# Reusing Search Data in Ranking and Selection: What Could Possibly Go Wrong?

David Eckman and Shane Henderson
School of Operations Research and Information Engineering
Cornell University
Ithaca, New York 14853

November 18, 2016

### Abstract

It is tempting to reuse replications taken during a simulation optimization search as input to a ranking-and-selection procedure. However, even when the random inputs used to generate replications are i.i.d. and independent across systems, we show that for searches that use the observed performance of explored systems to identify new systems, the replications are conditionally dependent given the sequence of returned systems. Through simulation experiments, we demonstrate that reusing the replications taken during search in selection and subset-selection procedures can result in probabilities of correct and good selection well below the guaranteed levels. Based on these negative findings, we call into question the guarantees of established ranking-and-selection procedures that reuse search data. We also rigorously define guarantees for ranking-and-selection procedures after search and discuss how procedures that only provide guarantees in the preference zone are ill-suited to this setting.

## 1 Introduction

Real-world simulation optimization problems often feature a large, possibly uncountable, number of systems and a limited computational budget with which to evaluate them. To have any hope of tackling such problems, practitioners must restrict their attention to evaluating only a subset of systems. In the absence of structural information about the objective function, candidate systems are usually identified via sampling or search.

Whereas sampling methods, e.g. random sampling, Latin hypercube sampling, and orthogonal designs, determine a set of candidate systems without regard to observed performance, search methods move from system to system, taking replications and using the observed performance of explored systems to identify the next system. As a result, search methods are expected to return better systems than sampling. Search methods used in simulation optimization problems range from the naïve, e.g. random search, to more advanced methods combined with optimization, e.g. stochastic approximation, sample average approximation; see, for example, Chapters 18, 20, and 21 of [12] and Chapters 10, 11, and 12 of [9] for more discussion.

As pointed out by [5], heuristic search methods, e.g. simulated annealing, Nelder-Mead, and tabu search, do not typically guarantee convergence to the globally optimal system and procedures with guarantees of converging to a local optimum often require the simulation budget to approach infinity. In either case, no finite-time guarantee is made with respect to identifying the best system among those explored. To obtain a statistical guarantee of this kind, ranking-and-selection procedures can be applied on the set of candidate systems [5].

Ranking-and-selection procedures specify how to take sufficient replications of each system to provide a statistical guarantee with respect to two objectives: *selection* and *subset selection*. Selection procedures provide a guarantee on the probability of selecting the best system—in terms of performance—of a set of systems. This guarantee is often conditional on the performance of the best system being at least $\delta$ better than the second best, where $\delta > 0$ is an indifference-zone parameter specified by the decision-maker. When this condition is not satisfied, some selection procedures still provide a guarantee on the probability of selecting a good system—one whose performance is strictly within $\delta$ of the best. On the other hand, subset-selection procedures provide a guarantee on the probability that a returned subset includes the best system.

[4, 5] examine how ranking-and-selection procedures can be applied on a database of explored systems to "clean up" *after* search. [4] outline how simulation optimization software can be configured to permit ranking and selection after search. [5] further elucidate selection and subset-selection procedures that reuse replications taken during a heuristic search. The procedures give guarantees on the probability of correct selection for instances in which the configuration of the set of returned systems is in the preference zone, i.e., the best system is at least $\delta$ better than the others.

It is enticing to reuse the replications gathered during a search as input to a ranking-and-selection procedure, especially when generating simulation replications is computationally expensive. Nevertheless, search data violates one of the key assumptions of most ranking-and-selection procedures: that replications are independent across systems. More specifically, since the identities of new systems depend on the observed performance of previously explored systems, the replications taken during search are *conditionally dependent given the sequence of returned systems*. This conditional dependence causes issues when attempting to make guarantees conditional on the output of a search. Through simulation experiments, we demonstrate how this dependence can undermine the guarantees of ranking-and-selection procedures that reuse search data.

Similar issues will arise for other applications of ranking-and-selection procedures in which search data is reused. For instance, ranking-and-selection procedures have been embedded within various search methods to provide some degree of statistical control at each iteration of the search; examples include nested partitions [20], simulated annealing [1], and pattern search [25], all of which do not reuse past replications. In this setting, a notable ranking-and-selection procedure that reuses previously collected replications is Sequential Selection with Memory (SSM) [23]. SSM provides a guarantee on the probability of selecting the best candidate system in a given search iteration. SSM has been applied within random search [23] and nested partitions for discrete-variable optimization [22]. See [21] for an earlier presentation of SSM.

