

NEW DEVELOPMENT OF OPTIMAL COMPUTING BUDGET ALLOCATION FOR DISCRETE EVENT SIMULATION

Hsiao-Chang Chen

Dept. of Systems Engineering
University of Pennsylvania
Philadelphia, PA 19104-6315, U.S.A.

Chun-Hung Chen

Dept. of Systems Engineering
University of Pennsylvania
Philadelphia, PA 19104-6315, U.S.A.

Liyi Dai

Dept. of Systems Science and Mathematics
Washington University
St. Louis, MO 63130, U.S.A.

Enver ücesan

INSEAD
Technology Management Area
Fontainebleau, FRANCE

ABSTRACT

This paper deals with ranking and selection problem via simulation. We present an optimal computing budget allocation technique which can select the best of k simulated designs. This approach can intelligently determine the best simulation lengths for all simulation experiments and significantly reduce the total computation cost to obtain the same confidence level. Numerical testing results are included. Also we provide the results of analysis for some parameters which affect the performance of our approach. Besides, we compare our method with traditional two-stage procedures. Numerical results show that our approach is much faster than the traditional two-stage procedures.

1 INTRODUCTION

In order to design and efficiently manage large man-made systems such as communication networks, traffic systems, and automated manufacturing facilities, it is often necessary to apply extensive simulation to study their performance since no closed-form analytical solutions exist for such problems. Unfortunately, simulation can be both expensive and time consuming. Suppose we want to compare k different system designs. We conduct N simulation replications for each of the k designs. Therefore, we need kN simulation replications. The simulation results become more accurate as N increases. If the accuracy requirement is high (N is not small), and if the total number of designs in a decision problem is not small (k is large), then kN can be very large, which may easily make total simulation cost extremely high and preclude the feasibility of simulation approach. The effective reduction of computation costs while obtaining a good decision is crucial in simulation.

Dudewicz and Dalal (1975) develop a two-stage procedure for selecting the best design or a design which is very close to the best design. In the first stage, all designs are simulated with n_0 replications. Based on the results obtained from the first stage, the number of additional simulation replications is determined for each design in the second stage to achieve the desired confidence level. Rinott (1978) presents alternative way to estimate the number of required simulation replications in the second stage. Many researchers have extended this idea to more general ranking and selection problem in conjunction with new developments (Charnes, 1991, Matejck and Nelson, 1993, Bechhofer, Santner, and Goldsman, 1995, and Hsu, 1996).

To further reduce the overall computation cost, Chen (1995) formulates the procedures of selecting the best design as another optimization problem. The ideas are as follows. Intuitively, some inferior designs can be discarded at an early stage of simulation. As the simulation proceeds, some designs can be further ignored when higher simulation accuracy for the remaining designs is obtained. This procedure is repeated until a desired confidence level is achieved. Proceeding in this fashion, little effort is wasted on simulating inferior designs; the overall simulation time is hence reduced. Then the question is how to systematically do this? When? And which designs? Ideally, we want to optimally choose the number of simulation replications for all designs to minimize the total simulation cost, while obtaining the desired confidence level. In fact, this question is equivalent to optimally decide which designs will receive additional computing budget for continuing simulation or to find an optimal way to reach an optimal design.

Chen et al. (1996) provide an approach to solve such an optimization problem. They use Chernoff bounds to estimate the gradient information and then apply steepest

descent method to solve this optimization problem. While this approach can effectively solve the optimal computing budget allocation problem, the complexity of In this paper, we present another simple way of estimating gradient information. From numerical results we observe that the performance of this new approach is even better than that in Chen et al. (1996). Additionally, we compare our approach with the traditional two-stage procedures by conducting a numerical experiment. Numerical results show that our approach is more than ten times faster than the two-stage procedures.

Since our approach is based on the Bayesian model presented in Chen (1995), we will give an overview of that model for selecting the best design in the next Section. In Section 3, we define the “optimal computing budget allocation” problem and propose a sequential approach. We also demonstrate the numerical testing in a simple example. Section 4 discusses two crucial factors, initial simulation length and one-time incremental computing budget. Section 5 gives a brief review of traditional two-stage procedure and compares our method with the two-stage procedure. Section 6 concludes this paper.

