Proceedings of the 2000 Winter Simulation Conference
J. A. Joines, R. R. Barton, K. Kang, and P. A. Fishwick, eds.

SIMULATION-BASED OPTIMIZATION

Averill M. Law
Michael G. McComas

Averill M. Law & Associates, Inc.
P.O. Box 40996
Tucson, AZ 85717, U.S.A.

ABSTRACT

In this tutorial we present an introduction to simulation-based optimization, which is, perhaps, the "hottest" topic in discrete-event simulation today. We give a precise statement of the problem being addressed and also experimental results for two commercial optimization packages applied to a manufacturing example with seven decision variables.

1 INTRODUCTION

One of the disadvantages of simulation historically is that it was not an optimization technique. An analyst would simulate a relatively small number of system configurations and choose the one that appeared to the give the best performance. However, based on the availability of faster PCs and improved heuristic optimization techniques (genetic algorithms, simulated annealing, tabu search, etc.) most discrete-event simulation software vendors have now integrated optimization packages into their simulation software or will do so in the very near future. It could arguably be said that optimization is the "hottest" topic in discrete-event simulation today.

The goal of an "optimization" package is to orchestrate the simulation of a sequence of system configurations [each configuration corresponds to particular settings of the decision variables (factors)] so that a system configuration is eventually obtained that provides an optimal or near optimal solution.

In Section 2, we describe the problem that is being addressed by simulation-based optimization. Section 3 gives the results that were obtained from applying two commercial optimization packages to a manufacturing example with seven decision variables, and Section 4 provides a summary. A detailed description of available optimization packages, their vendors, and the heuristic search procedures that they use may be found in Law and Kelton (2000, Section 12.6).

2 STATEMENT OF THE PROBLEM

Let $V_1, V_2, \ldots, V_k$ be decision variables (quantitative factors) for a simulation model. Let $f(v_1, v_2, \ldots, v_k)$ be an output random variable for the simulation model corresponding to the set of values $V_1 = v_1, V_2 = v_2, \ldots, V_k = v_k$.

2.1 Example 1

Consider the manufacturing system consisting of four work stations and three buffers (queues) as shown in Figure 1. Whenever a machine in work station 1 is idle, it will pull a blank (new) part in from an infinite supply. A machine cannot discharge a part if the succeeding buffer is full. Processing times have an exponential distribution with a mean that is given in Table 1. Let V_i (for $i = 1, 2, \ldots, 4$) be the number of machines in work station i and let V_i (for $i = 5, 6, 7$) be the number of buffer positions in buffer $i - 3$. Then $f(3, 2, 2, 3, 3, 1, 2)$ could, for example, be the number of completed parts for a 30-day period for the configuration shown in Figure 1.

Then the optimization problem of interest, in general, is given by the following:

$$\max E[f(v_1, v_2, \ldots, v_k)]$$
$$l_i \le v_i \le u_i$$

subject to the p linear constraints:

$$\begin{aligned} a_{11}v_1 + a_{12}v_2 + \ldots + a_{1k}v_k &\le c_1 \\ a_{21}v_1 + a_{22}v_2 + \ldots + a_{2k}v_k &\le c_2 \\ &\vdots \\ a_{p1}v_1 + a_{p2}v_2 + \ldots + a_{pk}v_k &\le c_p \end{aligned} \tag{1}$$

Thus, we want to maximize the objective function $E[f(v_1, v_2, \ldots, v_k)]$ ["E" means the expected value (or mean) of the random variable $f(v_1, v_2, \ldots, v_k)$] over all possible values of $v_1, v_2, \ldots, v_k$ that satisfy that range constraints $l_i \le v_i \le u_i$

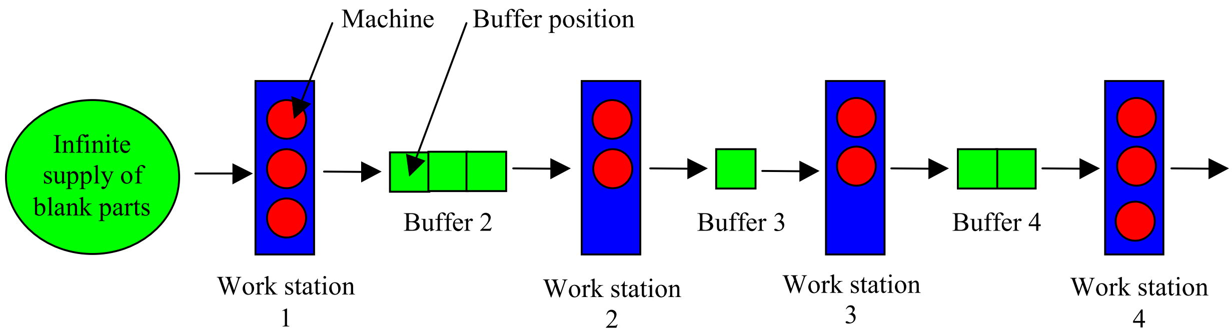

Figure 1: Mean Processing Times for Machines in the Four Work Stations

Table 1: Manufacturing System Consisting of Four Work Stations and Three Buffers

Work station	**Mean processing time for a machine (in hours)**
1	0.33333
2	0.50000
3	0.20000
4	0.25000