[13] address a related setting in which systems are revealed in batches by a generic "system-generating algorithm" that can be considered a search. In this framework, selection decisions are made at each iteration and early selection decisions are intended to direct the search in identifying new systems. In making each selection decision, the proposed procedures reuse the past replications of systems returned in earlier batches. These procedures provide guarantees over the selection decisions at each search iteration as well as an overall guarantee on the system ultimately selected.

In this paper, we focus on ranking-and-selection procedures designed for normally-distributed replications. Although this assumption appears restrictive, it is often approximately satisfied when performance measures are the result of batching individual replications into macroreplications and appealing to the Central Limit Theorem. When this normality assumption is not fully justified, other procedures might be used, e.g. [14] and [10]. A related class of examples arises in the literature of multi-armed bandits in pure exploration, wherein the objective is to identify the best system in finite time. Existing procedures provide probably approximately correct (PAC) guarantees, i.e., a system within $\delta$ of the best is selected with probability above a prespecified level [7]. The PAC guarantees of these selection procedures could also be negatively affected by reusing search data.

The remainder of the paper is outlined as follows. In Section 2, we rigorously define the types of guarantees that arise from applying ranking and selection after sampling or search and discuss conditions under which these guarantees hold. In the process, we explain why guarantees based on the indifference-zone formulation are unsatisfactory. In Section 3, we present a search-like method, called Adversarial Search, in which an adversary introduces new systems in a way that exploits the conditional dependence of search data. In Section 4, we test several selection and subset-selection procedures that reuse the replications taken by Adversarial Search. We also test a subset-selection procedure that reuses data from a more realistic search. In Section 5 we summarize our findings and outline open research questions.

## 2 Ranking and Selection after Sampling or Search

We consider the optimization problem $\max_{x \in \Theta} g(x)$ where $\Theta$ is a space of systems, presumably large, and $g : \Theta \mapsto \mathbb{R}$ is an objective function. Here $x$ represents a vector of decision variables, e.g. simulation parameters, and is henceforth referred to as a *system*. In keeping with convention, we use $x$ when the identity of a system is fixed and $X$ when it is a random variable. At a given system $x$, the objective function can only be estimated from noisy replications $Y_j(x)$, $j = 1, 2, \ldots$, satisfying $g(x) = \mathbb{E}[Y_j(x)]$. We further assume that the replications are of the form $Y_j(x) = g(x) + \sigma(x)\xi_j$ where $\sigma(x)$ represents the variability at system $x$ and the random error terms $\xi_j \sim \mathcal{N}(0, 1)$ are i.i.d. In this paper, we consider only ranking-and-selection procedures for which the random error terms $\xi_j$ are independent across systems. In doing so, we leave unanswered the question of whether our results extend to ranking-and-selection procedures that make use of common random numbers, e.g. [18].

## 2.1 Sampling vs. Search

We use the term *sampling* to describe methods that choose systems $X_1, \ldots, X_k \in \Theta$ without the need to take any replications, where $k$ is specified in advance by the decision-maker. More precisely, sampling methods are of the form $X_i \in m\mathcal{F}_{i-1}$ ($X_i$ is measurable w.r.t. $\mathcal{F}_{i-1}$) for $i = 1, \ldots, k$, where $\mathcal{F}_{i-1} := \sigma\{X_1, \ldots, X_{i-1}, U_0, \ldots U_{i-1}\}$, is the $\sigma$-field generated by all information obtained just prior to the identification of system $X_i$ and $U_{i-1}$ represents an (optional) random input associated with identifying system $X_i$. In other words, each system returned by a sampling method is a deterministic function of the sequence of previously returned systems and the random inputs used to identify them.