2 PROBLEM FORMULATION AND CONFIDENCE LEVEL QUANTIFICATION

Suppose that our goal is to select a design associated with the smallest mean performance measure among k designs with unknown variances that are not necessarily equal. Further assume that the computing budget is limited and the number of designs is large. Denote

- k : the total number of designs,
- X_{ij} : the j th *i.i.d.* sample of the performance measure from design i ,
- N_i : the number of simulation replications for design i ,
- $\bar{\mu}_i$: the sample mean for design i , $\bar{\mu}_i = \frac{1}{N_i} \sum_{j=1}^{N_i} X_{ij}$,
- μ_i : the mean performance measure; $\mu_i = E(X_{ij})$,
- σ^2 : the variance for design i .

When N 's are large, $\bar{\mu}_i$ can be a good approximation for μ_i , since, according to the law of large numbers, $P\{\bar{\mu}_i \rightarrow \mu_i\} \rightarrow 1$, as $N_i \rightarrow \infty$. Batch means method (Schmeiser 1982) can be used if the simulation samples from any design are not independent. Given the fact that we can conduct only a finite number of simulation replications, $\bar{\mu}_i$ is simply an approximation to μ_i . Using the approximation results to select the best design we have to access the probability of *correct selection* (without loss of generality, we consider minimization problems in this paper; thus, the “best” design means the design with the smallest μ_i). *Correct selection* can be

Chernoff bounds becomes a major obstacle to include some second-order optimization techniques, such as Newton’s method, to achieve a faster convergence rate. defined as a design with the smallest sample performance measure which is actually the best design. In the remainder of this paper, let “CS” denote “correct selection”

There exists a large literature on selecting the best design. Goldsman and Nelson (1994) provide an excellent survey on current approaches (e.g., Goldsman, Nelson, and Schmeiser, 1991, Gupta and Panchapakesan, 1979, and Law and Kelton, 1991) to estimating simulation confidence level. In addition, Bechhofer, Santner, and Goldsman (1995) give a systematic and more detailed discussion on this issue. These approaches are mainly suitable for problems having a small number of designs (e.g. Goldsman and Nelson (1994) suggest 2 to 20 designs). However, for real-life problems, the number of designs can easily grow extremely large. Chen (1996) provides an effective way to quantify confidence level when the number of designs is large. From Chen (1996) we have

$$\begin{aligned}
 &P\{\text{CS}\} \\
 &= P\{\text{a design with the smallest sample mean performance is really the best design}\} \\
 &= \prod_{\substack{i=1 \\ i \neq b}}^k P\{\hat{\mu}_b < \hat{\mu}_i\} = APCS
 \end{aligned} \tag{1}$$

where index b designates the design having the smallest sample mean performance and μ_i is the posterior distribution which consists of information from both prior distribution and the samples $\{X_{ij}, j = 1, 2, \dots, N_i\}$. Under the assumption of normality,

$$\hat{\mu}_i \sim N\left(\frac{1}{N_i} \sum_{j=1}^{N_i} X_{ij}, \frac{\sigma_i^2}{N_i}\right) \text{ for } i = 1, 2, \dots, k.$$

We refer to this lower bound of the correct selection probability as the Approximate Probability of Correct Selection (APCS) While $P\{\text{CS}\}$ is very difficult to obtain, APCS can be computed easily. We will use APCS to approximate $P\{\text{CS}\}$. Numerical testing in Chen (1996) shows that it can provide reasonably good approximation. Furthermore, the sensitivity information of the confidence level with respect to simulation replications can be easily obtained when the approach in Chen (1996) is applied, which provide the basis to determine the allocation of the computing budget in this paper.

3 GRADIENT APPROACH FOR OPTIMAL COMPUTING BUDGET ALLOCATION

We wish to minimize the total computation cost while obtaining a desired confidence level in selecting the best system out of k competing designs. If simulations are performed on a sequential computer, the computation cost can be approximated by $N_1 + N_2 + \dots + N_k$. Ideally we want

$$\begin{aligned} \min(N_1 + N_2 + \dots + N_k) \\ \text{s.t. } APCS \geq P^* \end{aligned} \quad (2)$$

where P^* is a user-defined confidence level requirement. Note that N_1, N_2, \dots, N_k are integers and $APCS(N_1, N_2, \dots, N_k)$ can be computed only after exhausting the total simulation budget, $N_1 + N_2 + \dots + N_k$. Solving this problem can be difficult, especially when k is extremely large. Since the purpose of solving (2) is to reduce computation cost with a desired confidence level, we should not spend too much effort solving (2) during simulation. Otherwise, the additional cost of solving (2) will overwhelm the benefits of the computing budget allocation. Hence, we need to solve (2) very efficiently, even if this means obtaining a sub-optimal solution. Efficiency is more crucial than optimality in this setting.

