix' this problem by collecting more output data and redoing the analysis.

Remember that not all problems in interpreting simulation output relate to the statistical analysis. If a very narrow interval covered this breakpoint, then the problem may be best answered by revisiting the model specifications. The decision-maker might wish to check the model assumptions for correctness, check the so-called break-even point for accuracy, or run the simulation using other potential demand patterns to develop best case, worst case, and baseline scenarios.

The confidence interval width is essentially under your control, since (formally or informally) you decide how many runs to make. If you're studying a nonterminating simulation, you also control the total sample size, with the caveat that the runs (or batches) should be long enough for you to deal with any initialization effects. For the method of batch means, you'll need to set the batch size and number of batches before making the final run. From a practical perspective, unless you're willing to make a really long run and hope that it yields a suitable number of batches, you may want to conduct a pilot run in order to ballpark a desirable batch size. It is easy to add additional runs under the the basic replication or replication/deletion method, although a pilot run is still beneficial to assess whether or not initialization bias is a problem. Whether the unit of analysis is a run or a batch, the most important rule is: THE NUMBER 1 IS TOO LOW! You are exposing yourself and your client to great danger if you rely on a *single summary value* from simulation output, even if you let the computer run a long time to get this value.

While sample size is controllable in statistical sampling in general, as a simulation analyst you have more control over experimental conditions than, say, someone performing experiments on a physical system. You can specify the random number seeds used to generate the output for each of the simulation runs. You can control the simulation model's initial conditions. You can control the levels for various parameters embedded in the simulation model to assess its performance under different conditions.

With this additional level of control comes the opportunity to evaluate the output more efficiently or in greater depth. These simulation-specific controllable factors can be used to plan your data collection effort. For example, consider the bank simulation where each run generates output for a single day of operation. Distinct random number seeds will mean the output data are independent from run to run. Alternatively, if your random variables are generated by inversion, you could pair runs by generating a random number stream for the one run, and using the *antithetic* stream for the second run. The antithetic stream essentially generates a low value whenever the original random number stream generates a high value, and vice versa. Under such a sampling scheme, you are insuring that you investigate the system under a variety of different scenarios.

As we discuss in the next section, exercising your control over the simulation may be particularly beneficial when you are making comparisons.

4.3 Comparisons

At times, the purpose of preparing a simulation model is not to assess the capability of a single system, but to compare one or more systems to a standard level of performance, to compare several systems to one another, or to determine how the performance of one system changes according to particular variants of operating conditions. Appropriate output analysis tools have been developed for all these cases, although many of these questions are difficult and there is still room for further work.

Hypothesis tests, confidence intervals, or multiple comparison procedures can be used when comparing systems to a pre-determined standard. When comparing several systems to one another, *selection and ranking procedures* can be used to specify 'good' or 'best' systems, while allowing the analyst to make an intuitively appealing probability guarantee about the selection process. For example, you might focus on choosing the system with the highest mean: a selection method could guarantee that the best system will be chosen with high probability provided the difference between the true best and second-best exceeds some pre-specified "smallest practical difference." Subset selection procedures are good screening methods if you're investigating a large number of systems and wish to identify those which merit further investigation. *Multiple comparison procedures* augment the selection and ranking approaches by providing estimates of the true performance measures in addition to determining the selected system or group of systems. For more on selection and multiple comparison procedures, see chapter 10 of Law and Kelton (2000), Goldsman and Nelson (1998), Matejcik and Nelson (1995) or Nakayama (1997).

The selection and ranking approach is useful for comparing distinct systems or systems characterized by distinct protocols, such operating performance of queueing networks under FIFO or LIFO priority queues, or different layouts of a manufacturing facility. If, however, different system configurations result from changing levels of some *quantitative* variables, then response surface methodology is another alternative. Response surface metamodels seek to approximate the simulation input/output relationship analytically, as in a polynomial regression model for the relationship between parameter settings (over limited ranges) and the mean performance of the simulation. Regression-based response surface metamodels in the simulation arena are discussed in Hood and Welch (1993), Kleijnen (1987, 1998) and chapter 12 of Law and Kelton (2000). Frequency domain approaches have been examined by Schruben and Cogliano (1987) and Morrice and Schruben (1993). Barton (1998) has detailed references regarding a broad range of response surface metamodels, which include structures that may be more suitable than polynomial regression models for the highly non-linear structures that may arise in complex stochastic simulations.