In contrast, we define *search* methods to be those that on the $i$th iteration take $n_{0i}$ replications of a system $X_i$ and identify the $(i+1)$st system based on these replications and those of previously explored systems. More precisely, let $\mathbf{Y}(X_i) := [Y_1(X_i), \ldots, Y_{n_{0i}}(X_i)]^T$ be the vector of $n_{0i}$ replications of system $X_i$ and let $\bar{Y}(X_i) := (n_{0i})^{-1} \sum_{j=1}^{n_{0i}} Y_j(X_i)$ denote the sample mean of these replications; thus $\bar{Y}(X_i)$ is a conditionally unbiased estimator of $g(X_i)$ conditioned on $X_i$. Then search methods are of the form $X_i \in m\mathcal{G}_{i-1}$ ($X_i$ is measureable w.r.t. $\mathcal{G}_{i-1}$) for $i = 1, \ldots, k$ where $\mathcal{G}_{i-1} := \sigma\{X_1, \ldots, X_{i-1}, \mathbf{Y}(X_1), \ldots, \mathbf{Y}(X_{i-1}), U_0, \ldots, U_{i-1}\}$. By containing all of the search data $\mathbf{Y}(X_1), \ldots, \mathbf{Y}(X_{i-1})$, the $\sigma$-field $\mathcal{G}_{i-1}$ is very general; all of the search methods we consider use only the sample means $\bar{Y}(X_1), \ldots, \bar{Y}(X_{i-1})$.

By using the replications $\mathbf{Y}(X_1), \ldots, \mathbf{Y}(X_{i-1})$ to choose system $X_i$, search methods may return better systems than sampling methods. However, this improvement in the quality of returned systems comes at a cost: the replications taken during search are dependent across systems. In particular, because the identity of a new system is dependent on the observed performance—and not just the identities—of previously explored systems, *the search data are conditionally dependent given the sequence of returned systems.*

To help illustrate this dependence, consider a small example in which only two systems $X_1$ and $X_2$ are returned by a search. Their replications are

$$Y_j(X_1) = g(X_1) + \sigma(X_1)\xi_{1j} \quad \text{for } j = 1, \ldots, n_{01},$$

$$Y_l(X_2) = g(X_2) + \sigma(X_2)\xi_{2l} \quad \text{for } l = 1, \ldots, n_{02},$$

where $\xi_{ij}$ denotes the random error term of the $j$th replication of the $i$th returned system. Although the random error terms $\xi_{1j}$ and $\xi_{2l}$ are independent for all $j$ and $l$, the replications $Y_j(X_1)$ and $Y_l(X_2)$ are dependent because the identity of the second system, $X_2$, depends—through the search—on the identity of the first system $X_1$ and its replications $\mathbf{Y}(X_1)$. Furthermore, when we condition on the identities of the returned systems, i.e., the event $\{X_1 = x_1, X_2 = x_2\}$, the fact that $X_2 = x_2$ provides information about $\mathbf{Y}(x_1)$ and therefore about the replications $Y_j(X_1)$ and their error terms $\xi_{1j}$. Because of this information about the observed performance of system $X_1$, the replications $Y_j(X_1)$, $j = 1, \ldots, n_{01}$ are conditionally dependent given the sequence of the returned systems.

In the preceding argument, by comparing the replications of systems $X_1$ and $X_2$ we implicitly assumed knowledge of the order in which the systems were returned by the search; in our notation the subscript indices indicate that system $X_1$ was returned before system $X_2$. While this information is often available in practice, we argue that knowledge of the order of returned systems is not necessary for search data to be conditionally dependent.

That is, if the returned systems were randomly permuted after the search, so that their initial ordering was no longer known, the search data would still be dependent conditioned only on the identities of the returned systems. As justification for this claim, all of the ranking-and-selection procedures tested in this paper—and almost all ranking-and-selection procedures in general—are invariant under permutation, i.e., they make the same selection decisions regardless of the order in which the systems are labeled. Our experimental results for these procedures when reusing search data (see Section 4) show lower probabilities of correct and good selection compared to the case of independent data. These results suggest that search data can be conditionally dependent given the identities of the returned systems. We will discuss in Section 2.4 how this conditional dependence can affect the guarantees of ranking-and-selection procedures that reuse search data.

## 2.2 Guarantees of Ranking-and-Selection Procedures

We introduce notation necessary for describing aspects of ranking-and-selection procedures. Let $\mathcal{X} := \{x_1, \ldots, x_k\} \subseteq \Theta$ denote a fixed set of systems and define the vectors $\mu := [g(x_1), \ldots, g(x_k)]^T$ and $\Sigma := [\sigma^2(x_1), \ldots, \sigma^2(x_k)]^T$. Thus $\mu$ and $\Sigma$ are the means and variances, respectively, of the normal distributions from which replications of systems $x_1, \ldots, x_k$ are drawn. Let $x_{[i]}$ denote the $i$th system when the systems are ordered by the objective function $g$, i.e., $g(x_{[1]}) \leq g(x_{[2]}) \leq \cdots \leq g(x_{[k]})$.