We now present a sequential approach to determine the number of simulation replications. Before conducting the simulation, there is neither knowledge about APCS nor an idea about how to allocate budget, therefore all designs are simulated with n_0 replications, and the posterior distribution for design i is

$$N\left(\frac{1}{n_0} \sum_{j=1}^{n_0} X_{ij}, \frac{\sigma_i^2}{n_0}\right)$$

We use this statistical information to decide on further allocation. In other words, after running n_0 replications for each design, we have a basic idea about each design and can decide which designs are worthy of being allocated more of the computing budget. Furthermore, let Δ_i be the additional computing budget allocated to design i in each step (Δ_i is a non-negative integer). In order to effectively allocate the computing budget for further simulation, it is necessary to know how $APCS$ would be affected if further replication Δ_i is added independently to each design i . Under a Bayesian model, it is convenient to use the statistical information at N_i to estimate $APCS$ at $N_i + \Delta_i$ by using an approximated posterior distribution

$$N\left(\frac{1}{N_i} \sum_{j=1}^{N_i} X_{ij}, \frac{\sigma_i^2}{N_i + \Delta_i}\right) \text{ for design } i$$

We refer this approximation to *EAPCS* (Estimated Approximate Probability of Correct Selection). We assume that Δ_i is not large and N_i is close to $N_i + \Delta_i$; otherwise *EAPCS* is not a good estimator for *APCS*. Note that N_i can be replaced by n_0 for each design when posterior distribution is approximated at $n_0 + \Delta_i$.

We hope that *APCS* becomes larger as simulation proceeds; we sequentially add computing budget by Δ each time until that *APCS* achieves a satisfactory level P^* . In order to minimize the total computation cost, this budget Δ should be allocated among designs so as to maximize the *EAPCS*. Thus, at step $l, l = 1, 2, \dots$,

$$\begin{aligned} \max_{\Delta_1, \dots, \Delta_k} EAPCS(N_1^l + \Delta_1^l, N_2^l + \Delta_2^l, \dots, N_k^l + \Delta_k^l) \\ \text{s.t. } \sum_{i=1}^k \Delta_i^l = \Delta \text{ and } \Delta_i^l \geq 0 \text{ for all } i. \end{aligned} \quad (3)$$

We assume the system parameter are continuous so that the gradient method (Luenberger 1984) is applied to approximately solve (3). A major issue for using the gradient method is the estimation of gradient information. Chen et al. (1996) apply Chernoff bounds to estimate the gradient information and provide an effective solution to (3). However, the complexity of Chernoff bounds becomes a major obstacle for applying some second-order optimization techniques, such as Newton's method, to achieve a faster convergence rate. In this paper, finite differencing, we approximate the gradient using a straightforward formula:

$$\frac{\partial}{\partial N_i} EAPCS = \frac{APCS(N_1, N_2, \dots, N_i + \tau, \dots, N_k) - APCS(N_1, N_2, \dots, N_i, \dots, N_k)}{\tau}$$

where τ is a small number. To avoid spending too much time in iteratively finding the solution for (3), we only execute a small number of iterations (e.g. 2 times) when applying the gradient method. In summary, we have the following algorithm:

A Sequential Algorithm for Optimal Computing Budget Allocation (OCBA)

- Step 0.** PERFORM SIMULATION with n_0 replication for all designs,
 $l \leftarrow 0$,
 $N_1^l = N_2^l = \dots = N_k^l = n_0$.
- Step 1.** If $APCS(N_1^l, N_2^l, \dots, N_k^l) \geq P^*$, stop,
otherwise, go to Step 2.

Step 2. Solve (3),
 $N_i^{l+1} = N_i^l + \Delta_i^l$, for $i = 1, \dots, k$.
 $l \leftarrow l + 1$.

Step 3. PERFORM SIMULATION until
 $(N_1^l, N_2^l, \dots, N_k^l)$; go to Step 1.

Numerical Testing

To compare the method provided in this paper with our previous approach (Chen et al. 1996), we test a simple $G/G/1$ queue ($k = 10$). There is one server with uniformly distributed service times and customer interarrival times are also distributed uniformly. In this single-node example, all designs have the same arrival time uniformly distributed over $[0.1, 1.9]$, and service time in design i is $\text{uniform}[0.1, 1.3 + 0.05i]$, $i = 1, 2, \dots, 10$. We want to find a design with minimum average system time for customers served in the first 10 time units (terminating simulation). Obviously, higher service rate results in shorter system time in this example, therefore, design 1 is the true best design. In the numerical experiment, we compare the computation costs and the actual convergence probabilities $P\{CS\}$ for different approaches.