When used in conjunction with robust design approach, response surface metamodels can identify systems which are relatively insensitive to uncontrollable uncertainties (such as customer demand rates) or deviations of system decision factor levels from planned values. For details and related references, see Sanchez (2000), or Sanchez et al. (1996, 1998). Saltelli (1999) explores the dynamics of changing sources of variation for complex systems.

5 BEYOND THE BASICS

We have just touched on some of the aspects of output analysis for stochastic simulation models. A rich body of literature exists on extensions or alternatives to the topics described earlier. We present a very brief summary of some of these topics, along with references for the reader interested in further details.

Another output analysis technique which has received attention in the literature is the regenerative method. This approach seeks to gain independence by bunching the data in a different way: the output stream begins a new *regenerative cycle* whenever it returns to a particular state. For example, an M/M/1 queue regenerates each time the system is empty and idle with an operational server. Regenerative cycles are often easy to detect and conceptually pleasant, but the analysis is not without difficulties. Planning the runs is harder, since the time between cycles is random and generally not known *a priori*. The choice of a regenerative state is not straightforward: easy ones to describe, such as empty-and-idle, may occur only rarely, and the estimates of mean performance are only asymptotically unbiased. This means even with long runs you have no guarantee that the desired estimation precision will be attainable.

The regenerative method and the output analysis approaches of Section 3 seek to aggregate data in such a way as to treat summaries of portions of the total output as independent for purposes of analysis. There are other output analysis techniques that take different approaches. In *time-series analysis*, the correlated, nonstationary simulation output series is treated just like a time series of economic data, such as stock prices or new business starts over time. Then a time-series model (such as an ARMA model) is fit to the data, and the fitted model is used for inference. The *spectral analysis method* directly estimates the correlation structure of the process, and uses this in turn to form a variance estimate for statistical analysis. In the *standardized time series* approach, a process version of the central limit theorem is applied to "standardize" the output series, and appropriate methods for statistically analyzing this standardized series have been worked out. More on these topics and the methods of Section 3 can be found in chapter 11 of Banks et al. (2000), chapter 3 of Bratley, Fox and Schrage (1987), chapters 2, 3 and 5 of Fishman (1978), chapter 7 of Khoshnevis (1994), Kleijnen (1987), chapter 9 of Law and Kelton (2000), Lewis and Orav (1989), chapter 6 of Ripley (1987), and chapter 6 of Thesen and Travis (1992). More recently, Bayesian approaches to simulation output analysis have been proposed. See Chick (2000) or Cheng (1998) for examples and further references.

Appropriate planning is much more efficient than trial-and-error for assessing the system performance under different scenarios. This means that you may benefit from the use of variance reduction or experimental design techniques, particularly in cases where it is expensive or time-consuming to generate the simulation output. Resulting gains in efficiency will allow you to either construct narrower confidence intervals for output measures for the same amount of data, or to complete the simulation runs more quickly for a particular desired level of confidence. Many variance reduction (or variance reallocation) techniques have been proposed to increase the efficiency of estimating mean performance. The simplest of these is to use *common random number streams* when comparing two or more systems. To those familiar with experimental design terminology, this is a form of blocking in order to better estimate the difference in performance attributable to the alternative systems, rather than that due to stochastic (random) error. A host of creative methods for variance reduction/reallocation have appeared in the literature; see chapter 2 of Bratley, Fox and Schrage (1987), chapter 3 of Fishman (1978), Kleijnen (1987), chapter 11 of Law and Kelton (2000), L'Ecuyer (1994), Lewis and Orav (1989), Nelson (1992) or chapter 5 of Ripley (1987).

