Published in final edited form as:
Automatica (Oxf). 2014 May 1; 50(5): 1391–1400. doi:10.1016/j.automatica.2014.03.011.

An Efficient Simulation Budget Allocation Method Incorporating Regression for Partitioned Domains*

Mark W. Brantley[a], **Loo Hay Lee**[b], **Chun-Hung Chen**[c], and **Jie Xu**[c]

[a]Goose Point Analysis L.L.C., Covington, LA 70433

[b]Department of Industrial and Systems Engineering, National University of Singapore, Singapore 119077 Logistics Engineering College, Shanghai Maritime University, China

[c]Department of Systems Engineering and Operations Research, George Mason University, Fairfax, VA 22030

Abstract

Simulation can be a very powerful tool to help decision making in many applications but exploring multiple courses of actions can be time consuming. Numerous ranking & selection (R&S) procedures have been developed to enhance the simulation efficiency of finding the best design. To further improve efficiency, one approach is to incorporate information from across the domain into a regression equation. However, the use of a regression metamodel also inherits some typical assumptions from most regression approaches, such as the assumption of an underlying quadratic function and the simulation noise is homogeneous across the domain of interest. To extend the limitation while retaining the efficiency benefit, we propose to partition the domain of interest such that in each partition the mean of the underlying function is approximately quadratic. Our new method provides approximately optimal rules for between and within partitions that determine the number of samples allocated to each design location. The goal is to maximize the probability of correctly selecting the best design. Numerical experiments demonstrate that our new approach can dramatically enhance efficiency over existing efficient R&S methods.

Keywords

simulation; budget allocation; regression

1 Introduction

Simulation optimization is a method to find a design consisting of a combination of input decision variable values of a simulated system that optimizes a particular output

*Corresponding author C.-H. Chen. Tel. +1-703-9933572. Fax +1-703-9931521. mark@goosepointanalysis.com (Mark W. Brantley), iseleelh@nus.edu.sg (Loo Hay Lee), cchen9@gmu.edu (Chun-Hung Chen), jxu13@gmu.edu (Jie Xu).

performance measure of the system. We propose to investigate stochastic problems on a discrete domain with a finite simulation budget consisting of runs conducted sequentially on a single computer. To assess the performance at a single design location on the domain, the uncertainty in the system performance measure requires multiple runs to obtain a good estimate of the performance measure.

When presented with a relatively small number of designs in the domain, the problem we consider is that of selecting the best design from among the finite number of choices. Ranking and Selection (R&S) procedures are statistical methods specifically developed to select the best design or a subset that contains the best design from a set of k competing design alternatives. Rinott [20] developed two-stage procedures for selecting the best design or a design that is very close to the best system. Many researchers have extended this idea to more general R&S settings in conjunction with new developments (e.g., [2]).

To improve efficiency for R&S, several approaches have been explored for problems of selecting a single best design. Intuitively, to ensure a high probability of correct selection (*PCS*) of the best design, a larger portion of the computing budget should be allocated to those designs that are critical in the process of identifying the best design. A key consequence is the use of both the means and variances in the allocation procedures, rather than just the variances, as in [20]. Among examples of such approaches, the Optimal Computing Budget Allocation (OCBA) approach by Chen et al. [9,11] and Lee et al. [17,18] is the most relevant to this paper. OCBA maximizes a simple heuristic approximation of the *PCS*. The approach by Chick and Inoue [12] estimates the *PCS* with Bayesian posterior distributions and allocates further samples using decision-theory tools to maximize the expected value of information in those samples. Branke et al. [3] provide a nice overview and extensive comparison for some of relevant selection procedures.

Brantley et al. [5] take an approach called optimal simulation design (OSD) that is different than most R&S methods by incorporating information from across the domain into a regression equation. Morrice et al. [19] extended the concepts from OSD to a method for selecting the best configuration based on a transient mean performance measure. Unlike traditional R&S methods, this regression based approach requires simulation of only a subset of the alternative design locations and so the simulation efficiency can be dramatically enhanced. While the use of a regression metamodel can dramatically enhance efficiency, the OSD method also inherits some typical assumptions from most DOE approaches. It is assumed that there is an underlying quadratic function for the means and the simulation noise is homogeneous across the domain of interest. Such assumptions are common in some of the DOE literature but become a limit for simulation optimization.

Motivated by iterative search methods (e.g., Newton's method in nonlinear programming) which rely upon a quadratic assumption only in a small local area of the search space during each iteration, we assume that we have several adjacent partitions and that in each partition the mean of the underlying function is approximately quadratic. Thus, we can utilize the efficiency benefit of a regression metamodel. From the perspective of simulation efficiency, we want to determine how to simulate each design point in the different partitions so that the overall simulation efficiency can be maximized.

Specifically, we want to determine i) how much simulation budget to allocate to each partition; ii) which design points in each partition must be simulated from the predetermined set of design points; iii) how many replications should we simulate for those design points? This paper develops a Partitioning Optimal Simulation Design (POSD) method to address these issues. Numerical testing demonstrates that partitioning the domain and then efficiently allocating within the partitions can enhance simulation efficiency, even compared with some existing efficient R&S methods such as OCBA. By incorporating efficient allocations between the partitions in addition to efficient allocation within the partitions, the POSD method offers dramatic further improvements. As compared with only efficiently allocating within each partition, the POSD method offers an improvement over not only the well-known D-optimality approach in DOE literature (by 70~74% reduction) but also the OSD method developed in [5] (by 55%~65% reduction). The rest of the paper is organized as follows.

In Section 2, we introduce the simulation optimization problem setting and Bayesian framework. Section 3 develops an approximate *PCS* while Section 4 provides heuristic approximations of the optimal simulation allocations to maximize the approximate *PCS*. Numerical experiments comparing the results using the new partitioned OSD (POSD) method and other methods are provided in Section 5. Finally, Section 6 provides the conclusions and suggestions for future work using the concepts introduced here.

2 Problem Setting and Bayesian Framework

This paper explores a problem with the principal goal of selecting the best of multiple alternative design locations. Without loss of generality, we assume that we have *m* adjacent partitions and that each partition has *k* design locations. We aim to find the minimization problem shown below in (1) where the "best" design location is the one with smallest expected performance measure

$$\min_{x_{hi}} y(x_{hi})=E[f(x_{hi})];\ x_{hi} \in [x_{11},\cdots,x_{1k},x_{21},\cdots,x_{2k},x_{m1},\cdots,x_{mk}]. \quad (1)$$

Addressing how the domain is partitioned is not within the scope of this paper and we assume this partitioning scheme is derived from knowledge of the domain, through iterative refinement, or through an optimal selection procedure such as multivariate adaptive regression splines (MARS) [14].

In this paper, we consider that the expectation of the unknown underlying function for each partition is quadratic or approximately quadratic in nature on the prescribed domain, i.e., for each partition *h*,

$$y(x_{hi})=\beta_{h0}+\beta_{h1}x_{hi}+\beta_{h2}x_{hi}^2. \quad (2)$$

For ease of notation, we define $\beta_h = [\beta_{h0}, \beta_{h1}, \beta_{h2}]$. In (2), the parameters β_h are unknown and we consider a common case where $y(x_{hi})$ must be estimated via simulation with noise. The simulation output $f(x_{hi})$ is independent from replication to replication such that

$$f(x_{hi})=y(x_{hi})+\varepsilon_h; i=1,\dots,k, \varepsilon_h \sim N(0,\sigma_h^2). \quad (3)$$

The parameters β_h are unknown so $y(x_{hi})$ are also unknown. However, we can estimate expected performance measure at x_{hi}, that we define as $\hat{y}(x_{hi})$, by using a least squares estimate of the form shown in (4) below where $\hat{\beta}_{h0}$, $\hat{\beta}_{h1}$, and $\hat{\beta}_{h2}$ are the least squares parameter estimates for the corresponding parameters associated with the constant, linear, and quadratic terms in (2).

$$\hat{y}(x_{hi})=\hat{\beta}_{h0}+\hat{\beta}_{h1}x_{hi}+\hat{\beta}_{h2}x_{hi}^2. \quad (4)$$

In a similar manner, we define $\hat{\beta}_h = [\hat{\beta}_{h0}, \hat{\beta}_{h1}, \hat{\beta}_{h2}]$. In order to obtain the least squares parameter estimates for each partition, we take n_h samples on any choice of x_{hi} (on at least three design locations for each partition to avoid singular solutions). We assume that these x_{hi} are given beforehand and we can only take samples from these points. Given the n_h samples, we define F_h as the n_h dimensional vector containing the replication output measures $f(x_{hi})$ and X_h as the $n_h \times 3$ matrix composed of rows consisting of $[1, x_{hi}, x_{hi}^2]$ with each row corresponding to its respective entry of $f(x_{hi})$ in F_h. Using the matrix notation and a superscript t to indicate the transpose of a matrix, for each partition we determine the least squares estimate for the parameters β_h which minimize the sum of the squares of the error terms $(F_h - X_h\beta_h)^t (F_h - X_h\beta_h)$. As shown in many regression texts, we obtain the least squares estimate for the parameters as $\hat{\beta}_h=(X_h^t X_h)^{-1}X_h^t F_h$.

Our problem is to select the design location associated with the smallest mean performance measure from among the mk design locations within the constraint of a computing budget with only T simulation replications. Given the least squares estimates for the parameters, we can use (4) to estimate the expected performance measure at each design location. We designate the design location with the smallest estimated mean performance measure in each partition as x_{hb} so that $\hat{y}(x_{hb}) = \min_i \hat{y}(x_{hi})$ and designate x_{Bb} as the design location with the smallest estimated mean performance measure across the entire domain so that $\hat{y}(x_{Bb}) = \min_h \hat{y}(x_{hb})$. Given the uncertainty of the estimate of the underlying function, x_{Bb} is a random variable and we define Correct Selection as the event where x_{Bb} is indeed the best location. We define N_{hi} as the number of simulation replications conducted at design location x_{hi}. Since the simulation is expensive and the computing budget is restricted, we seek to develop an allocation rule for each N_{hi} in order to provide as much information as possible for the identification of the best design location. Our goal then is to determine the optimal allocations to the design locations that maximize the probability that we correctly select the best design (PCS). This Optimal Computing Budget Allocation (OCBA) problem is reflected in (5) below.

$$\max_{N_{11},\dots,N_{mk}} PCS=P\{y(x_{Bb}) \le y(x_{hi}), \forall h=1,\dots,m, i=1,\dots,k\} \text{s.t.} \sum_{h=1}^{m}\sum_{i=1}^{k} N_{hi}=T. \quad (5)$$

The constraint $\sum_{h=1}^{m}\sum_{i=1}^{k}N_{hi}=T$ denotes the total computational cost and implicitly assumes that the simulation execution times for one sample are constant across the domain.