We set $\Delta = 12$ and $n_0 = 10$ in this example. To avoid spending too much time in solving (3), we only do two iterations in the gradient method. 10,000 independent experiments are performed so that the average computation cost and $P\{CS\}$ can be estimated. Different confidence level requirements are also tested. Table 1 contains the test results using a Chernoff bound approach (Chen et al. 1996). Table 2 shows the results using the approach presented in this paper.

Table 1. Average total number of simulation replications and $P\{CS\}$ by using Chernoff bounds ($n_0 = 10$ and $\Delta = 12$).

P^*	With Chernoff Bounds	$P\{CS\}$
60%	190.15	0.722
80%	352.73	0.89
90%	541.37	0.962
95%	764.58	0.982

Table 2. Average total number of simulation replications and $P\{CS\}$ for OCBA application ($n_0 = 10$ and $\Delta = 12$).

P^*	With OCBA	$P\{CS\}$
60%	196.47	0.722
80%	344.31	0.866
90%	523.64	0.963
95%	735.43	0.981

From Table 1 and Table 2, we observe that the performances of the two approaches are not much different for small P^* (e.g., $\leq 80\%$), while the new approach is faster when the confidence level requirement P^* is high (e.g., $> 90\%$).

4 SELECTION FOR n_0 AND Δ

To apply our approach, we need to select the initial simulation replication n_0 and one-time increment of computing budget Δ . These two parameters may affect the performance of our approach. In this section we give recommendations for selecting these two parameters, although a good choice can be problem-specific.

Initial Simulation Replication Number n_0

We test the example presented in Section 3 for different values of n_0 . Figure 1 shows the numerical results with respect to different confidence level requirements P^* . It is well understood that n_0 can not be too small; otherwise, the estimates of mean and variance may be very poor, resulting in poor computing budget allocation. On the other hand, if n_0 is too large, we may waste our computation time to reach a confidence level which is much higher than the desired level. Intuitively such an effect is less significant when P^* is large.

The computation cost vs. n_0 for different confidence level requirements is shown in Figure 1. From Figure 1, we observe that the computation cost is not sensitive to n_0 when P^* is greater than 90%. For the cases that P^* is less than 80%, which is not common in real-life applications, a good choice for n_0 is a number between 10 and 20 based on our empirical experiences.

One-time Incremental Computing Budget Δ

Again, the selection of Δ is problem-specific. Generally speaking, large Δ may result in wasting computation time to obtain an unnecessarily high confidence level. On the other hand, if Δ is too small, we need to solve the budget allocation problem many times, diluting the benefit of this approach. We test the same example for different selections of Δ . To have a fair comparison, instead of comparing the simulation replication numbers, we compare total CPU time, which consists of *simulation time* (the CPU time for simulation only) and *optimization time* (the CPU time to solve budget allocation by applying the gradient method). Figure 2 shows the numerical results with respect to $P^* = 80\%$. The optimization time for large Δ is much smaller than that for small Δ . Conversely, the simulation time for large Δ is larger than that for small Δ . We observe that a good selection of Δ for a 10-designs problem is a number between 15 and 30 based on our empirical experiences.

5 COMPARISON WITH OTHER METHODS

In this section we compare our approach with two-stage procedures given by Dudewicz and Dalal (1975) and Rinott (1977) by using the example presented in Section 3. Unlike our Bayesian approach, these two-stage procedures are developed based on classic statistical model. It is convenient to let $(1), \dots, (k)$ denote the unknown indices such that

$$\mu_{(1)} \leq \mu_{(2)} \leq \dots \leq \mu_{(k)}$$

The goal in two-stage procedures is the same as ours: to select a design with the smallest mean, $\mu_{(1)}$. Based on the “indifference-zone” idea we may be willing to choose design (2) if $\mu_{(1)}$ and $\mu_{(2)}$ are very close (e.g. $\mu_{(2)} - \mu_{(1)} \leq d^*$, where d^* is the indifference-zone). By doing so, the procedure can avoid making a large number of simulation replications to separate small difference. More specifically, we intend to determine the number of simulation replications to ensure

$$P\{\bar{\mu}_{(i)} < \bar{\mu}_{(i)}, \forall i \neq 1 | \mu_{(i)} - \mu_{(1)} \geq d^*\} \geq P^*$$