Experimental design can be particularly beneficial when the overall purpose is to perform comparisons of many systems to system configurations. It is also useful for optimization, where the analyst seeks to identify the input factor settings that optimize some performance measure. Other researchers address the optimization problem in different ways. One idea is to use gradient estimation techniques in conjunction with steepest ascent (for maximization problems) or steepest descent (for minimization problems). Techniques such as adapting stochastic programming methods are under investigation. For more on experimental design and optimization in the simulation context, see chapter 12 of Banks et al. (2000), Cheng and Lamb (1998), Fu (1994), Fu and Hu (1997), chapter 12 of Law and Kelton (2000), Kleijnen (1987, 1998), Sanchez et al. (1996, 1998), Tew and Wilson (1994).

Finally, you may find that in order to utilization your simulation most effectively you will examine several performance measures rather than just one. Your simulation model can generate many output streams from each run, and these streams are likely to be related to one another in some way. For example, large customer waiting times are likely to be associated with long waiting lines. This means you really have a vector of output measures. Multivariate statistics may be useful for simultaneous estimation and for gaining insight into the relationships between output measures. For details and further references, see Charnes (1995) or Law and Kelton (2000).

6 CONCLUSIONS

Although a 'veritable plethora' of output analysis techniques exists, the ABC's described in this tutorial illustrate that by paying attention to a few basic principles, you will be able to conduct a useful, valid output analysis. This is a great way to get the most from your simulation model! Whether the ultimate purpose of the simulation modeling process is to provide insights into model behavior or to answer specific questions, output analysis is the bridge between the model-building and the decision-making processes.

ACKNOWLEDGMENTS

An earlier version of this paper appeared in Sanchez (1999).

REFERENCES

Alexopoulos, C. and A. F. Seila. 1998. Output data analysis. Chapter 7 in *Handbook of Simulation*, ed. J. Banks. New York: John Wiley and Sons.

Banks, J., J. S. Carson, B. L. Nelson and D. Nichol. 2000. *Discrete-event system simulation*, 3d ed. Upper Saddle River, New Jersey: Prentice-Hall.

Barton, R. R. 1998. Simulation metamodels. In *Proceedings of the 1998 Winter Simulation Conference*, ed. D. J. Medeiros, E. F. Watson, J. S. Carson and M. S. Manivannan, 167–176. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.

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

Charnes, J. M. 1995. Analyzing multivariate output. In *Proceedings of the 1995 Winter Simulation Conference*, ed. C. Alexopoulos, K. Kang, W. Lilegdon and D. Goldsman, 201–208. 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.

Cheng, R. C. H. and J. D. Lamb. 1998. Interactive implementation of optimal simulation experiment designs. In *Proceedings of the 1998 Winter Simulation Conference*, ed. D. J. Medeiros, E. F. Watson, J. S. Carson and M. S. Manivannan, 707–712. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.

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, New Jersey: Institute of Electrical and Electronics Engineers.

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

Fu, M. C. 1994. A tutorial review of techniques for simulation optimization. In *Proceedings of the 1994 Winter Simulation Conference*, ed. J.D. Tew, M.S. Manivannan, D.A. Sadowski, and A.F. Seila, 149–156. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.

Fu, M. C. and J-Q Hu. 1997. *Conditional Monte Carlo, Gradient Estimation and Optimization Applications*. Boston, Massachusetts: Kluwer Academic Publishers.

Goldsman, D. and B. L. Nelson. 1998. Statistical screening, selection, and multiple comparison procedures in computer simulation. In *Proceedings of the 1998 Winter Simulation Conference*, ed. D. J. Medeiros, E. F. Watson, J. S. Carson and M. S. Manivannan, 159–166. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.

Grier, D. A. 1992. Graphical techniques for output analysis. In *Proceedings of the 1992 Winter Simulation Conference*, ed. J.J. Swain, D. Goldsman, R.C. Crain, and J.R. Wilson, 314–319. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.

Hood, S.J. and P.D. Welch. 1993. Response surface methodology and its application in simulation. In *Proceedings of the 1993 Winter Simulation Conference*, ed. G.W. Evans, M. Mollaghasemi, E.C. Russell, and W.E. Biles, 115–122. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.

Kelton, W. D. 1997. Statistical analysis of simulation output. *Proceedings of the 1997 Winter Simulation Conference*, ed. S. Andradóttir, K. Healy, D. Withers, and B. L. Nelson. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.

Khoshnevis, B. 1994. *Discrete systems simulation*. New York: McGraw-Hill.