The nature of this problem makes it extremely difficult to solve. To understand the underlying functions for each partition $y(x_{hi})$, we must conduct simulation runs to obtain $f(x_{hi})$, which is a measure of the system performance. Compounding this property is the fact that $f(x_{hi})$ is a function of the random variable ε_h. To even assess the performance at one point on the partition, the uncertainty in the system performance measure requires multiple runs to obtain good approximations of the performance measure. Since the optimal allocation is dependent upon the uncertainty of the parameters and the random variable x_{Bb}, we can only estimate the *PCS* even after exhausting the total simulation budget *T*. Incorporating the information from the underlying functions of each partition adds an additional level of complexity to the derivation of the optimal allocations; however, it is this concept that we aim to exploit in order to provide a significant improvement in the ability to maximize *PCS*.

In order to solve the problem in (5), we must obtain estimates for the parameters β_h. Due to the ease of the derivation, we will proceed with a Bayesian regression framework where the parameters β_h are assumed to be unknown and are treated as random variables. We aim to find the posterior distributions of β_h as the simulation replications are conducted and use these distributions to update the posterior distribution of the performance measures for each design location. We can then perform the comparisons with the performance measure at design location x_{Bb} as expressed in (5).We will use $\tilde{\beta}_h$ and $\tilde{y}(x_{hi})$ to denote the random variables whose probability distributions are the posterior distribution of β_h and $y(x_{hi})$ conditional on F_h given samples respectively. Therefore, given a set of initial n_h simulation runs with the output contained in vector F_h and the design location x_{Bb} obtained from the least squares results derived in the previous section, we can redefine *PCS* from (5) based on the Bayesian concept [8,10] as

$$PCS=P\{\tilde{y}(x_{Bb} \le \tilde{y}(x_{hi}), \forall h=1,\dots,m, i=1,\dots,k\}. \quad (6)$$

Using a non-informative prior distribution and assuming that the conditional distribution of the simulation output vector F_h is a multi-variate normal distribution with mean $X_h\beta_h$ and a covariance matrix $\sigma_H^2 I$ where I is an identity matrix, DeGroot ([13]) shows that the posterior distribution of β_h is then given by

$$\tilde{\beta}_h \sim N[(X_h^t X_h)^{-1} X_h^t F_h, \sigma_h^2 (X_h^t X_h)^{-1}]. \quad (7)$$

Since $\tilde{y}(x_{hi})$ is a linear combination of $\tilde{\beta}_h$, we have

$$\tilde{y}(x_{hi}) \sim N[X_{hi}(X_h^t X_h)^{-1} X_h^t F_h, \sigma_h^2 X_{hi}^t (X_h^t X_h)^{-1} X_{hi}], \quad (8)$$

where $X_{hi}^t=[1, x_{hi}, x_{hi}^2]$.

Similar to the approach in [5] for the one partition case, we are interested in how the *PCS* in (6) changes if we conduct additional runs before we actually conduct the simulation replications so that we can make allocations that maximize the *PCS* in (5). See [5] for a detailed discussion of the predictive posterior distributions.

In order to further simplify (5), the following theorem allows us to reduce the number of support points required for our allocations for each partition.

Theorem 1 *Given that we assume the expectation of our underlying function is quadratic within each partition, we require only three support points on each partition and two of these support points will be at the extreme design locations (x_{h1} and x_{hk}) on each partition*.

Proof: Having established that the *PCS* criterion conforms to a Loewner ordering, Brantley et al. ([5]) then utilize the results commonly used in the DOE literature [16].

Given the results of Theorem 1, we will refer to the support points for each partition as $\{x_{h1}, x_{hs}, x_{hk}\}$ where $x_{h1} \le x_{hs} \le x_{hk}$. (Note that since x_{hb} may be at different locations on each partition, then x_{hs} may also be at different locations on each partition). For notation sake, we define the number of runs allocated to partition h as $N_{h\cdot}$ and the percentage of $N_{h\cdot}$ that is allocated to each support point as $\alpha_{hi} = N_{hi} / N_{h\cdot}$, $i \in \{1, s, k\}$. Using this notation and the *PCS* equation in (6), we can now restate the OSD problem in Equation (5) as the OSD problem in Equation (9) below.

$$\max P\{\tilde{y}(x_{Bb}) \le \tilde{y}(x_{hi}) \forall h=1,\ldots,m, i=1,\ldots,k\} \text{s.t.} \sum_{h=1}^{m} N_{h\cdot}(\alpha_{h1}+\alpha_{hs}+\alpha_{hk})=T. \quad (9)$$

We can estimate σ_h^2 from our least squares results and can calculate $\tilde{y}(x_{hi})$ using (8). While *PCS* can then be estimated using Monte Carlo simulation with (9), it can be very time consuming. The next section reduces the number of comparisons required and presents a way to approximate the *PCS* without running Monte Carlo simulations.

3 Approximate *PCS*

The previous section demonstrated how we can utilize the quadratic structure of the underlying function in order to provide estimates of the performance measure across each partition and to reduce the number of design locations that will receive simulation allocations. In this section, we will find an approximation for our *PCS* equation and then express the approximation in terms of the number of simulations allocated to each design location.

Upon inspection, the *PCS* equation in (9) has two types of comparisons that are delineated in (10). The first type consists of the $k - 1$ comparisons between $\tilde{y}(x_{Bb})$ and each $\tilde{y}(x_{Bi})$ for $i \ne b$ in the best partition. The second type consists of the $k(m-1)$ comparisons between $\tilde{y}(x_{Bb})$ and each $\tilde{y}(x_{hi})$ when $h \ne B$.

$$PCS=P\{(\tilde{y}(x_{Bb}) \le \tilde{y}(x_{Bi}) \forall i \ne b) \cap (\tilde{y}(x_{Bb}) \le \tilde{y}(x_{hi}) \forall h \ne B, i=1,\ldots,k)\}. \quad (10)$$

Given x_{Bb} estimated from the second order polynomial metamodel results, the assumption that our underlying function is quadratic within each partition allows us to reduce the required number of comparisons within the best partition from the $k - 1$ comparisons expressed in (10) to two comparisons. For the interior design case shown in (11) below, Brantley et al. ([5]) show using the assumption of a underlying function that is quadratic that we have correctly selected if the design that we have selected is better than both of its neighboring designs. They also show in a similar manner for the two boundary cases in (11) that we know that we have correctly selected if the selected design is better than both the adjacent design and the opposite boundary design. As such, (10) can be rewritten as shown in (11) subject to the three cases following (11).

$$PCS=P\{(\tilde{y}(x_{Bb}) \leq \tilde{y}(x_{Bi})\forall i=\mathrm{A},\mathrm{Z})\cap(\tilde{y}(x_{Bb}) \leq \tilde{y}(x_{hi})\forall h \neq B, i=1,\dots,k)\}. \quad (11)$$

Case 1(Interior Design Case) $b \neq 1, k$; A = $b - 1$; Z = $b + 1$,

Case 2(Left Boundary Design Case) $b = 1$; A = 2; Z = k,

Case 3(Right Boundary Design Case) $b = k$; A = 1; Z = $k - 1$.

We have the same assumption of an underlying function that is quadratic in the non-best partitions also. However, the comparisons in (10) for the non-best partitions are against $\tilde{y}(x_{Bb})$ instead of the local best $\tilde{y}(x_{hb})$. If we apply the Bonferroni inequality to the comparisons with the global best for a non-best partition, we obtain

$$P\{\tilde{y}(x_{Bb}) \leq \tilde{y}(x_{hi}), i=1,\dots,k\} \geq 1 - \sum_{i=1}^{k} P\{\tilde{y}(x_{Bb}) \geq \tilde{y}(x_{hi})\}. \quad (12)$$

We can also establish a different lower bound for the comparisons from a non-best partition by using the quadratic information within the partition as expressed in the following lemma.

Lemma 2 *Subject to the conditions expressed in Case 1 – Case 3 after* (11), *a lower bound for the comparisons with the global best for a non-best partition can be expressed by using the quadratic information within the partition as shown in* (13).

$$P\{\tilde{y}(x_{Bb}) \leq \tilde{y}(x_{hi}), i=1,\dots,k\} \geq 1-P\{\tilde{y}(x_{Bb}) \geq \tilde{y}(x_{hb})\}-P\{\tilde{y}(x_{hb}) \geq \tilde{y}(x_{hA})\}-P\{\tilde{y}(x_{hb}) \geq \tilde{y}(x_{hZ})\}. \quad (13)$$

Proof: See [5].

For ease of discussion, we will refer to comparisons involving design locations from more than one partition as “between partition” comparisons and we will refer to comparisons involving design locations from just one partition as “within partition” comparisons.

Given these two possible lower bounds for each non-best partition, applying the Bonferroni inequality to (11) yields a lower bound for our *PCS* as shown in (14), which we will consider our approximate *PCS* (*APCS*).

NIH-PA Author Manuscript
NIH-PA Author Manuscript

$$\begin{aligned}PCS \geq APCS = 1 &- P\{\tilde{y}(x_{Bb}) \geq \tilde{y}(x_{BA})\} \\ &- P\{\tilde{y}(x_{Bb}) \geq \tilde{y}(x_{BZ})\} \\ &- \sum_{h \neq B} \min(\sum_{i=1}^{k} P\{\tilde{y}(x_{Bb}) \geq \tilde{y}(x_{hi})\}, P\{\tilde{y}(x_{Bb}) \geq \tilde{y}(x_{hb})\} \\ &+ P\{\tilde{y}(x_{hb}) \geq \tilde{y}(x_{hA})\} \\ &+ P\{\tilde{y}(x_{hb}) \geq \tilde{y}(x_{hZ})\}).\end{aligned} \tag{14}$$

To simplify the notation later in the paper, we will define the set of partitions Ψ as those partitions where we use the lower bound associated with (13) such that

$$\Psi = \{h: \sum_{i=1}^{k} P\{\tilde{y}(x_{Bb}) \geq \tilde{y}(x_{hi})\} \geq P\{\tilde{y}(x_{Bb}) \geq \tilde{y}(x_{hb})\} + P\{\tilde{y}(x_{hb}) \geq \tilde{y}(x_{hA})\} + P\{\tilde{y}(x_{hb}) \geq \tilde{y}(x_{hZ})\}\}.$$

Using this definition of Ψ, we can write (14) in an alternate form as

$$\begin{aligned}APCS \geq 1 &- P\{\tilde{y}(x_{Bb}) \geq \tilde{y}(x_{BA})\} \\ &- P\{\tilde{y}(x_{Bb}) \geq \tilde{y}(x_{BZ})\} \\ &- \sum_{h \neq B, h \notin \Psi} \sum_{i=1}^{k} P\{\tilde{y}(x_{Bb}) \geq \tilde{y}(x_{hi})\}, - \sum_{h \neq B, h \in \Psi} P\{\tilde{y}(x_{Bb}) \geq \tilde{y}(x_{hb})\} \\ &+ P\{\tilde{y}(x_{hb}) \geq \tilde{y}(x_{hA})\} \\ &+ P\{\tilde{y}(x_{hb}) \geq \tilde{y}(x_{hZ})\}).\end{aligned}$$

Using this alternate form of the A*PCS* in (14), we can now restate the POSD problem in (9) as the POSD problem in (15) below with three cases.