The procedure is as follows. Before performing the simulation we specify three parameters: the confidence level requirement P^* , the simulation replication in the first stage n_0 , and indifference zone d^* . Let h solve Rinott’s or Dudewicz’s integral given P^* , n_0 , and k (h can also be found from the tables in Wilcox (1984) for Rinott’s procedure, and those in Gibbons (1977) for Dudewicz’s procedure). In the first stage, all designs are simulated with n_0 replications. Based on the resulting variance estimate obtained from the first stage, we determine how many additional simulation replications for each design should be conducted in the second stage in order to reach the desired confidence level:

$$N_i = \max(n_0, \lceil (hS_i / d^*)^2 \rceil), \text{ for } i = 1, 2, \dots, k.$$

where $\lceil \bullet \rceil$ is the integer “round-up” function.

Bigger d^* means larger indifference zone, resulting in lower simulation cost. In our previous testing, we are interested in finding the design with smallest mean $\mu_{(1)}$. To have a fair comparison, d^* can not be greater than $\mu_{(2)} - \mu_{(1)}$ which is around 0.059, estimated by using simulation.

Two confidence levels $P^* = 0.9, 0.95$ and initial sample replication $n_0 = 10$ and $n_0 = 20$ have been tested. In these tests, we have 2 iterations in each sequential optimization step when using OCBA to solve (3) solved. For the case of OCBA we apply one-time incremental computing budget $\Delta = 12$. When applying Rinott’s and

Dudewicz’s procedures, we set the indifference zone $d^* = 0.059$ ($\mu_{(2)} - \mu_{(1)} \leq 0.059$). In both cases 10,000 independent experiments are run to evaluate the computational efficiency and to estimate the actual convergence probabilities $P\{CS\}$. The computation cost and $P\{CS\}$ are given in Table 3 and Table 4 corresponding to $n_0 = 10$ and $n_0 = 20$, respectively. From Table 3, significant speedup is observed for our method over both two-stage procedures, while the actual convergence probabilities for all approaches are no less than the desired levels. The average speedup is approximately 13.74 times and 12.48 times over the Rinott’s and Dudewicz’s procedures, respectively, when $P^* = 95\%$ and $n_0 = 20$.

Figures 3 and 4 show the average N_i for all i over the 10,000 experiments using our approach and Rinott’s procedure respectively ($P^* = 90\%$ and $n_0 = 10$). From these two figures, we observe that the budget allocation using our approach is quite different from that using Rinott’s procedure. Two-stage procedures determine the number of simulation replications for all designs using only the information of sample variances. The information on their sample means is not utilized. On the other hand, our approach exploits the information of both sample means and variances, achieving much faster performance.

Table 3. Average total number of simulation runs and the probability of correct selection as $n_0 = 10$.

	$n_0 = 10$			
	$P^*=90\%$	$P\{CS\}$	$P^*=95\%$	$P\{CS\}$
OCBA	523.64	0.962	755.23	0.989
Dudewicz’s	8059.97	0.984	11023.2	0.995
Speedup	15.39		14.6	
Rinott’s	9479.37	0.986	12646.2	0.995
Speedup	18.1		16.5	

Table 4. Average total number of simulation runs and the probability of correct selection as $n_0 = 20$.

	$n_0 = 20$			
	$P^*=90\%$	$P\{CS\}$	$P^*=95\%$	$P\{CS\}$
OCBA	557.64	0.974	738.6	0.989
Dudewicz’s	6832.02	0.975	9152.33	0.989
Speedup	12.25		12.48	
Rinott’s	7978.62	0.99	10150.56	0.995
Speedup	14.31		13.74	

6 CONCLUDING REMARKS

In this paper we present an optimal computing budget allocation technique that can select the best of k simulated designs. We also compare our approach with traditional two-stage procedures by conducting a numerical experiment. Preliminary numerical testing shows that

our approach is more than ten times faster than two-stage procedures. Further testing and analysis of our approach is under way. Also the development of a more efficient budget allocation technique which utilizes methods with higher convergence rate, like Newton's method, is one of our research topics.