Kleijnen, J. P. C. 1987. *Statistical tools for simulation practitioners*. New York: Marcel Dekker, Inc.

Kleijnen, J. P. C. 1998. Experimental design for sensitivity analysis, optimization, and validation of simulation models. *Handbook of simulation*, ed. J. Banks. New York: John Wiley and Sons.

L'Ecuyer, P. 1994. Efficiency improvement and variance reduction. In *Proceedings of the 1994 Winter Simulation Conference*, ed. J. D. Tew, M. S. Manivannan, D. A. Sadowski, and A. F. Seila, 122–132. 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.

Lewis, P.A.W. and E.J. Orav. 1989. *Simulation methodology for statisticians, operations analysts, and engineers, volume I*. Belmont, California: Wadsworth, Inc.

Matejcik, F. J. and B. L. Nelson. 1995. Two-stage multiple comparisons with the best for computer simulation. *Operations Research* 43(4):633–640.

Morrice, D. J. and L. W. Schruben. 1993. Simulation factor screening using harmonic analysis. *Management Science* 39 (12): 1459–1476.

Nakayama, M. 1997. Multiple comparison procedures for steady-state simulations. *Annals of Statistics* 25: 2433–2450.

Nelson, B. L. 1992. Designing efficient simulation experiments. In *Proceedings of the 1992 Winter Simulation Conference*, ed. J. J. Swain, D. Goldsman, R. C. Crain, and J. R. Wilson, 126–132. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.

Nelson, B. L. 1995. *Stochastic modeling: analysis and simulation*. New York: McGraw–Hill.

Ripley, B.D. 1987. *Stochastic simulation*. New York: John Wiley & Sons.

Saltelli, A., S. Tarantola and K. P.-S. Chan. 1999. A quantitative model-independent method for global sensitivity analysis of model output. *Technometrics* 41 (1), 39–56.

Sanchez, S. M. 1999. ABC's of output analysis. In *Proceedings of the 1999 Winter Simulation Conference*, ed. P. A. Farrington, H. B. Nembhard, D. T. Sturrock and G. W. Evans, 24–32. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.

Sanchez, S. M. 2000. Robust design: seeking the best of all possible worlds. In *Proceedings of the 2000 Winter Simulation Conference*, ed. J. A. Joines, R. R. Barton, K. Kang and P. A. Fishwick, 69-76. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.

Sanchez, S. M., P. J. Sanchez, J. S. Ramberg and F. Moeeni. 1996. Effective engineering design through simulation." *International Transactions on Operational Research* 3 (2): 169–185.

Sanchez, S. M., P. J. Sanchez and J. S. Ramberg. 1998. "A simulation framework for robust system design." Chapter 12 in *Concurrent Design of Products, Manufacturing Processes and Systems*, ed. B. Wang, 279–314. New York: Gordon and Breach.

Schruben, L. W. and V. J. Cogliano. 1987. An experimental procedure for simulation response surface model identification. *Communications of ACM*, 30 (8): 716–730.

Tew, J. D. and Wilson, J. R. 1994. Estimating Simulation Metamodels Using Combined Correlation-Based Reduction Techniques. *IIE Transactions*, 26 (3): 2–16.

Thesen, A. and L.E. Travis. 1992. *Simulation for decision making*. St. Paul, Minnesota: West Publishing Company.

AUTHOR BIOGRAPHY

SUSAN M. SANCHEZ is a Professor of Operations Research at the Naval Postgraduate School, where she also holds a joint appointment in the Graduate School of Business and Public Policy. She received her B.S. in Industrial and Operations Engineering from the University of Michigan, and her M.S. and Ph.D. in Operations Research from Cornell University. She is a member of INFORMS, DSI, ASA, and ASQ, and is currently Vice President/President Elect of the INFORMS College on Simulation. She serves as Guest Editor-in-Chief of *Naval Research Logistics* and as the Simulation Area Editor for the *INFORMS Journal on Computing*; she is a former associate editor of *Operations Research*. Her e-mail and web addresses are `<ssanchez@nps.navy.mil>` and `<http://diana.or.nps.navy.mil/~susan>`.