(for $i = 1, 2, \ldots, k$) and the linear constraints given by (1). Note that l_i and u_i are lower and upper bounds for v_i. Also, the a_{ji}'s and c_j's in the constraints (1) are constants. Finally, "max" can be replaced by "min" in the objective function. In Example 1, a possible constraint might be

$$v_1 + v_2 + v_3 + v_4 \leq 10$$

i.e., the total number of machines cannot exceed 10

In general, we will need to make n independent replications of the simulation for system configuration v_1, v_2, …, v_k and to use the sample mean over the n replications, $\bar{f}(v_1, v_2, \ldots, v_k)$, as an estimate of $E[f(v_1, v_2, \ldots, v_k)]$, since $f(v_1, v_2, \ldots, v_k)$ is a random variable.

3 A DETAILED EXAMPLE

In this section we apply the OptQuest (Glover et al. 1999) (as implemented in Arena) and WITNESS Optimizer (Lanner 1998) optimization packages to the manufacturing system discussed in Example 1. There are seven decision variables and we assume that $u_i = 3$ for $i = 1, 2, \ldots, 4$ and $u_i = 10$ for $i = 5, 6, 7$; $l_i = 1$ for all values of i. Thus, there are $81{,}000 = 3^4 \cdot 10^3$ different combinations of the decision variables. There are no linear constraints for this problem.

Let

n_ machines = the total number of machines in all work stations

n_positions = the total number of positions in all buffers

throughput = the total number of parts produced in a 30-day period of time

Then define the objective function random variable f (profit) as follows:

$$f = (\$200 \cdot \text{throughput}) - (\$25{,}000 \cdot \text{n\_machines}) - (\$1000 \cdot \text{n\_positions})$$

The simulation run length for our experiments was 720 hours (30 days) with an *additional* warmup period of 240 hours (10 days). The throughput was computed from the final 720 hours of each 960-hour replication. We made $n = 5$ replications for each system configuration for each optimization package.

For OptQuest (Glover et al. 1999), we used a stopping rule that lets the optimization algorithm run until a user-specified number of configurations (*NC*) has been completed. (An alternative stopping rule is to let the optimization algorithm run until a user-specified amount of wall-clock time has elapsed.) Another parameter for OptQuest is the population size (*PS*), which is the number of system configurations that is simultaneously being considered by the algorithm. We considered two different experiments for OptQuest: *PS* =10; *NC* = 100 and *NC* = 300. (This value for *PS* is the smallest one available.) We performed each experiment five times using different random numbers, with the average results being given in Table 2. The average profit is approximately the same for the two different experiments.

The stopping rule for the WITNESS Optimizer (Lanner 1998) has two user-specified parameters: the maximum number of configurations (*MC*) and the number of configurations for which there is no improvement (*CNI*) in the value of the objective function. For example, suppose that *MC* = 500 and *CNI* = 25, and that the objective function value at configuration *i* is the largest up to that point. Then the algorithm will terminate at configuration *i* + 25 if the objective function values at configurations *i* + 1, *i* + 2, …, *i* +25 are all less than or equal to the objective function value at configuration *i*; however, the algorithm will never go beyond 500 configurations. We considered two different experiments: *MC* = 500; *CNI* = 25 and 75. Each experiment was performed five times using different random numbers, with the average results being given in Table 3. The average profit is once again approximately the same for the two different experiments.

Table 2: Average Results (over the Five Realizations) for OptQuest (Version 1.0)

Parameters	**Configurations at termination**	**Configurations at best solution**	**Best objective function value (profit)**
NC = 100	100	59	$591,224
NC = 300	300	136	$596,536

Table 3: Average Results (over the Five Realizations) for the WITNESS Optimizer (Version 2.0a)

Parameters	**Configurations at termination**	**Configurations at best solution**	**Best objective function value (profit)**
CNI = 25	77	56	$588,416
CNI = 75	147	93	$589,256