Ranking-and-selection procedures can be classified into two types: selection and subset-selection. Selection procedures ultimately select a single system, denoted $x_K$, that is believed to be the best while subset-selection procedures return a subset of systems $I \subseteq \mathcal{X}$ that is believed to contain the best system. Ranking-and-selection procedures are concerned with the events of correct selection (CS), i.e., selecting or preserving the best system, and good selection (GS), i.e., selecting or preserving a system whose performance is strictly within $\delta$ of the best. For selection procedures, correct and good selection are defined as $\text{CS} := \{\mu(x_K) = \mu(x_{[k]})\}$ and $\text{GS} := \{\mu(x_K) \geq \mu(x_{[k]}) - \delta\}$. For subset-selection procedures, correct and good selection are likewise defined as $\text{CS} := \{x_{[k]} \in I\}$ and $\text{GS} := \{\exists x \in I \text{ s.t. } g(x) > g(x_{[k]}) - \delta\}$.

In the literature, $\mu$ is referred to as the *configuration* of the systems. The preference zone is defined as $\text{PZ}(\delta) := \{\mu : g(x_{[k]}) - g(x_{[k-1]}) \geq \delta\}$, the set of configurations in which the best system is at least $\delta$ better than the second best. Similarly, the indifference zone $\text{IZ}(\delta)$ is defined to be the complement of the preference zone. Both zones are parameterized by a scalar $\delta > 0$, whose value is chosen to reflect the largest difference in performance to which the decision-maker is indifferent. Thus for configurations in the preference zone, the decision-maker strongly prefers the best system to the other systems.

Ranking-and-selection procedures often give guarantees on the probability of correct selection (PCS) based on the indifference-zone formulation:

$$\mathbb{P}_{(\mu,\Sigma)}(\text{CS}) \geq 1 - \alpha \quad \text{for all } \mu \in \text{PZ}(\delta), \tag{1}$$

for $1/k < 1 - \alpha < 1$ where $\mathbb{P}_{(\mu,\Sigma)}$ is the probability measure with respect to the normal distributions specified by the elements of $\mu$ and $\Sigma$. That is, for any configuration in the preference zone, the ranking-and-selection procedure guarantees a lower bound on PCS. Some procedures also give guarantees on the probability of good selection (PGS), regardless of whether the configuration is in the preference zone:

$$\mathbb{P}_{(\mu,\Sigma)}(\text{GS}) \geq 1 - \alpha \quad \text{for all } \mu. \tag{2}$$

Guarantee (2) implies Guarantee (1) because in PZ($\delta$) the only good system is the best system.

## 2.3 PGS and PCS Guarantees after Sampling or Search

In the ranking-and-selection literature, the set of systems under consideration is usually fixed in advance, as in Section 2.2. We consider extensions of Guarantees (1) and (2) to instances in which the set of systems $\mathcal{X}$—and hence the configuration $\mu$—is not fixed, but is instead random, namely, the output of a sampling or search method. We first ask the motivating question: what are meaningful guarantees on PGS and PCS when a ranking-and-selection procedure $\mathcal{R}$ is run after an arbitrary sampling or search method $\mathcal{S}$ identifies a set of candidate systems?

For PGS, one might be interested in the guarantee that for an instance of the combined procedure $\mathcal{S} + \mathcal{R}$, a good selection is made with probability $\geq 1 - \alpha$, i.e.,

$$\mathbb{P}(\text{GS after } \mathcal{S}) \geq 1 - \alpha. \tag{3}$$

Guarantee (3) is particularly relevant in practical problems for which the combined procedure $\mathcal{S} + \mathcal{R}$ is run only once. In contrast to $\mathbb{P}_{(\mu,\Sigma)}$, the probability measure $\mathbb{P}$ in Guarantee (3) is defined with respect to the replications taken by $\mathcal{R}$ *and* the replications and random inputs of $\mathcal{S}$; hence we will refer to Guarantee 3 as an "overall" guarantee.

The analogous overall PCS guarantee is given by

$$\mathbb{P}(\text{CS after } \mathcal{S} \mid \mu(\mathcal{X}) \in PZ(\delta)) \geq 1 - \alpha, \tag{4}$$

where $\mu(\mathcal{X})$ denotes the configuration of the random set of returned systems $\mathcal{X}$. Guarantee (4) is restrictive in the sense that it is only over instances in which the configuration of the set of returned systems is in the preference zone. When using search methods on practical problems, this is likely a rare event because search methods typically return systems with similar performance as they approach a local optimum. This is especially the case for problems in which the space of systems and the objective function are continuous. Furthermore, the decision-maker has little control over whether a search returns a configuration in PZ($\delta$) and no way of verifying this event with certainty. For these reasons, we strongly believe that PCS guarantees based on the indifference-zone condition are inappropriate for the setting of ranking and selection after search.