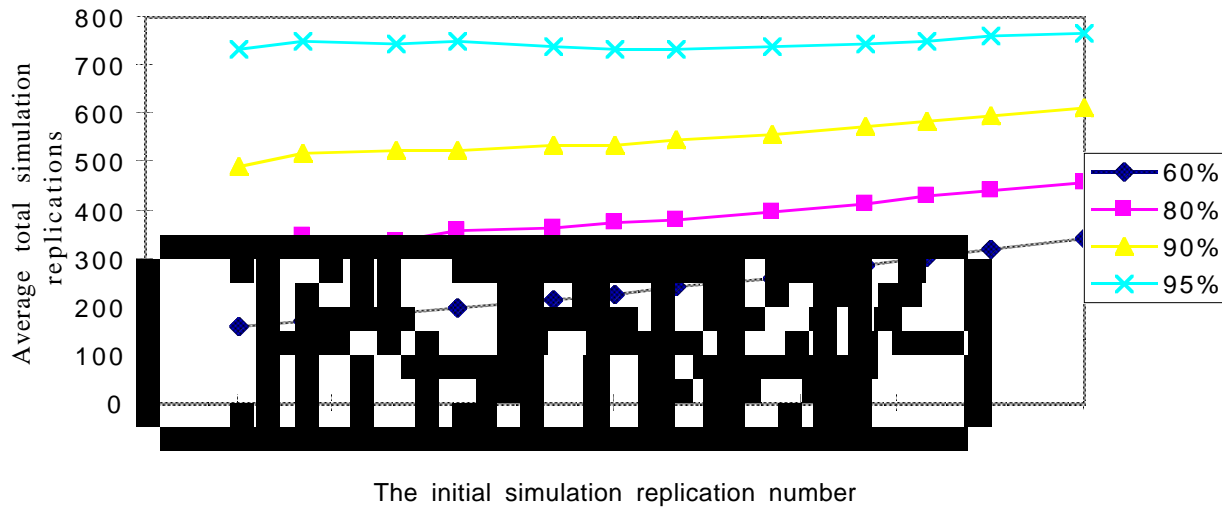


Figure 1. Average total simulation replications vs. n_0 for different P^* as OCBA applied (2 iterations). The results are based on an average over 10,000 replicates ($\Delta=12$).

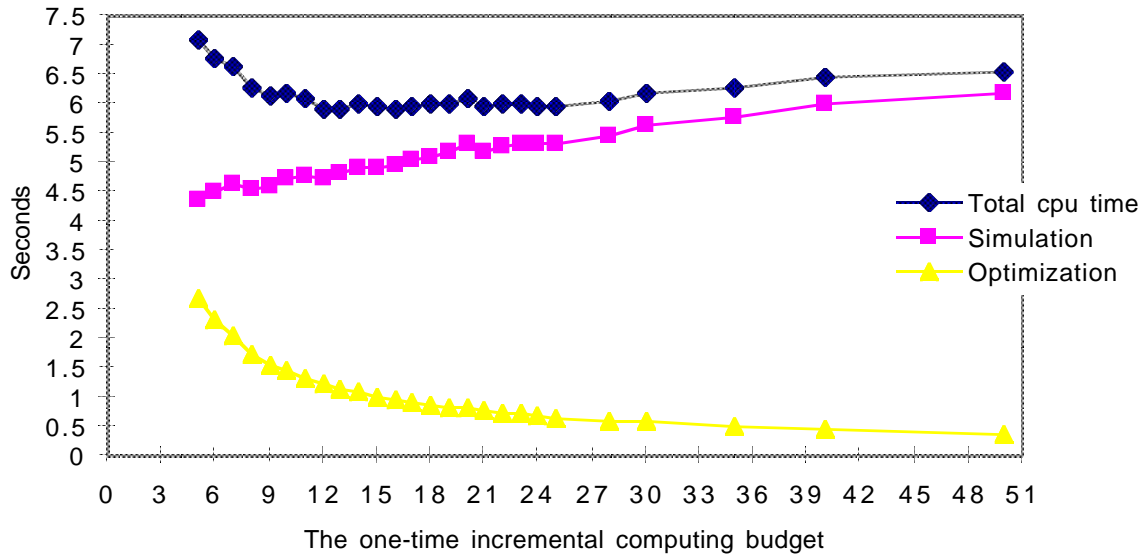


Figure 2. The total CPU time vs Δ for $P^*=80\%$ and $n_0 = 10$ when OCBA is applied (2 iterations). The results are based on an average over 10,000 replicates (total CPU time = simulation time + optimization time).