<u>POSD Problem</u>

$$\begin{aligned}&\max 1 - P\{\tilde{y}(x_{Bb}) \geq \tilde{y}(x_{BA})\} - P\{\tilde{y}(x_{Bb}) \geq \tilde{y}(x_{BZ})\} - \sum_{h \neq B, h \notin \Psi} \sum_{i=1}^{k} P\{\tilde{y}(x_{Bb}) \geq \tilde{y}(x_{hi})\}, - \sum_{h \neq B, h \in \Psi} P\{\tilde{y}(x_{Bb}) \geq \tilde{y}(x_{hb})\} + P\{\tilde{y}(\\ &\text{s.t.} \sum_{h=1}^{m} N_{h\cdot}(\alpha_{h1} + \alpha_{hs} + \alpha_{hk}) = T; \\ &\alpha_{hi} = 0 \forall i \neq 1, s, k\end{aligned} \tag{15}$$

Case 1(Interior Design Case) $b \neq 1, k$; A = $b - 1$; Z = $b + 1$,

Case 2(Left Boundary Design Case) $b = 1$; A = 2; Z = k,

Case 3(Right Boundary Design Case) $b = k$; A = 1; Z = $k - 1$.

4 Approximations of the Optimal Allocations

In this section, we will derive an efficient heuristic approximations of the optimal allocations of simulation runs to the designated support points $\{x_{h1}, x_{hs}, x_{hk}\}$. Since our aim is to efficiently allocate the computing budget to the three support points in each partition, we will rewrite the *APCS* equation in (15) so that it is expressed in terms of the number of simulation runs allocated to each partition and the percentage of these partition allocations that is allocated to each support point within the partitions. For the within partition comparisons, define $\tilde{d}(x_{hi}) \equiv \tilde{y}(x_{hi}) - \tilde{y}(x_{hb}) = \tilde{\beta}_{h2}(x_{hi}^2 - x_{hb}^2) + \tilde{\beta}_{h1}(x_{hi} - x_{hb})$. This result shows that $\tilde{d}(x_{hi})$ is a linear combination of the β_h elements so the $\tilde{d}(x_{hi})$ terms are also normally distributed. Using the results of Section 2, $\tilde{d}(x_{hi}) \sim N(\hat{d}(x_{hi}), \zeta_{hi})$ where $\hat{d}(x_{hi}) \equiv \hat{y}(x_{hi}) - \hat{y}(x_{hb})$. As shown in [5],

$$\zeta_{hi} = \frac{\sigma_h^2}{N_{h\cdot}}\left[\frac{D_{hi,1}^2}{\alpha_{h1}} + \frac{D_{hi,s}^2}{\alpha_{hs}} + \frac{D_{hi,k}^2}{\alpha_{hk}}\right], \text{ where}$$

$$\begin{aligned} D_{hi,1} &= \frac{(x_{hs}-x_{hi})(x_{hk}-x_{hi}) - (x_{hs}-x_{hb})(x_{hk}-x_{hb})}{(x_{h1}-x_{hs})(x_{h1}-x_{hk})}, \\ D_{hi,s} &= \frac{(x_{h1}-x_{hi})(x_{hk}-x_{hi}) - (x_{h1}-x_{hb})(x_{hk}-x_{hb})}{(x_{hs}-x_{h1})(x_{hs}-x_{hk})}, \\ D_{hi,k} &= \frac{(x_{h1}-x_{hi})(x_{hs}-x_{hi}) - (x_{h1}-x_{hb})(x_{hs}-x_{hb})}{(x_{hk}-x_{h1})(x_{hk}-x_{hs})}. \end{aligned} \quad (16)$$

For the between partition comparisons, define $\tilde{\delta}(x_{hb}) \equiv \tilde{y}(x_{hb}) - \tilde{y}(x_{Bb})$. As with the within partition comparisons, this shows $\tilde{\delta}(x_{hb})$ is a linear combination of the $\tilde{\beta}_h$ elements so the $\tilde{\delta}(x_{hb})$ terms are also normally distributed. Using the results of Section 2, $\tilde{\delta}(x_{hb}) \sim N(\hat{\delta}(x_{hb}), \xi_{hb})$, where $\hat{\delta}(x_{hb}) \equiv \hat{y}(x_{hb}) - \hat{y}(x_{Bb})$. Assuming independence of the simulation runs between partitions and as shown in [4],

$$\zeta_{hi} = \frac{\sigma_h^2}{N_{h\cdot}}\left[\frac{E_{hb,1}^2}{\alpha_{h1}} + \frac{E_{hb,s}^2}{\alpha_{hs}} + \frac{E_{hb,k}^2}{\alpha_{hk}}\right] + \frac{\sigma_B^2}{N_{B\cdot}}\left[\frac{E_{Bb,1}^2}{\alpha_{B1}} + \frac{E_{Bb,s}^2}{\alpha_{Bs}} + \frac{E_{Bb,k}^2}{\alpha_{Bk}}\right], \quad (17)$$

where $E_{hi,1} = [(x_{hs} - x_{hi})(x_{hk} - x_{hi})]/[(x_{h1} - x_{hs})(x_{h1} - x_{hk})]$, $E_{hi,s} = [(x_{h1} - x_{hi})(x_{hk} - x_{hi})]/[(x_{hs} - x_{h1})(x_{hs} - x_{hk})]$, $E_{hi,k} = [(x_{h1} - x_{hi})(x_{hs} - x_{hi})]/[(x_{hk} - x_{h1})(x_{hk} - x_{hs})]$.

To simplify the notation, we define probability $P_\Omega = \max_{h \neq B; i=1,\ldots,k} [P(\{\tilde{\delta}(x_{hi}) \geq 0\}]$. This probability is the most competitive comparison from among the $(m - 1)/k$ between partition comparisons. We also define for the best partition probability P_{BM} and, for $h \neq B$, we define probability P_{hM} such that for $h = B$

$$M = \underset{\mathrm{A,Z}}{\operatorname{argmax}}\left[P\{-\tilde{d}(x_{BA}) \geq 0\}, P\{-\tilde{d}(x_{BZ}) \geq 0\}\right], \; P_{BM} = \max\left[P\{-\tilde{d}(x_{BA}) \geq 0\}, P\{-\tilde{d}(x_{BZ}) \geq 0\}\right], \quad (18a)$$

and, for $h \neq B$

$$M=\begin{cases} \arg\ \max_{\mathrm{A,Z},b}\left[P\{-\tilde{d}(x_{h\mathrm{A}})\geq 0\}, P\{-\tilde{d}(x_{h\mathrm{Z}})\geq 0\}, P\{-\tilde{\delta}(x_{hb})\geq 0\}\right], h\in\Psi, \\ \arg\ \max_{i=1,\cdots,k}\left[P\{-\tilde{\delta}(x_{hb})\geq 0\}\right], h\notin\Psi, \end{cases}$$
$$P_{hM}=\begin{cases} \max_{\mathrm{A,Z},b}\left[P\{-\tilde{d}(x_{h\mathrm{A}})\geq 0\}, P\{-\tilde{d}(x_{h\mathrm{Z}})\geq 0\}, P\{-\tilde{\delta}(x_{hb})\geq 0\}\right], h\in\Psi, \\ \max_{i=1,\cdots,k}\left[P\{-\tilde{\delta}(x_{hb})\geq 0\}\right], h\notin\Psi. \end{cases} \quad (18b)$$

Finally, we define R_B for the best partition and R_h for $h \neq B$ as shown below.

$$R_B=\frac{\hat{d}^2(x_{BM})}{\sigma_B^2[\,|D_{BM,1}|+|D_{BM,s}|+|D_{BM,k}|]^2} \quad (19a)$$

$$R_h=\begin{cases} \frac{\hat{d}^2(x_{hM})}{\sigma_B^2[\,|D_{hM,1}|+|D_{hM,s}|+|D_{hM,k}|]^2}\text{when} P_{hM}=P\{-\tilde{d}(x_{h\mathrm{A}})\geq 0\}, P\{-\tilde{d}(x_{hZ})\geq 0\}, \\ \frac{\hat{\delta}^2(x_{hb})}{\sigma_h^2}\text{otherwise}. \end{cases} \quad (19b)$$

Theorem 3 *Using lower and upper bounds of the APCS, approximately optimal between partition allocations and within partition allocations are as shown in* (20a) *and* (20b) *for the best partition and* (21a) *and* (21b) *for h ≠ B. For brevity, we use OSD to refer to allocations in accordance with the OSD conditions in (B1) and as derived for the one partition case presented in* [5]. *For h = B,*

$$N_{B\cdot}=\begin{cases} T, P_{\mathrm{BM}}\geq P_\Omega, \\ \frac{R_jN_{j\cdot}}{R_i}, P_{\mathrm{BM}}<P_\Omega; h\in\Psi, \mathrm{M{=}A,Z}\forall h, \\ \sigma_B^2\sqrt{\sum_{h\notin\Psi}^m\frac{N_{h\cdot}^2}{\sigma_h^2}+\sum_{h\in\Psi,\mathrm{M}=b}^m\frac{N_{h\cdot}^2}{\sigma_h^2}}, \textit{otherwise} \end{cases} \quad (20a)$$

$$\alpha_{Bi}=\begin{cases} OSD, & P_{\mathrm{BM}}\geq P_\Omega, \\ OSD, & P_{\mathrm{BM}}<P_\Omega; h\in\Psi, \mathrm{M{=}A,Z}\forall h, \\ \alpha_{Bb}=1.0, & \textit{otherwise} \end{cases} \quad (20b)$$

For h ≠ B,

$$N_{h\cdot}=\begin{cases} 0, & P_{\mathrm{BM}}\geq P_\Omega, \\ \frac{R_jN_{j\cdot}}{R_h}, & \textit{otherwise.} \end{cases} \quad (21a)$$

$$\alpha_{hi}=\begin{cases} OSD, P_{hM}=P\{-\tilde{d}(x_{h\mathrm{A}})\geq 0\}, P\{-\tilde{d}(x_{hZ})\geq 0\}, \\ \alpha_{h\mathrm{M}}=1.0, \textit{otherwise}. \end{cases} \quad (21b)$$

Proof: See Appendix A.