Guarantees (3) and (4) appear to be difficult to prove directly, the main obstacle being the probability measure $\mathbb{P}$. Whereas the probability measure $\mathbb{P}_{(\mu,\Sigma)}$ of Guarantees (1) and (2) is specific to the set of systems, $\mathbb{P}$ is defined on an enlarged probability space. Proving statements involving $\mathbb{P}$ may require knowledge of the likelihood that $\mathcal{S}$ returns an arbitrary set of systems—knowledge that is unavailable to us.

A more promising approach to proving Guarantees (3) and (4) is to condition on the set of returned systems $\mathcal{X}$. This yields the conditional guarantees

$$\mathbb{P}(\text{GS after } \mathcal{S} \mid \mathcal{X}) \geq 1 - \alpha \quad \text{for all } \mathcal{X}, \tag{5}$$

$$\mathbb{P}(\text{CS after } \mathcal{S} \mid \mathcal{X}) \geq 1-\alpha \quad \text{for all } \mathcal{X} \text{ s.t. } \mu(\mathcal{X}) \in \text{PZ}(\delta). \tag{6}$$

By the law of total expectation, Guarantees (5) and (6) imply Guarantees (3) and (4), respectively. Conditioning on the set of systems $\mathcal{X}$ has the advantage of fixing the distribution of the replications taken by $\mathcal{R}$. Therefore the probability measure in Guarantees (5) and (6) more closely resembles $\mathbb{P}_{(\mu,\Sigma)}$ of Guarantees (1) and (2).

Proposition 2.1 makes use of this observation to establish conditions under which Guarantee (5)—and hence Guarantee (3)—follows from Guarantee (2). An analogous result for the PCS guarantees follows from the same proof. Although Proposition 2.1 is trivial to prove, it is important because it shows how ranking-and-selection procedures can be used safely after sampling or search.

**Proposition 2.1 (Good Selection after Sampling or Search)** *Suppose that a ranking-and-selection procedure $\mathcal{R}$ takes as input a random set of systems $\mathcal{X}$ returned by a search or sampling method $\mathcal{S}$. If $\mathcal{R}$ does not reuse any replications taken by $\mathcal{S}$, but instead takes its own replications and guarantees*

$$\mathbb{P}_{(\mu,\Sigma)}(\text{GS}) \geq 1-\alpha \quad \textit{for all } \mu,$$

*then*

$$\mathbb{P}(\text{GS } \textit{after } \mathcal{S} \mid \mathcal{X}) \geq 1-\alpha \quad \textit{for all } \mathcal{X}.$$

*Proof.* After running $\mathcal{S}$, the set of returned systems $\mathcal{X}$ is fixed. Therefore the replications taken by $\mathcal{R}$ are drawn from fixed distributions parameterized by the mean vector $\mu(\mathcal{X})$ and variance vector $\Sigma(\mathcal{X})$. It follows that for all $\mathcal{X}$,

$$\mathbb{P}(\text{GS after } \mathcal{S} \mid \mathcal{X}) = \mathbb{P}_{(\mu(\mathcal{X}),\Sigma(\mathcal{X}))}(\text{GS}) \geq 1-\alpha. \quad \square$$

An obvious deficiency of Proposition 2.1 is that $\mathcal{R}$ does not use any of the replications taken by $\mathcal{S}$. When $\mathcal{S}$ is a sampling method, any replications that may have been collected during $\mathcal{S}$ can be reused in $\mathcal{R}$ without affecting Guarantees (5) and (6) since conditional on the set of returned systems these replications are statistically identical to those taken by $\mathcal{R}$, namely, i.i.d. and independent across systems. The more complicated case of reusing replications when $\mathcal{S}$ is a search method is discussed in the next section.

## 2.4 Reusing Search Replications

Reusing search data in a ranking-and-selection procedure is desirable since computational effort has already been expended to generate them. For example, in parallel computing environments, great efficiency can be gained by having processors communicate the observed performance of explored systems to other processors to help with screening and search.