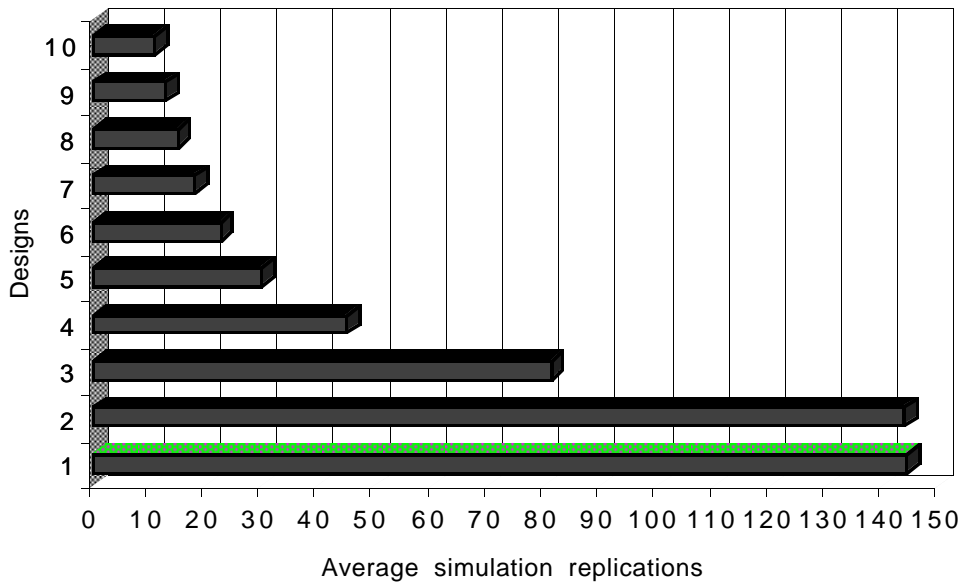


Figure 3. Computing budget allocation are determined by OCBA when $P^* = 0.90$, and $n_0=10$ (system 1 is the best design).

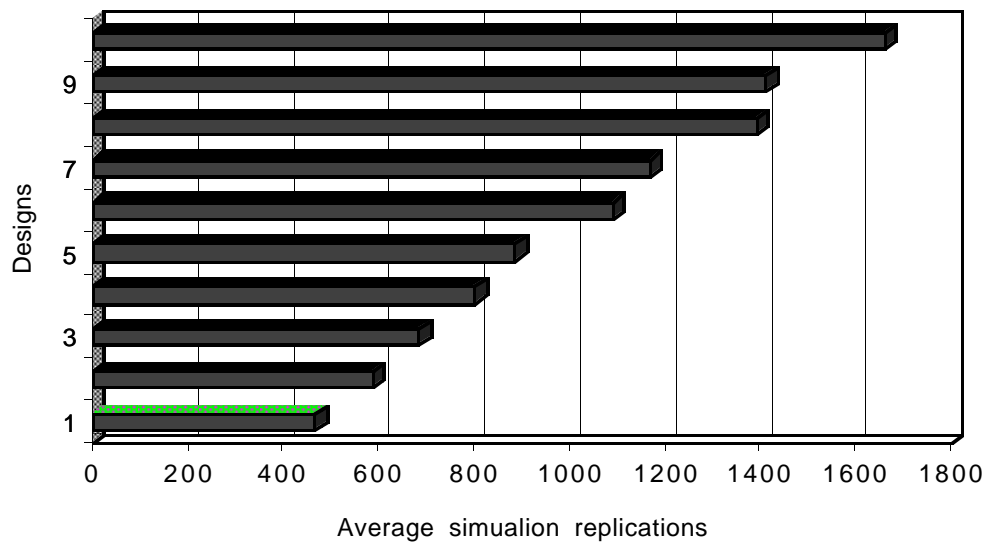


Figure 4. Computing budget allocation determined by Rinott's procedure when $P^*=0.90$, and $n_0=10$ (design 1 is the best design).

ACKNOWLEDGMENTS

This work was supported in part by the National Science Foundation under grant No. ECS-962479.

REFERENCES

Bechofer R. E., Santner, T. J., and Goldman, D. M. 1995. *Design and Analysis of Experiments for Statistical Selection, Screening, and Multiple Comparisons*. John Wiley & Sons, Inc.