Given the results from Theorem 2 expressed in (21a–b) for the non-best partitions, we now revisit how to approximate the probabilities in (12) and (13) in order to choose between the two different lower bounds expressed in the *APCS* in (15). As an approximation, we use the Cantelli inequality [21] such that for within partition comparisons

$$P\{-\tilde{d}(x_{hi})\geq 0\}\leq \frac{1}{1+\hat{d}^2(x_{hi})/\zeta_{hi}}.$$

A similar expression can be made for between partition comparisons such that for each partition *h* in (15), we seek

$$\min\left(\sum_{i=1}^{k}\frac{1}{1+\hat{\delta}^2(x_{hi})/\xi_{hi}},\frac{1}{1+\hat{\delta}^2(x_{hb})/\xi_{hb}}+\frac{1}{1+\hat{d}^2(x_{hA})/\zeta_{hA}}+\frac{1}{1+\hat{d}^2(x_{hZ})/\zeta_{hZ}}\right). \quad (22)$$

The following is the algorithm that we used to implement the POSD method for the experiments in this paper:

Algorithm 1 (OSD Procedure (Maximizing *PCS*))

INPUT: *k (the number of design locations), T (the computing budget), x_i (the design locations with partitions already determined), n (the number of initial runs), θ_j (the number runs allocated each iteration j);*

INITIALIZE: *$j \leftarrow 0$; Perform n_0 simulation replications for three design locations in each partition; by convention we use the D-opt support points such that*
$\alpha^j_{h1}=\alpha^j_{h(k+1)/2}=\alpha^j_{hk}=n_0/3$.

LOOP WHILE $\sum_{i=1}^{k}N_i^j<T$ ***DO***

UPDATE *:*

- Estimate a quadratic regression equation using the information from all prior simulation runs for each partition.
- *Estimate the mean and variance of each design location using* (4).
- *Determine the observed global best design so that* $x_{Bb} = \arg\min_{Bi} \hat{y}(x_i)$ *and the local best design in each partition so that* $x_{hb} = \arg\min_{hi} \hat{y}(x_i)$.
- *Based upon the location of the best design in each partition, use* (15) *to determine* x_{hA} *and* x_{hZ}.
- *Determine* P_{BM} *and* P_{hM} *using* (18a–b) *and corresponding* R_B *and* R_h *using* (19a–b).
- *Determine* $P_{\Omega} = \arg\max_{h\neq B; i=1,\ldots,k} [P\{-\tilde{\delta}(x_{hi})\geq 0\}]$.
- *Determine* $h \in \Psi$ *using* (22).

ALLOCATE *: Increase the computing budget by* θ_{j+1} *and calculate the new between budget allocations* $N_{h\cdot}^{j+1}$ *using* (20a) *and* (21a) *(round as needed). Using* $N_{h\cdot}^{j+1}$ *as well as* (20b) *and* (21b), *determine the within budget allocations for* $\alpha_{h1}^{j+1}, \alpha_{hs}^{j+1}$ *and* α_{hk}^{j+1} *(round as needed).*

SIMULATE *: Perform* α_{hi}^{j+1} *simulations for partition* h, $h = 1, \ldots, m$; *design* i, $i = 1, s, k$; $j \leftarrow j + 1$.

END OF LOOP

5 Numerical Experimentation

In this section, we describe how we compared the results from our new POSD method against the results from five other allocation procedures. We start by providing a description of the other methods chosen to provide a perspective of the efficiency gained by using optimal allocations, by using the information from a regression equation, and by optimally allocating between and within the partitions. We then describe our testing framework and provide our experimental results.

5.1 Comparison Methods

The simplest allocation case is a naive method that equally allocates (EA) the runs to each design location such that $N_i = T/k$ for each i. For this method, we designate the design location with the smallest mean performance measure as x_b so that

$$x_b = \underset{i}{\operatorname{argmin}} \frac{\sum_{j=1}^{T/k} f(x_i)}{T/k}.$$

Instead of equally allocating, we also tested the OCBA method, which is one of the efficient R&S performers [3]. This method requires a set of initialization runs and, based upon the findings of [11], we used an initial allocation of 5 runs for each design location. We then used the method described in [11] to determine how to allocate 99 additional runs within each partition.

Both EA and OCBA rely upon comparisons of the mean response at the global best design location and each individual design location and do not rely upon a response surface within each partition to aid in the comparisons. For our experiments, we will also compare against three methods that utilize a response surface within partitions. The first of these response methods in our progression equally allocates to each design location but uses a response surface (EA-RS) within each partition to compare the results. For EA-RS, each partition will receive an equal number of runs if each partition has an equal number of designs.

The second response method leverages the results from [16] in which we require only three support points for each partition to capture all of the information in the response. A very popular way to do this in DOE literature is to use the D-optimality criterion (D-opt) that maximizes the determinant of the information matrix resulting in minimizing the generalized variance of the parameter estimates. Atkinson and Donev ([1]) provide a list of properties of this criterion and note that D-opt often performs well compared to other criteria. For an underlying quadratic function, this criterion establishes support points at the two extreme points and at the center of the domain and allocates one third of the simulation budget to each of these support points. Using the notation from our early POSD derivation, this criterion will always allocate with $\{\alpha_{h1}, \alpha_{h,(k+1)/2}, \alpha_{hk}\} = \{1/3, 1/3, 1/3\}$. For our experiments, the simulation budget within a partition is allocated in accordance with D-opt and we will equally allocate to each partition.

The final method that we will compare against is a direct use of the OSD method [5]. The simulation budget within a partition is allocated in accordance with the OSD conditions, while different partitions receive an equal amount of the computing budget. We will initialize as described in [5] with $N_{h1} = N_{h,(k+1)/2} = N_{hk} = n_0 = 20$ and then use the OSD conditions to allocate 99 additional runs within each partition.

For the POSD method, we initialize as described in the previous section with $N_{h1} = N_{h,(k+1)/2} = N_{hk} = n_0 = 20$. We then used the algorithm described in Section 4.1 to allocate an additional 84 runs between each partition and within each partition.

5.2 Testing Framework

In this subsection, we present the four experiments we will conduct. The first experiment considers a function with three local minima on a domain with 60 design locations and compares the results of using POSD against the other methods described in subsection 5.1. The second experiment uses the same domain and underlying function as the first experiment but the simulation noises are not normally distributed. The last two experiments also use the same domain and underlying function as the first experiment but portions of the domain have much higher simulation noises than the rest of the domain. Based upon heuristics that exploit the adaptive nature of the first two cases of (B1) [4], we partitioned the domains of 60 design locations of the experiments into six disconnected partitions.

We conducted all four experiments using a total computing budget of 10,000 runs. The results will show that these amounts are sufficient to compare the performance of the methods and then determine the sensitivity of the POSD to the assumption of normally distributed noises. We repeat this whole procedure 10,000 times and then calculate the *PCS* obtained for each method after these 10,000 independent applications.

5.3 Experiment 1 (three local minima, 60 design locations)

This experiment is taken from the global optimization literature [22] and uses the following function: $f(x_i) = \sin(x_i) + \sin(10x_i/3) + \ln(x_i) - 0.84x_i + 3 + N(0, 1)$. We used a domain consisting of 60 evenly spaced design locations where $x \in [3, 8]$ such that the global minimum is $x_{27} \approx 5.20$ and $y(x_{27}) \approx -1.60$. This function also has two local minima at $x_6 \approx 3.42$ with $y(*x_6) \approx 0.16$ and $x_{47} \approx 7.07$ with $y(*x_{47}) \approx -1.27$.

As mentioned in subsection 5.2, we partitioned the domain for the regression based methods into six partitions and each of the local minimums are in a separate partition. Fig. 1 contains the simulation results. POSD clearly performs the best since it uses a regression equation to capture the information and then efficiently allocates both between and within the partitions.

The OSD and D-opt methods are the next best methods. They are regression-based methods that at least allocate efficiently within the partitions. As a point of comparison, OSD achieves a 95% *PCS* after about 2,200 runs and D-optimal achieves the same *PCS* after 3,300 runs. POSD achieves the same *PCS* after about 1,000 runs or about 45% of those required by OSD and 30% of those required by D-opt. EA-RS requires 5,700 runs to achieve a 95% *PCS* and the other two methods are even less competitive. After 10,000 runs, OCBA achieves an 83% *PCS* and EA only achieves a 49% *PCS*.

5.4 Experiment 2 (Different noise distributions)

This experiment uses the same underlying function and domain used in Experiment 1 $f(x_i) = \sin(x_i) + \sin(10x_i/3) + \ln(x_i) - 0.84x_i + 3$. We varied the type of the distribution for the noise terms of the simulation output while ensuring that each experiment used a distribution with a mean equal to zero and the variance is the same as that in Experiment 1. In addition to $\varepsilon_h \sim N(0, 1)$, we used:

- $\varepsilon_h \sim Uniform(\theta_1 = -\sqrt{3}, \theta_2 = \sqrt{3})$ where θ_1 and θ_2 are the lower and upper limits of the distribution,
- $\varepsilon_h \sim Exponential(\mu = 1) - 1$ where μ is the mean of the distribution, and
- $\varepsilon_h \sim Binomial(N = 2, p = 0.5) - 1$ where N is the number of trials and p is the probability of success.

The results of the experiment demonstrate that for this problem POSD is robust and performs relatively similar when assuming that the noise distribution terms are normally distributed even if the noise terms are actually from one of the other three distributions. Table 1 below provides a sample of the results.

5.5 Experiment 3 (High noise in non-best partitions)

This experiment again uses the same underlying function and domain used in Experiment 1. We varied the distribution for the noise terms of the simulation output such that $\varepsilon_h \sim N(0, 1)$ when $x_i \le x_{40}$; otherwise $\varepsilon_h \sim N(0, 10)$.

This distribution of the noise terms provides a much higher variance in the last two partitions, one of which includes the most competitive local minimum. The results are generally consistent with the first experiment and are shown in Fig. 2. D-optimal achieves a 90% *PCS* after about 4,500 runs and OSD achieves the same *PCS* after only 3,700 runs. POSD achieves a 90% *PCS* after about 1,370 runs or about 37% of those required by OSD and 30% of those required by D-opt. Notably different than the results from Experiment 1, OCBA is much more competitive with EA-RS and performs almost identically (and slightly better at times) until about 2,300 runs. The efficient allocation of OCBA competes well against the set of inefficient response surfaces generated by EA-RS. After 10,000 runs, EA-RS achieves an 89% *PCS*, OCBA achieves a 75% *PCS*, and EA only achieves a 41% *PCS*.

5.6 Experiment 4 (High noise, including the best partition)

This experiment is similar to Experiment 3 and uses the same underlying function and domain. We varied the distribution for the noise terms of the simulation output such that $\varepsilon_h \sim N(0, 1)$ when $x_{21} \le x_i \le x_{40}$; otherwise $\varepsilon_h \sim N(0, 10)$.