When a ranking-and-selection procedure reuses search replications, Guarantees (5) and (6) are harder to prove directly because the selection decisions now depend on the data $\mathbf{Y}(x_1), \ldots, \mathbf{Y}(x_k)$, replications we have shown to be dependent. A potential approach to proving Guarantees (5) and (6) is to further condition on $\mathcal{G}_k$, i.e., the $\sigma$-field generated by the sequence of systems, the search replications, and the random inputs used to identify new systems. Conditioning on $\mathcal{G}_k$ simplifies the probability measure over the selection decisions.

However, this approach also runs into a major problem: guarantees conditional on $\mathcal{G}_k$ will not hold in the almost-sure sense for any fixed ranking-and-selection procedure that takes a finite number of samples. For realizations of the search in which the good systems underperform, the resulting PCS and PGS will be reduced. Letting the sample means of the good systems tend to negative infinity will almost always ensure that PCS and PGS are made arbitrarily small, eventually violating their guarantees.

With no apparent approach to directly prove Guarantees (5) and (6), it is natural to wonder if they in fact hold for existing ranking-and-selection procedures that reuse search data. To address this question, we attempt to find instances of optimization problems and searches for which Guarantees (3) and (4) are violated, thereby implying that Guarantees (5) and (6) are violated.

# 3 Adversarial Search

We present a search-like method, Adversarial Search (AS), that is designed to exploit the dependence of the search data in a way that misleads ranking-and-selection procedures. AS is designed to amplify the difficulties in ensuring Guarantees (3) and (4) in a contrived manner. Later we will do the same for a more realistic search procedure.

A detailed description of AS is given in Algorithm 1. In short, AS introduces a $\delta$-better system (relative to the best) when the best system thus far looks best, i.e., has the highest sample mean, and a $\delta$-worse system otherwise.

**ALGORITHM 1:** Adversarial Search (AS)

```
Data: An integer $k > 1$ and initial system $X_1$ (either fixed or random).
Result: A sequence of systems $X_2, \dots, X_k$.
Take replications $Y_j(X_1)$, $j = 1, \dots, n_{01}$, and calculate $\bar{Y}(X_1)$;
$i^*_{true} \leftarrow 1$ (index of system with highest true performance $g(X_i)$);
$i^*_{obs} \leftarrow 1$ (index of system with highest observed mean $\bar{Y}(X_i)$);
for $i \leftarrow 2$ to $k$ do
    if $i^*_{obs} = i^*_{true}$ then
        Introduce a system $X_i$ s.t. $g(X_i) = g(X_{i^*_{true}}) + \delta$;
        $i^*_{true} \leftarrow i$;
    else
        Introduce a system $X_i$ s.t. $g(X_i) = g(X_{i^*_{true}}) - \delta$;
    end
    Take replications $Y_j(X_i)$, $j = 1, \dots, n_{0i}$, and calculate $\bar{Y}(X_i)$;
    if $\bar{Y}(X_i) > \bar{Y}(X_{i^*_{obs}})$ then
        $i^*_{obs} \leftarrow i$;
    end
end
```

The intuition behind AS is as follows. When the best system thus far has the highest observed performance, AS introduces a new best system, thereby leaving the previous best system as a tough competitor. And when the best system thus far does not have the highest observed performance, systems that are $\delta$ worse than the best system are introduced, thereby leaving the best system as a weak competitor. In this way, AS attempts to return configurations with search data in which the best system does not look best.

Every set of systems returned by AS has a unique best system that is at least $\delta$ better than all of the others, i.e., $\mu(\mathcal{X}) \in \text{PZ}(\delta)$ for every $\mathcal{X}$. Therefore the events of correct selection and good selection are equivalent—as are Guarantees (3) and (4)—when applying a ranking-and-selection procedure after AS.

We call AS a "search-like" method because it relies on two unworkable assumptions that are beyond our formal definition of search: (i) a sufficiently large space of systems from which to draw systems of a given performance and (ii) knowledge of the objective function. Although AS does not satisfy our definition of search, it can serve as a near-worst-case benchmark for testing if the guarantees of ranking-and-selection procedures reusing search data are robust to all optimization problems and all search methods. This claim is justified by the observation that there exist instances of search methods and optimization problems that behave exactly the same as AS.