- Charnes, J. M. 1991. Multivariate Simulation Output Analysis. *Proceedings of the 1991 Winter Simulation Conference*, 187-193.
- Chen, C. H. 1995. An Effective Approach to Smartly Allocate Computing Budget for Discrete Event Simulation. *Proceedings of the 34th IEEE Conference on Decision and Control*, 2598-2605.
- Chen, C. H. 1996. A Lower Bound for the Correct Subset-Selection Probability and Its Application to Discrete Event System Simulations. *IEEE Transactions on Automatic Control.*, Vol. 41, No. 8, 1227-1231.
- Chen, C. H., Chen, H. C., and Dai, L. 1996. A Gradient Approach for Smartly Allocating Computing Budget for Discrete Event Simulation. *Proceedings of the 1996 Winter Simulation Conference*, 398-405.
- Dudewicz, E. J. and Dalal, S. R. 1975. Allocation of Observations in Ranking and Selection with Unequal Variances. *Sankhya*, B37:28-78.
- Gibbons, J. D., Olkin, I., and Sobel, M. 1977. *Selecting and Ordering Populations: A New Statistical Methodology*. John Wiley & Sons, Inc.
- Goldsman, G., B. L. Nelson, and B. Schmeiser. 1991. Methods for Selecting the Best System. *Proceedings of the 1991 Winter Simulation Conference*, 177-186.
- Goldsman, G., and B. L. Nelson. 1994. Ranking, Selection, and Multiple Comparison in Computer Simulation. *Proceedings of the 1994 Winter Simulation Conference*, 192-199.
- Hsu, J. C. 1996. *Multiple Comparisons Theory and Methods*. Chapman & Hall.
- Gupta, S. S. and S. Panchapakesan. 1979. *Multiple Decision Procedures: Theory and Methodology of Selecting and Ranking Populations*. John Wiley.
- Law, A. M. and W. D. Kelton. 1991. *Simulation Modeling & Analysis*. McGraw-Hill, Inc.
- Luenberger, D. G., 1984. *Linear and Nonlinear Programming*. Addison-Wesley.
- Matejcik, F. J. and Nelson, B. L. 1991. Simultaneous Ranking, Selection and Multiple Comparisons for Simulation. *Proceeding of the 1993 Winter Simulation Conference*, 386-391.
- Rinott, Y. 1978. On Two-stage Selection Procedures and Related Probability Inequalities. *Communications in Statistics* A7, 799-811.
- Schmeiser, B. 1982. Batch Size Effects in the Analysis of Simulation Output. *Operations Research*, Vol. 30, No. 3, , 556-567.
- Wilcox, R. R. 1984. A Table For Rinott's Selection Procedure. *Journal of Quality Technology*, Vol. 16, No. 2, 97-100.

AUTHOR BIOGRAPHIES

HSIAO-CHANG CHEN is a Ph.D. candidate at the Systems Engineering Department, University of Pennsylvania. He received a double B.S. degree in Mathematics and Computer Science from the Eastern Michigan University in 1992, and he received an M.S. degree in Systems Science and Mathematics from Washington University, St. Louis in 1994. He is working on developing efficient approaches for discrete event simulation.

CHUN-HUNG CHEN is an Assistant Professor of Systems Engineering at the University of Pennsylvania, Philadelphia, PA. He received his Ph.D. degree in Simulation and Decision from Harvard University in 1994. His research interests cover a wide range of areas in Monte Carlo simulation, optimal control, stochastic decision processes, ordinal optimization, perturbation analysis, and their applications to manufacturing systems. Dr. Chen won the 1994 Harvard University Eliahu I. Jury Award for the best thesis in the field of control. He is also one of the recipients of the 1992 MasPar Parallel Computer Challenge Award.

LIYI DAI is an assistant professor in the Department of Systems Science and Mathematics at Washington University, MO. He received the M.S. degree from the Institute of Systems Science, Academia Sinica, Beijing, China, in 1986, and the Ph.D. degree from Harvard University in 1993. His research interests include discrete event dynamic systems, simulation, stochastic optimization, communication systems, and singular systems. He has coauthored over 30 papers in various journals and is the author of *Singular Control Systems* (Berlin: Springer-Verlag, 1989). Dr. Dai is listed in *Who's Who* among Asian Americans and is a recipient of the NSF CAREER award.

ENVER YÜCESAN is an associate Professor of Operations Research at INSEAD in Fontainebleau, FRANCE. He holds a BSIE degree from Purdue University, and an MS and a Ph.D. both in OR, from Cornell University. He is currently visiting the Department of Systems Engineering at the University of Pennsylvania.