This distribution of the noise terms provides a much higher variance in the two middle partitions, one of which includes the global minimum. The results as shown in Fig. 3 are generally consistent with the first experiment and third experiment. D-optimal achieves a 70% *PCS* after about 9,200 runs and OSD achieves the same *PCS* after only 6,000 runs. POSD achieves a 70% *PCS* after about 2,400 runs or about 40% of those required by OSD and 26% of those required by D-opt. As with Experiment 3, OCBA is more competitive. The performance is almost identical with D-opt until about 1,500 runs and it outperforms EA-RS until about 4,700 runs. After 10,000 runs, EA-RS achieves an 89% *PCS*, OCBA achieves a 75% *PCS*, and EA only achieves a 41% *PCS*.

6 Conclusions

This paper explores the potential of further enhancing R&S efficiency by incorporating simulation information from across a partitioned domain into a regression based metamodel. We have developed a POSD method that can further enhance the efficiency of the simulation run allocation for selecting the best design. Our new method uses a heuristic based upon approximately optimal rules for between and within partitions that determine the number of samples allocated to each design location. Numerical experiments demonstrate that our new approach can dramatically enhance efficiency over existing efficient R&S methods.

Though the use of regression metamodels can dramatically enhance simulation efficiency, the regression-based methods are constrained with some typical assumptions such as an underlining quadratic function for the means and homogeneous simulation noise. As shown in our numerical experiments, these assumptions can be alleviated if we can efficiently partition the domain so that we focus only on a small local area of the domain where the assumptions will hold. The integration of the POSD method with intelligent search or partitioning algorithms for general simulation optimization problems is an ongoing research. If the function is highly nonlinear, more partitions may be needed in order to have a good fitting, which leads to more testing points. However, this issue can be alleviated if we have a smarter partition scheme. Similar to the idea of our optimal simulation allocation, we want to have more partitions near the optimal point in order to have good fitting. On the other hand, the quality of fitting can be significantly lowered if an area is much worse than a good one, i.e., it has a low chance to contain an optimal point. For such an area, we may require fewer partitions. Ideally, we want to have an ”optimal partition” scheme which can maximize the overall efficiency or the probability of correct selection. Other possible extensions include incorporating the work in Yang (2010) that extends the de la Garza phenomenon to other nonlinear forms such as exponential and log-linear models [23]. Yang’s effort provides the minimum number of support points and the optimal locations for some of the support points for these and other non-linear forms.

The dimensional curse is inherently a big challenge faced by any regression like procedures, including our approach. The focus of this paper is to enhance the efficiency of regression through a smarter computing budget allocation. We have shown that our proposed method offers a significant improvement over the well-known D-optimality approach in DOE literature (by 70~74% reduction). However, smarter computing budget allocation alone is not enough to tackle the dimensional issue. One promising approach we are taking as an ongoing research is to integrate our POSD method with some multi-dimensional search methods such as the stochastic trust region gradient-free method [7].

Acknowledgments

This work has been supported in part by NSF under Award CMMI-1233376, Department of Energy under Award DE-SC0002223, NIH under Grant 1R21DK088368-01, National Science Council of Taiwan under Award NSC-100-2218-E-002-027-MY3 and the Program for Professor of Special Appointment (Eastern Scholar) at Shanghai Institutions of Higher Learning.

References

1. Atkinson, AC.; Donev, AN. Optimum Experimental Designs. Oxford: Oxford Science Publications; 1998.
2. Bechhofer, RE.; Santner, TJ.; Goldsman, DM. Design and Analysis of Experiments for Statistical Selection, Screening, and Multiple Comparisons. New York: John Wiley & Sons; 1995.
3. Branke J, Chick S, Schmidt C. Selecting a selection procedure. Management Science. 2007; 53:1916–1932.
4. Brantley, MW. Simulation-based Stochastic Optimization on Discrete Domains: Integrating Optimal Computing and Response Surfaces. PhD thesis, George Mason University; 2011.
5. Brantley MW, Lee L-H, Chen C-H. Efficient simulation budget allocation with regression. IIE Transactions. 2013; 45:291–308.
6. Burden, F.; Faires, J. Numerical Analysis. 3rd edition. Boston, Massachusetts: PWS Publishers; 1993.
7. Chang K, Hong L, Wan H. Stochastic trust-region response-surface method (strong)-a new response-surface framework for simulation optimization. INFORMS Journal on Computing. 2013; 25:230–243.
8. Chen C-H, Yücesan E, Dai L, Chen H-C. Efficient computation of optimal budget allocation for discrete event simulation experiment. IIE Transactions. 2010; 42:60–70.
9. Chen C-H, He D, Fu M, Lee L-H. Efficient simulation budget allocation for selecting an optimal subset. Informs Journal on Computing. 2008; 20:579–595.
10. Chen, C-H.; Lee, L-H. Stochastic Simulation Optimization: An Optimal Computing Budget Allocation. Singapore: World Scientific Publishing Co.; 2011.
11. Chen C-H, Lin J, Yücesan E, Chick SE. Simulation budget allocation for further enhancing the efficiency of ordinal optimization. Journal of Discrete Event Dynamic Systems: Theory and Applications. 2000; 10:251–270.
12. Chick SE, Inoue K. New two-stage and sequential procedures for selecting the best simulated system. Operations Research. 2001; 49:1609–1624.
13. DeGroot, MH. Optimal Statistical Decisions. New York: McGraw Hill; 1970.
14. Friedman JH. Multivariate adaptive regression splines. The Annals of Statistics. 1991; 19:1–67.
15. Glynn PW, Juneja S. A large deviations perspective on ordinal optimization. Proceedings of the 2004 Winter Simulation Conference. 2004:577–585.
16. De la Garza A. Spacing of information in polynomial regression. The Annals of Mathematical Statistics. 1954; 25:123–130.

17. Lee L-H, Chen C-H, Chew E-P, Li J, Pujowidianto NA, Zhang S. A review of optimal computing budget allocation algorithms for simulation optimization problem. International Journal of Operations Research. 2010; 72:19–31.
18. Lee L-H, Pujowidianto A, Li L-W, Chen C-H, Yap C-M. Approximate simulation budget allocation for selecting the best design in the presence of stochastic constraints. IEEE Transactions on Automatic Control. 2012; 57:2940–2945.
19. Morrice DJ, Brantley MW, Chen C-H. An efficient ranking and selection procedure for a linear transient mean performance measure. Proceedings of the 2008 Winter Simulation Conference. 2008:290–296.
20. Rinott Y. On two-stage selection procedures and related probability inequalities. Communications in Statistics. 1978; A7:799–811.
21. Spall, JC. Introduction to Stochastic Search and Optimization: Estimation, Simulation, and Control. Hoboken, New Jersey: John Wiley & Sons; 2003.
22. Törn, A.; Žilinskas, A. Global optimization. In: Goos, G.; Hartmanis, J., editors. Lecture Notes in Computer Science. Vol. ume 350. Berlin: Springer-Verlag; 1989.
23. Yang M. On the de la garza phenomenon. Annals of Statistics. 2010; 38:2499–2524.

A Proof of Approximately Optimal Allocations

Proof of Theorem 2:We will examine the three cases that are delineated in (20a) and (20b). The first case in (20) is a general case where $P_{\mathrm{BM}} \geq P_{\Omega}$. The second case is a special case that addresses where $P_{\mathrm{BM}} < P_{\Omega}$ but there are no between partition comparisons in the lower and upper bounds of the *APCS* (such that $h \in \Psi \; \forall h$ and $\mathrm{M} = A,Z \; \forall h$). The third case is a general case where $P_{\mathrm{BM}} < P_{\Omega}$, except for the special case addressed by the second case. As such, our proof will establish the first and third cases and then address the special case.

<u>Case 1</u>: $P_{\mathrm{BM}} \geq P_{\Omega}$

When $P_{\mathrm{BM}} \geq P_{\Omega}$, we will not use the quadratic bound formulation for any of the non-best comparisons (such that $h \notin \Psi$ for every $n \neq B$). Therefore, our *APCS* from (14) simplifies to

$$APCS \geq 1 - P\{\tilde{y}(x_{Bb}) \geq \tilde{y}(x_{BA})\} - P\{\tilde{y}(x_{Bb}) \geq \tilde{y}(x_{BZ})\} - \sum_{h \notin \Psi} \sum_{i=1}^{k} p\{\tilde{y}(x_{Bb}) \geq \tilde{y}(x_{hi})\}.$$

To establish the upper bound, we show that $1 - \max [P\{\tilde{y}(x_{Bb}) \geq \tilde{y}(x_{BA})\},$

$$P\{\tilde{y}(x_{Bb}) \geq \tilde{y}(x_{BZ})\}] \geq 1 - P\{\tilde{y}(x_{Bb}) \geq \tilde{y}(x_{BA})\} - P\{\tilde{y}(x_{Bb}) \geq \tilde{y}(x_{BZ})\} - \sum_{h \notin \Psi} \sum_{i=1}^{k} P\{\tilde{y}(x_{Bb}) \geq \tilde{y}(x_{hi})\}$$
.

For the lower bound, given $P_{\mathrm{BM}} \geq P\{-\tilde{\delta}(x_{hi}) \geq 0\} \; \forall h \neq B$, $i = 1, \ldots, k$, we know that

$$\begin{aligned} 1 &- P\{\tilde{y}(x_{Bb}) \geq \tilde{y}(x_{BA})\} \\ &- P\{\tilde{y}(x_{Bb}) \geq \tilde{y}(x_{BZ})\} \\ &- \sum_{h \notin \Psi} \sum_{i=1}^{k} P\{\tilde{y}(x_{Bb}) \geq \tilde{y}(x_{hi})\} \geq 1 \\ &- P_{\mathrm{BM}} - P_{\mathrm{BM}} - \sum_{h \in \Psi} \sum_{i=1}^{k} P_{\mathrm{BM}} \end{aligned} .$$

Therefore, when $P_{\mathrm{BM}} \geq P_{\Omega}$, we can use the lower and upper bounds shown in equation (A. 1).

$$1-[(m-1)k+2]P_{\mathrm{BM}} \le APCS \le 1-P_{\mathrm{BM}}. \quad \text{(A.1)}$$

Since L and U only contain within comparisons for the best partition, we obtain $N_{B\cdot} = T$ and $N_{h\cdot} = 0$ for $h \neq B$. For the proof of the within allocation of α_{Bi} in accordance with the OSD conditions, see Lemma 4 in Appendix B.