To demonstrate this assertion, consider an example in which there are four systems in $\Theta$. Without loss of generality, let them be labeled 1, 2, 3, and 4 with performances $g(1) = 0$, $g(2) = 0$, $g(3) = \delta$ and $g(4) = 2\delta$. Suppose that the simulation budget allows only three of the four systems to be evaluated, i.e., $k = 3$. We consider a particular search method over these four systems. First, Systems 2 and 3 are evaluated by taking $n_{02}$ and $n_{03}$ samples, respectively. If System 2 has a higher sample mean, System 1 is next evaluated by taking $n_{01}$ samples, and if System 3 has a higher sample mean, System 4 is next evaluated by taking $n_{04}$ replications. One possible reasoning for identifying the third system in this fashion is that the decision-maker believes System 1 will have performance similar to System 2 while System 4 will have performance similar to System 3. One can verify that this search method behaves exactly the same as AS when letting $X_1 = 2$ w.p. 1.

In this fashion, one can construct similar optimization problems and search methods that mimic AS for larger numbers of returned systems ($k > 3$). Hence finding violated guarantees in our experiments with AS will show that the guarantees of ranking-and-selection procedures that reuse search data do not hold for all optimization problems and all search methods.

# 4 Experiments with Reusing Search Replications

In Sections 4.1 and 4.2, we test selection and subset-selection procedures, respectively, reusing search data from AS. In addition to using AS, we also test these procedures in the slippage configuration (SC): $g(x_{[i]}) = g(x_{[k]}) - \delta$ for all $1 \leq i < k$. For the ranking-and-selection procedures we consider, PCS is minimized in the slippage configuration; therefore testing the slippage configuration gives a lower bound on PCS for the procedures when applied after sampling. For both settings, we perform 10 000 macroreplications at each value of $k$ and calculate the empirical PCS—an unbiased estimate of the left-hand sides of Guarantees (3) and (4). In Section 4.3, we test a subset-selection procedure that reuses replications from a realistic search.

In all of our experiments, we choose a desired PCS $1 - \alpha = 0.95$ and fix an indifference-zone parameter $\delta = 1$, an initial sample size $n_0 = 10$, and a common variance $\sigma^2 = 1$. Although a common variance is unrealistic for simulation optimization problems, we should expect the guarantees of ranking-and-selection procedures that reuse search data to hold even in this stylized setting.

## 4.1 Selection Procedures

Selection procedures based on the indifference-zone formulation were first proposed by [2] for the case of common, known variance. Selection procedures for uncommon, unknown variances were later developed by [6] and [24]. In our experiments, we test the Bechhofer and Rinott procedures, the latter of which because it is easier to implement than the Dudewicz and Dalal procedure. The Bechhofer procedure is designed to ensure a tight PCS of $1-\alpha$ in the slippage configuration while the Rinott procedure is more conservative because it must allow for unequal, unknown variances.

We focus on single- and multi-stage selection procedures—of which the Bechhofer and Rinott procedures are two examples—because all of the required replications can be taken during the search. On the other hand, fully sequential procedures take replications from systems one at a time. Some fully sequential procedures, e.g. [15], can be easily incorporated into a selection-after-search framework by taking only $n_0$ replications of each system during the search. For other fully sequential procedures that do not require an initial sample size of $n_0$ replications from each system, e.g. [8], it remains an open question how they might be applied in conjunction with search.

The details of the two selection procedures we test are as follows:

**Bechhofer** Take $N = \lceil (2h_B^2\sigma^2)/\delta^2 \rceil$ replications of each system where $h_B$ is the $\alpha$-upper quantile of the maximum of a $k-1$ dimensional multivariate normal vector with means 0, variance 1, and pairwise correlations 1/2. The constant $h_B$ can be derived from the values of $N$ tabulated in Table 2.1 on page 19 of [3]. Select the system with the highest sample mean.

**Rinott** Take $n_0$ replications of each system and calculate $S_i^2$, the sample variance of the $i$th system based on the initial replications. Take additional replications so that the $i$th system has a total of $N_i$ replications where