We have seen that the average profit is approximately $590,000 in the four experiments that we considered. One might ask how close this is to the expected profit for the optimal solution and, also, what is the optimal system configuration? Work station 2 is potentially the bottleneck since its processing rate, 2 parts/hour, is the smallest of the four work stations. Therefore, we can argue heuristically that station 2 should have 3 machines, which gives station 2 a *potential* overall processing rate of 6 parts/hour. It follows that station 3 should have 2 machines – if it had only 1 (an overall processing rate of 5 parts/hour), then station 3 would be the bottleneck. (Three machines at station 3 are clearly not necessary.) By similar reasoning, station 4 should also have 2 machines. The question, then, is how many machines do we need at station 1? It might seem that we need 2 machines at station 1, since its maximum overall processing rate of 6 parts/hour would equal the maximum processing rate of station 2. However, it turns out that 3 machines are preferable, since this results in less idle time and a greater actual processing rate for station 2. The resulting additional profit more than compensates for the cost of one more machine at station 1.

Thus, it would *appear* that 3, 3, 2, and 2 machines at stations 1, 2, 3, and 4, respectively, are optimal. This is substantiated by the fact that the configuration 3, 3, 2, and 2 was selected in 20 out of the 20 experimental realizations (five realizations for each of four experiments).

We therefore fixed the machines at 3, 3, 2, and 2 and set out to determine the optimal number of buffer positions for each of the three buffers. We used the WITNESS Optimizer for this purpose, since it has an option that allows one to do an exhaustive enumeration of all possible system configurations. For each of the 1000 combinations of the numbers of buffer positions, we made n = 50 independent replications of the simulation model – 50,000 replications were made in all. (This experiment was performed only once.) We found that buffer configuration 7, 8, and 4 had an *estimated* profit of $591,588, which was the largest for the 1000 possible configurations. Furthermore, a 90 percent confidence for the expected profit for this configuration was [$591,456, $591,721]. The buffer configuration 7, 7, and 4 was a close second with an estimated profit of $591,512. Therefore, the optimal configuration should be close to 3, 3, 2, 2, 7, 8, and 4. Note that the estimated profit for the configuration *2*, 3, 2, 2, 7, 8, and 4 was only $548,488.

4 SUMMARY

We have tested two different optimization packages with certain settings for their parameters on *one sample problem*. We found that their performance was good for *this* problem and for *the parameter settings used*. One should definitely not infer from these results how these (or other) optimization packages will perform on different problems that might be considerably more difficult in terms of the number of possible system configurations, the shape of the response surface $E[f(v_1, v_2, \ldots, v_k)]$, or the amount of inherent variability in the simulation model. A major concern at this time is how should one select an optimization package's parameters for a particular problem of interest, since little guidance is given in this regard. In the actual conference presentation, we will give a much more extensive set of experimental results, in terms of the number of example problems and of the number the optimization packages tested. In particular, we will see for some optimization packages that the choice of parameter settings can have a big impact on the quality of the solution obtained.

Simulation-based optimization is just in its infancy. However, it appears that it will have a considerable impact on the practice of simulation in the future, particularly when computers become significantly faster.

REFERENCES

Glover, F., J. P. Kelly, and M. Laguna, 1999. *The OptQuest callable library user's documentation*, Optimization Technologies, Inc., Boulder, Colorado.

Lanner Group, Inc., 1998. *WITNESS optimizer module*, Release 2, Houston, Texas.

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

AUTHOR BIOGRAPHIES

AVERILL M. LAW is President of Averill M. Law & Associates, Inc. (Tucson, Arizona), a company specializing in simulation consulting, training, and software. He has been a simulation consultant to more than 100 organizations, including Andersen Consulting, Boeing, Cellular One, DMSO, Kimberly-Clark, M&M/Mars, 3M, Xerox, the U.S. Air Force, and the U.S. Army.

He has presented more than 325 simulation short courses in 17 countries, and delivered more than 100 talks on simulation modeling at technical conferences. He is the author (or coauthor) of three books and numerous papers on simulation, manufacturing, operations research, statistics, and communications, including the textbook *Simulation Modeling and Analysis* that is used by more than 70,000 people worldwide. He is the developer of the ExpertFit software package for selecting simulation input probability distributions, and he has developed several simulation videotapes. Dr. Law wrote a regular column on simulation for *Industrial Engineering* magazine from 1990 through 1991.

He has been a tenured faculty member at the University of Wisconsin and the University of Arizona. Dr. Law has a Ph.D. in industrial engineering and operations research from the University of California at Berkeley.

MICHAEL G. MCCOMAS is Vice President for Consulting Services of Averill M. Law & Associates, Inc. He has considerable simulation modeling experience in a wide variety of application areas, and is the coauthor of seven published papers on simulation. His educational background includes an M.S. in systems and industrial engineering from the University of Arizona.