Case 3: $P_{\mathrm{BM}} < P_{\Omega}$ (except for the special case addressed by Case 2 below)

When $P_{\mathrm{BM}} < P_{\Omega}$, the lower and upper bounds shown in (A2) can be established using a very similar approach as used for when $P_{\mathrm{BM}} \ge P_{\Omega}$.

$$1-2P_{\Omega\mathrm{M}}-3\sum_{h\in\Psi}P_{h\mathrm{M}}-k\sum_{h\notin\Psi}P_{h\mathrm{M}} \le APCS \le 1-\sum_{h\in\Psi}P_{h\mathrm{M}}-k\sum_{h\notin\Psi}P_{h\mathrm{M}}. \quad \text{(A.2)}$$

For the within partition allocations, see Lemma 4 in Appendix B for when $P_{h\mathrm{M}} = P\{-\tilde{d}(x_{h\mathrm{A}}) \ge 0\}$ or when $P_{h\mathrm{M}} = P\{-\tilde{d}(x_{h\mathrm{Z}}) \ge 0\}$. Lemma 5 in Appendix B addresses when $P_{h\mathrm{M}} = P\{-\tilde{\delta}(x_{hi}) \ge 0\}$ which occurs when $h \notin \Psi$ or when $h \in \Psi$ and $P_{h\mathrm{M}} = P\{-\tilde{\delta}(x_{hb}) \ge 0\}$. For the allocations within the best partition, except for the special case, see Lemma 6 in Appendix B. The proof for the between partition allocations closely follows those presented in [11,15]. See Lemma 7 and Lemma 8 in Appendix C.

Case 2: $P_{\mathrm{BM}} < P_{\Omega}$, $h \in \Psi$, $\forall h$ and M = A, Z, $\forall h$

For the special case when $P_{\mathrm{BM}} < P_{\Omega}$ but there are no between partition comparisons in the *APCS* (such that $h \in \Psi$, $\forall h$ and M = A,Z, $\forall h$), we also use the lower and upper bounds presented in (A.2). However, the allocations for the best partition are obtained using the same approach presented in Lemma 7 in Appendix C for the non-best partitions.

B Within Partition Allocations

Lemma 4 *When $P_{h\mathrm{M}} = P\{-\tilde{d}(x_{h\mathrm{A}}) \ge 0\}$ or when $P_{h\mathrm{M}} = P\{-\tilde{d}(x_{h\mathrm{Z}}) \ge 0\}$, the within partition comparisons are determined by the OSD conditions as expressed below in* (B.1).

$$x_{hs}=\begin{cases} x_{h(\mathrm{M}+b-1)}, & \frac{3x_{h1}+x_{hk}}{4} \le \frac{x_{h\mathrm{M}}+x_{hb}}{2} \le \frac{x_{h1}+x_{hk}}{2} \\ x_{h(\mathrm{M}+b-k)}, & \frac{x_{h1}+x_{hk}}{2} \le \frac{x_{h\mathrm{M}}+x_{hb}}{2} \le \frac{x_{h1}+3x_{hk}}{4} \\ x_{h(k-1)/2}, & otherwise \end{cases}, \quad \alpha_{hi}=\frac{|D_{h\mathrm{M},i}|}{|D_{h\mathrm{M},1}|+|D_{h\mathrm{M},s}|+|D_{h\mathrm{M},k}|}. \quad \text{(B.1)}$$

Proof: Setting $\partial U / \partial\alpha_{hj} = 0$, we obtain

$$\frac{D^2_{h\mathrm{A},1}}{\alpha^2_{h1}}=\frac{D^2_{h\mathrm{A},s}}{\alpha^2_{hs}}=\frac{D^2_{h\mathrm{A},k}}{\alpha^2_{hk}}=\frac{2\lambda N^2_{h\cdot}\zeta^{3/2}_{h\mathrm{A}}}{\sigma^2_h\hat{d}(x_{h\mathrm{A}})\phi(\hat{d}(x_{h\mathrm{A}})/\sqrt{\zeta_{h\mathrm{A}}})}.$$

Using the fact that $\alpha_{h1} + \alpha_{hs} + \alpha_{hk} = 1$, we obtain the result that

$$\alpha_{hi}=\frac{|D_{h\mathrm{M},i}|}{|D_{h\mathrm{M},1}|+|D_{h\mathrm{M},s}|+|D_{h\mathrm{M},k}|}.$$

For the proof of the optimal support point location, see [5]. The same results are obtained for the lower bound of the *APCS* when $P_{h\mathrm{M}} = P\{-d(\tilde{x}_{h\mathrm{A}}) \geq 0\}$ and for the case when $P_{h\mathrm{M}} = P\{-d(\tilde{x}_{h\mathrm{Z}}) \geq 0\}$.

Lemma 5 *When $P_{h\mathrm{M}} = P\{-\delta(\tilde{x}_{hi}) \geq 0\}$ and $h \neq B$, the within partition comparisons are determined by the c-optimality criterion where x_{hi} is selected as one of the three support points and* $\alpha_{hi} = 1.0$.

Proof: Setting $\partial U / \partial \alpha_{hj} = 0$, we obtain

$$\phi\left(\frac{\hat{\delta}(x_{hi})}{\sqrt{\xi_{hi}}}\right)\frac{\hat{\delta}(x_{hi})\sigma_h^2 E_{hi,j}^2}{2\xi_{hi}^{3/2} N_{h\cdot}^2 \alpha_{hj}^2}=\lambda. \quad \text{(B.2)}$$

Therefore,

$$\frac{E_{hi,1}^2}{\alpha_{h1}^2}=\frac{E_{hi,s}^2}{\alpha_{hs}^2}=\frac{E_{hi,k}^2}{\alpha_{hk}^2}=\frac{2\lambda N_{h\cdot}^2 \xi_{hi}^{3/2}}{\sigma_h^2 \hat{\delta}(x_{hi})\phi(\hat{\delta}(x_{hi})/\sqrt{\xi_{hi}})}.$$

Given a property of the Lagrange polynomial coefficients where $E_{hi,1} + E_{hi,s} + E_{hi,k} = 1$ (see [6]), we know that $E_{hi,j}^2 \neq 0$ for at least one of the support points. Using the symmetry of solutions and the fact that $\alpha_{h1} + \alpha_{hs} + \alpha_{hk} = 1$ and, assuming for example that $E_{hi,1}^2 \neq 0$, we obtain the general result that

$$\alpha_{hj}=\frac{|E_{hi,j}|}{|E_{hi,1}|+|E_{hi,s}|+|E_{hi,k}|}. \quad \text{(B.3)}$$

The same results are obtained for the lower bound of the *APCS* also. For the optimal support point location, we must consider three cases.

Case A-I: $x_{hi} = x_{h1}$. For this case, we obtain that $E_{hi,1} = 1$, $E_{hi,s} = 0$, $E_{hi,k} = 0$. Substituting these results into equation (B.3), we obtain $\alpha_{h1} = 1.0$.

Case A-II: $x_{hi} = x_{hk}$. For this case, we obtain that $E_{hi,1} = 0$, $E_{hi,s} = 0$, $E_{hi,k} = 1$ resulting in $\alpha_{hk} = 1.0$.

Case A-III: $x_{hi} \neq x_{h1}$ and $x_{hi} \neq x_{hk}$. From (17), when $x_{hs} = x_{hi}$, we obtain that $E_{hi,1} = 0$, $E_{hi,s} = 1$, $E_{hi,k} = 0$. Substituting these results into (B.3), we obtain $\alpha_{hs} = 1.0$. In order to show that $x_{hs} = x_{hi}$ is an optimal selection of x_{hs}, we can use the chain rule to establish that

$$\frac{\partial U}{\partial x_{hs}}=-\frac{\partial U}{\partial \xi_{hi}}\cdot\frac{\partial \xi_{hi}}{\partial x_{hs}}=-\phi(\frac{\hat{\delta}(x_{hi})}{\sqrt{\xi_{hi}}})\frac{\hat{\delta}(x_{hi})}{2\xi_{hi}^{3/2}}\cdot\frac{\partial \xi_{hi}}{\partial x_{hs}}.$$

Substituting (B.3) into (17), we obtain

$$\xi_{hi}=\frac{\sigma_h^2}{N_{h\cdot}}[|E_{hi,1}|+|E_{hi,s}|+|E_{hi,k}|]^2+\frac{\sigma_B^2}{N_{B\cdot}}\left[\frac{E_{Bi,1}^2}{\alpha_{B1}}+\frac{E_{Bi,s}^2}{\alpha_{Bs}}+\frac{E_{Bk,k}^2}{\alpha_{Bk}}\right]. \quad \text{(B.4)}$$

Using again the property where $E_{hi,1} + E_{hi,s} + E_{hi,k} = 1$ (see [6]), we know that $|E_{hi,1}| + |E_{hi,s}| + |E_{hi,k}| \geq 1$. Thus, when $x_{hs} > x_{hi}$, $\partial\xi_{hi} / \partial x_{hs} \geq 0$ such that $\partial U / \partial x_{hs} \leq 0$. Similarly, when $x_{hs} < x_{hi}$, $\partial\xi_{hi} / \partial x_{hs} \leq 0$ such that $\partial U / \partial x_{hs} \geq 0$.

Lemma 6 *When $P_{BM} < P_{\Omega M}$, the within partition comparisons for the best partition are determined by the c-optimality criterion where x_{Bb} is selected as one of the three support points and $\alpha_{Bb} = 1.0$ (for future allocations after initial runs so that we do not have a singular solution).*

Proof: When $P_{BM} < P_{\Omega M}$, setting $\partial U / \partial\alpha_{Bj} = 0$, we obtain $E_{Bb,j}^2/\alpha_{Bj}^2=(N_{B\cdot}^2\lambda)/(\sigma_B^2\Xi)$, where

$$\Xi=\sum_{h\notin\Psi}\phi(\frac{\hat{\delta}(x_{hM})}{\sqrt{\xi_{hM}}})\frac{\hat{\delta}(x_{hM})}{2\xi_{hM}^{3/2}}+\sum_{h\in\Psi,\mathrm{M}=b}\phi(\frac{\hat{\delta}(x_{hb})}{\sqrt{\xi_{hb}}})\frac{\hat{\delta}(x_{hb})}{2\xi_{hb}^{3/2}}.$$

The rest of the proof follows the proof from Lemma 5.

C Between Partition Allocations

Lemma 7 *When $P_{BM} < P_{\Omega M}$, the between partition allocations for $i \neq B$ and $j \neq B$ are obtained by* $\frac{N_{i\cdot}}{N_{j\cdot}}=\frac{R_j}{R_i}$.

Proof: We consider three cases.

<u>Case B-I:</u>

When $P_{hM} = P\{-d(\tilde{x}_{hA}) \geq 0\}$ or when $P_{hM} = P\{-d(\tilde{x}_{hZ}) \geq 0\}$ and $h \neq B$ for both comparisons, setting $\partial U / \partial N_{h\cdot} = 0$, we obtain

$$\phi(\frac{\hat{d}(x_{hM})}{\sqrt{\zeta_{hM}}})\frac{\hat{d}(x_{hM})\sigma_h^2}{2\zeta_{hM}^{3/2}N_{h\cdot}^2}\left[\frac{D_{hM,1}^2}{\alpha_{h1}}+\frac{D_{hM,s}^2}{\alpha_{hs}}+\frac{D_{hM,k}^2}{\alpha_{hk}}\right]=\lambda.$$

Chen et al. (2000) [11] provide using an asymptotic allocation rule where $T \to \infty$ and Glynn and Juneja (2004) [15] provide using a large deviation approach that $\hat{d^2}(x_{iM})/\zeta_{iM} = \hat{d^2}(x_{jM})/\zeta_{jM}$. Substituting the results from Lemma 4, we know that

$$\zeta_{hi} = \frac{\sigma_h^2[|D_{hM,1}|+|D_{hM,s}|+|D_{hM,k}|]^2}{N_{h\cdot}}.$$

We can then show

$$\frac{N_{i\cdot}}{N_{j\cdot}} = \frac{\sigma_i^2[|D_{iM,1}|+|D_{iM,s}|+|D_{iM,k}|]^2\hat{d}^2(x_{jM})}{\sigma_j^2[|D_{jM,1}|+|D_{jM,s}|+|D_{jM,k}|]^2\hat{d}^2(x_{iM})}.$$

Note that these results are very similar to the OCBA results with the major difference being that the Lagrange coefficients serve as an efficiency factor for the within partition allocations.

<u>Case B-II</u>:

When $P_{hM} = P\{-\tilde{\delta}(x_{hM}) \geq 0\}$ and $h \neq B$ for both comparisons,

$$\frac{\partial U}{\partial N_{h\cdot}} = \phi\left(\frac{\hat{\delta}(x_{hM})}{\sqrt{\xi_{hM}}}\right)\frac{\hat{\delta}(x_{hM})\sigma_h^2}{2\xi_{hM}^{3/2}N_{h\cdot}^2}\left[\frac{E_{hM,1}^2}{\alpha_{h1}}+\frac{E_{hM,s}^2}{\alpha_{hs}}+\frac{E_{hM,k}^2}{\alpha_{hk}}\right]-\lambda.$$

Substituting the results from Lemma 5,

$$\xi_{hb} = \frac{\sigma_h^2}{N_{h\cdot}}+\frac{\sigma_B^2}{N_{B\cdot}}+\left[\frac{E_{Bb,1}^2}{\alpha_{B1}}+\frac{E_{Bb,s}^2}{\alpha_{Bs}}+\frac{E_{Bb,k}^2}{\alpha_{Bk}}\right].$$

Using the assumption that $N_{B\cdot} \gg N_{h\cdot}$ [11,15]. $\varepsilon_{hb} \approx sigma_h^2/N_{h\cdot}$. The rest of the proof follows Case B-I such that

$$\frac{N_{i\cdot}}{N_{j\cdot}} = \frac{\sigma_i^2}{\sigma_i^2}\cdot\frac{\hat{\delta}^2(x_{jb})}{\hat{\delta}^2(x_{ib})}.$$

<u>Case B-III</u>:

When $P_{hM} = P\{-d(\tilde{x}_{hA}) \geq 0\}$ or when $P_{hM} = P\{-d(\tilde{x}_{hZ}) \geq 0\}$ and $h \neq B$ for one comparison and $P_{hM} = P\{-\tilde{\delta}(x_{hM}) \geq 0\}$ and $h \neq B$ for the other comparison, this case follows from the results for Case B-I and Case B-II such that

$$\frac{N_{i\cdot}}{N_{j\cdot}} = \frac{\sigma_i^2[|D_{iM,1}|+|D_{iM,s}|+|D_{iM,k}|]^2}{\sigma_i^2}\cdot\frac{\hat{\delta}^2(x_{jb})}{\hat{d}^2(x_{iM})}.$$

Lemma 8 *When $P_{\mathrm{BM}} < P_{\Omega\mathrm{M}}$, the between partition allocations for $h = B$ are obtained by*

$$\frac{N_{B\cdot}^2}{\sigma_B^2} = \sum_{h\in\Psi,\mathrm{M}=b}^{m} \frac{N_{h\cdot}^2}{\sigma_h^2} + \sum_{h\notin\Psi}^{m} \frac{N_{h\cdot}^2}{\sigma_h^2}.$$

Proof: This proof closely follows the one provided in [11]. Setting $\partial U / \partial N_{B\cdot} = 0$, we obtain

$$\frac{N_{B\cdot}^2}{N_B^2}\lambda = \left[\frac{E_{Bb,1}^2}{\alpha_{B1}} + \frac{E_{Bb,s}^2}{\alpha_{Bs}} + \frac{E_{Bb,k}^2}{\alpha_{Bk}}\right]\left(\sum_{h\notin\Psi}^{m}\phi\left(\frac{\hat{\delta}(x_{hM})}{\sqrt{\xi_{hM}}}\right)\frac{\hat{\delta}(x_{hM})}{2\xi_{hM}^{3/2}} + \sum_{h\in\Psi,\mathrm{M}=b}^{m}\phi\left(\frac{\hat{\delta}(x_{hb})}{\sqrt{\xi_{hb}}}\right)\frac{\hat{\delta}(x_{hb})}{2\xi_{hb}^{3/2}}\right).$$

Using the results from Lemma 5 and (B.2), it can be shown that for $h \neq B$

$$\phi\left(\frac{\hat{\delta}(x_{hM})}{\sqrt{\xi_{hM}}}\right)\frac{\hat{\delta}(x_{hM})}{2\xi_{hM}^{3/2}} = \lambda\frac{N_{h\cdot}^2}{\sigma_h^2}.$$

Using the results from Lemma 6, we know that

$$\left[\frac{E_{Bb,1}^2}{\alpha_{B1}} + \frac{E_{Bb,s}^2}{\alpha_{Bs}} + \frac{E_{Bb,k}^2}{\alpha_{Bk}}\right] = 1.$$

Substituting these results, we obtain

$$\frac{N_{B\cdot}^2}{\sigma_B^2} = \sum_{h\in\Psi,\mathrm{M}=b}^{m} \frac{N_{h\cdot}^2}{\sigma_h^2} + \sum_{h\notin\Psi}^{m} \frac{N_{h\cdot}^2}{\sigma_h^2}.$$

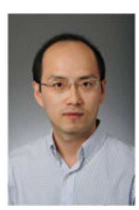
Biographies

Mark W. Brantley is the managing member of Goose Point Analysis L.L.C. He received his Ph.D. from George Mason University. His research interests include efficient methodologies for simulation and its applications.

Loo Hay Lee is an Associate Professor and Deputy Head (Research) in the Department of Industrial and Systems Engineering, National University of Singapore. He received his B.S.

(Electrical Engineering) degree from the National Taiwan University in 1992 and his S. M. and Ph.D. degrees in 1994 and 1997 from Harvard University. He is currently a senior member of IEEE, a member of INFORMS and ORSS. His research interests include Simulation-based Optimization, Maritime Logistics and Port Operations and Supply Chain Modeling He has received the best paper award at the INFOCOMP 2011, and the grand prize for the next generation container port challenge in 2013 by proposing a revolutionary double storey container terminal design, called SINGA Port. Dr Lee has also been appointed as the Eastern Scholar Professor for the Shanghai Maritime University by the Shanghai Municipal Education Commission. He has served as the associate editor for IEEE Transactions on Automatic Control, IEEE Transactions on Automation Science and Engineering, IIE Transactions, Flexible Services and Manufacturing Journal, the Asia Pacific Journal of Operational Research and Simulation: Transactions of The Society for Modeling and Simulation International. He is also the co-editor for Journal of Simulation and is a member in the advisory board for OR Spectrum.

Chun-Hung Chen received his Ph.D. degree in Engineering Sciences from Harvard University in 1994. He is a Professor of Systems Engineering & Operations Research at George Mason University and is also affiliated with National Taiwan University. Dr. Chen was an Assistant Professor of Systems Engineering at the University of Pennsylvania before joining GMU. Sponsored by NSF, NIH, DOE, NASA, MDA, and FAA, he has worked on the development of highly efficient methodology towards stochastic simulation optimization and its applications to air transportation system, semiconductor manufacturing, healthcare, security network, power grids, and missile defense system. Dr. Chen received “National Thousand Talents” Award from the central government of China in 2011, the Best Automation Paper Award from the 2003 IEEE International Conference on Robotics and Automation, 1994 Eliahu I. Jury Award from Harvard University, and the 1992 MasPar Parallel Computer Challenge Award. Dr. Chen has served as Co-Editor of the Proceedings of the 2002 Winter Simulation Conference and Program Co-Chair for 2007 Informs Simulation Society Workshop. He has served as a department editor for IIE Transactions, associate editor of IEEE Transactions on Automatic Control, area editor of Journal of Simulation Modeling Practice and Theory, associate editor of International Journal of Simulation and Process Modeling, and associate editor of IEEE Transactions on Automation Science and Engineering.



Jie Xu is Assistant Professor of Systems Engineering & Operations Research at George Mason University. His research interests include stochastic optimization via simulation, input modeling, risk analysis, and computational intelligence. He received his B.S. degree in Electronics & Information Systems from Nanjing University in 1999, his M.E. degree in Communications & Information Systems from Shanghai Jiaotong University in 2002, his M.S. degree in Computer Science from SUNY-Buffalo in 2004, and his Ph.D. degree in Industrial Engineering & Management Sciences from Northwestern University in 2009.

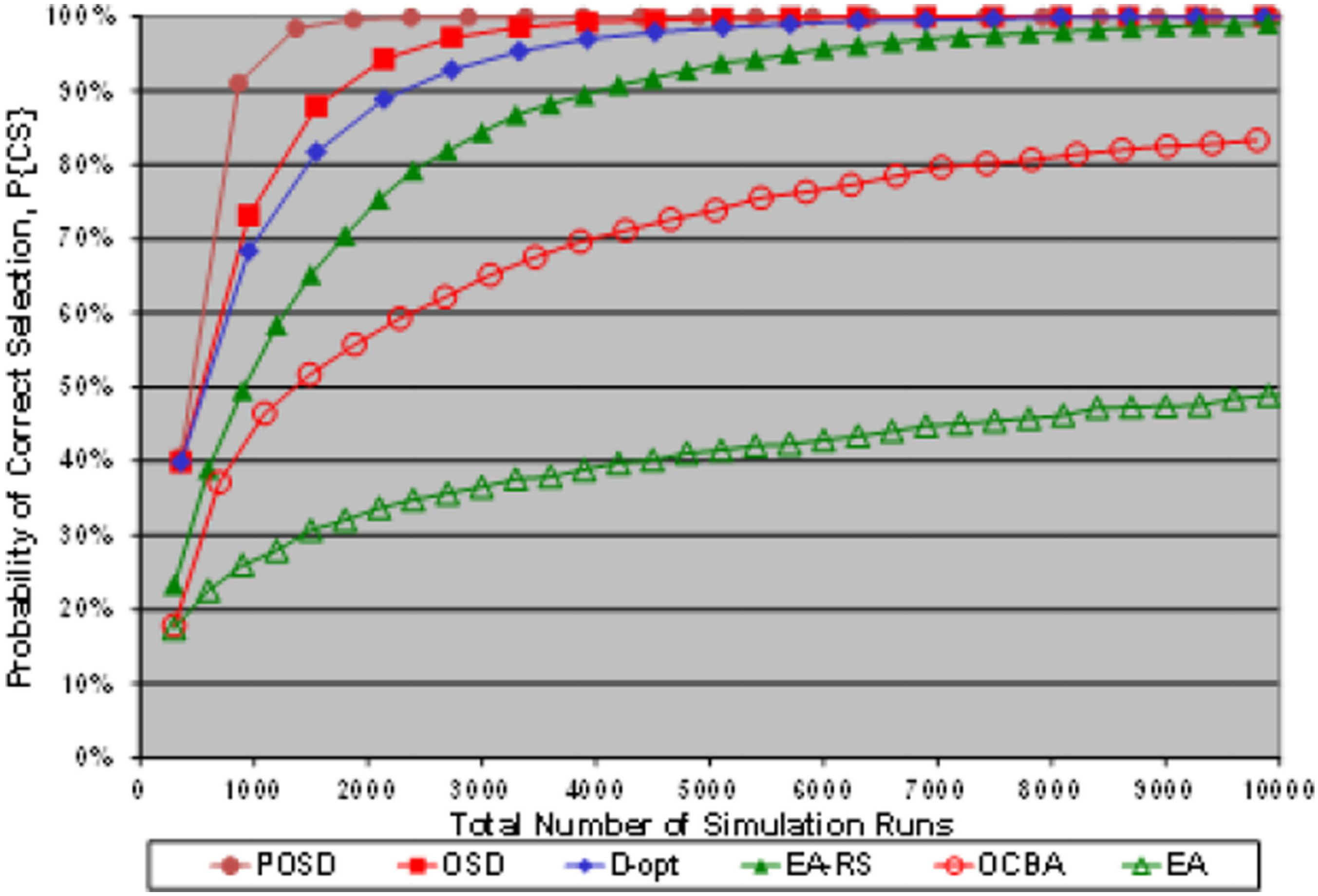


Fig. 1.
Results of Experiment 1

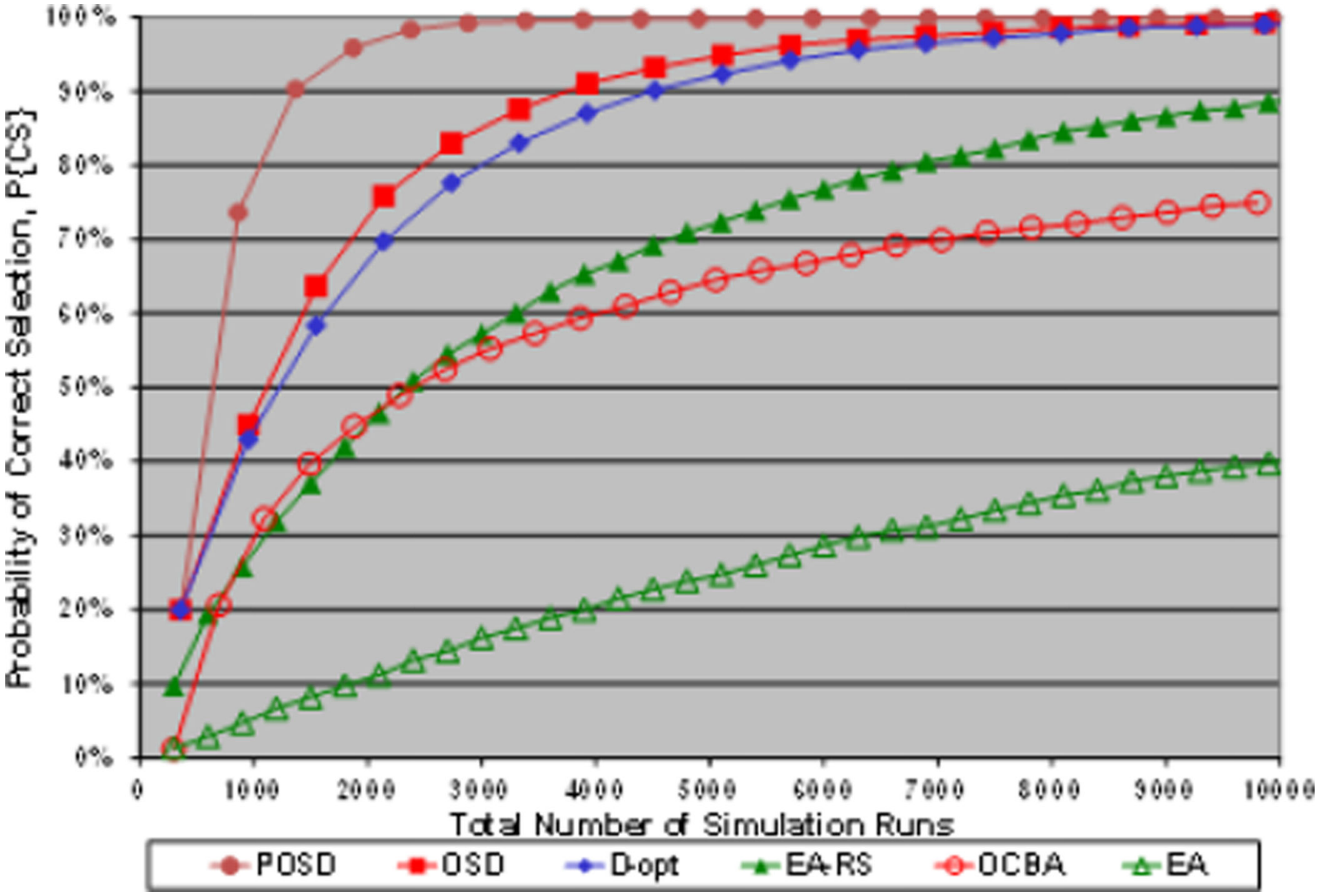


Fig. 2.
Results of Experiment 3

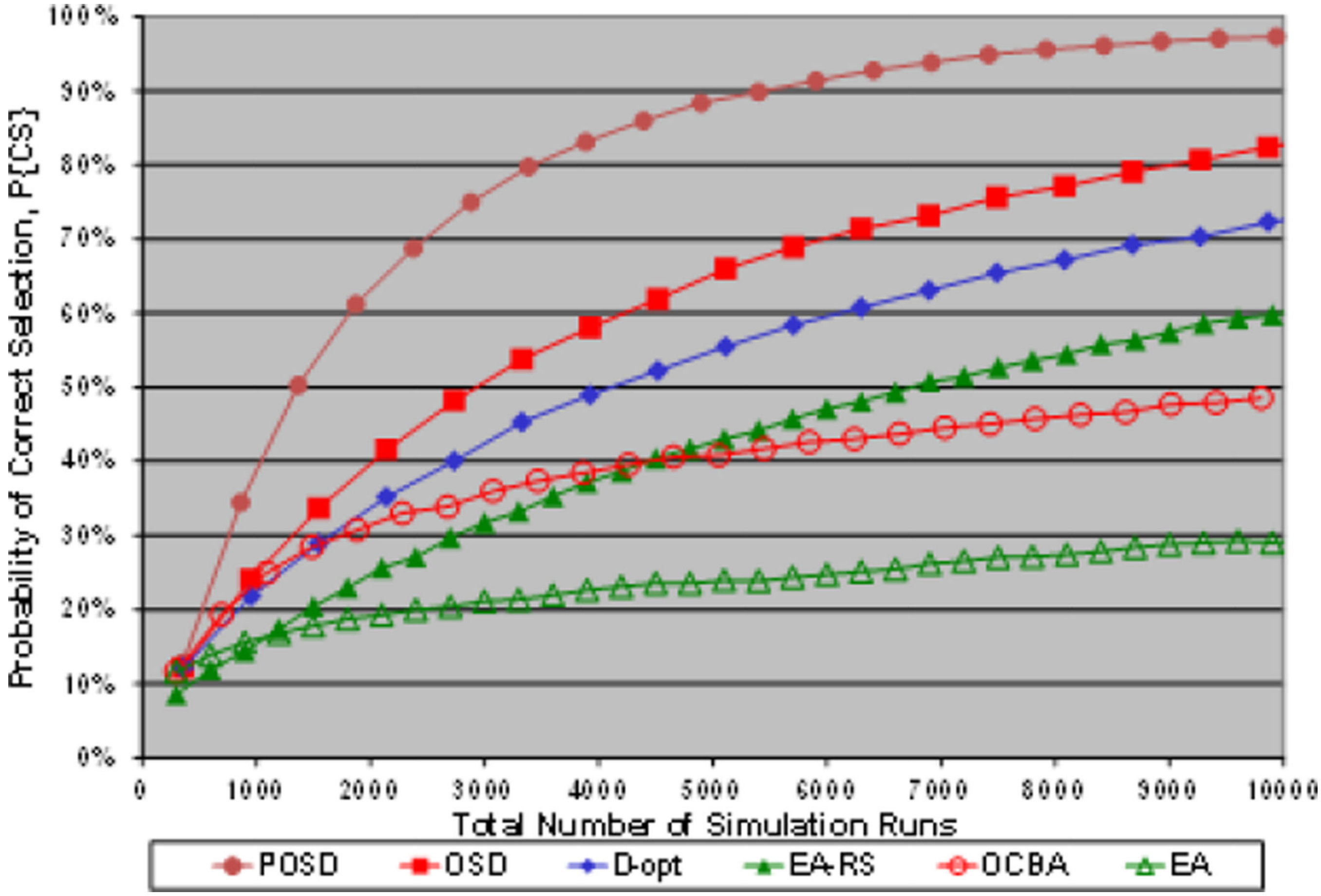


Fig. 3.
Results of Experiment 4

Table 1

Results from Experiment 2

Total Runs	***PCS* (Normal)**	***PCS* (Uniform)**	***PCS* (Exp)**	***PCS* (Bin)**
528	69.84%	70.06%	71.14%	70.70%
1536	99.14%	99.09%	98.97%	99.19%
2544	99.89%	99.93%	99.72%	99.88%