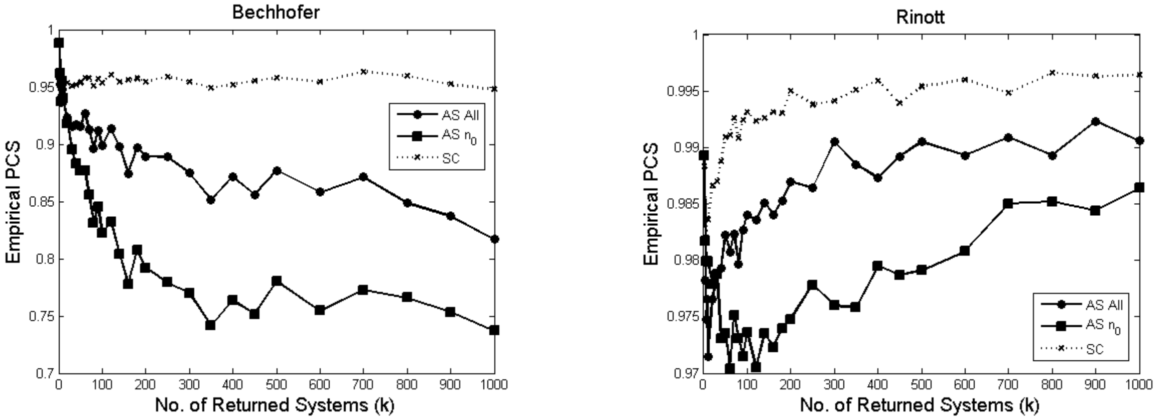
$$N_i = \max\left\{ n_0, \left\lceil \frac{h_R^2 S_i^2}{\delta^2} \right\rceil \right\}$$

and $h_R = h(k, 1-\alpha, n_0)$ is the solution to

$$\int_0^\infty \left[ \int_0^\infty \Phi\left( \frac{h}{\sqrt{\nu\,(1/x + 1/y)}} \right) f_\nu(x)dx \right]^{k-1} f_\nu(y)\,dy = 1-\alpha,$$

where $\nu = n_0 - 1$ and $f_\nu$ is the pdf of a chi-squared random variable with $\nu$ degrees of freedom. The constant $h_R$ is tabulated in Table 2.13 on pages 62–63 of [3]. Select the system with the highest overall sample mean.

We test the Bechhofer and Rinott procedures in two settings: (i) "AS All": all of the required replications are taken during AS, i.e., $n_{0i} = N$ and $n_{0i} = N_i$, respectively, for $i = 1, \ldots, k$ and (ii) "AS $n_0$": only $n_0$ replications of each system are taken during AS and the remaining replications are taken afterwards, i.e., $n_{0i} = 10$ for $i = 1, \ldots, k$.



(a) Bechhofer

(b) Rinott

Figure 1: Based on the usual normal confidence intervals, the empirical PCSs of the Bechhofer and Rinott procedures are accurate to within ±0.01 and ±0.005, respectively, of the true PCSs.

In Figure 1a, the empirical PCS for the Bechhofer procedure reusing replications from AS quickly falls below the guaranteed PCS of 0.95. The empirical PCS for the case of taking $n_0$ replications during AS deteriorates more rapidly than when all of the replications are taken during AS. One possible explanation is that with only $n_0$ replications, the initial ranking of systems is more variable and so AS likely returns less-favorable configurations of systems, such as the slippage configuration.

The conservativeness of the Rinott procedure is evident in Figure 1b where the empirical PCS for the slippage configuration increases to above 0.995 as the number of returned systems increases to 1000. The empirical PCS for the Rinott procedure applied after AS initially decreases until around 50 to 100 returned systems before increasing again. This trend suggests that the conservativeness of the Rinott procedure offsets most of the impact of AS on decreasing PCS. Although the empirical PCS of either taking $n_0$ or all replications never drops below the desired PCS of 0.95, it stays below the empirical PCS of the slippage configuration in both cases.

## 4.2 Subset-Selection Procedures

Subset-selection procedures were first developed by [11] as an alternative to the indifference-zone formulation of [2]. Instead of selecting a single system, subset-selection procedures return a subset of systems $I$ that contains the best system with at least a prespecified probability. In further contrast to many selection procedures, subset-selection procedures often take a fixed number of replications.

Since we wish to handle the cases of correct selection and good selection simultaneously, we will rely on variations of the Gupta procedure that provide PCS guarantees when the indifference-zone assumption is satisfied. That is, they deliver the guarantee

$$\mathbb{P}_{(\mu,\Sigma)}\{x_{[k]} \in I\} \geq 1-\alpha \quad \text{for all } \mu \in PZ(\delta).$$

Two such procedures are as follows:

**Modified Gupta** A modified version of the original Gupta procedure for common, known variance $\sigma^2$ that uses a different "yardstick" based on the indifference-zone assumption. Take $n_0$ replications from each system and return the set of systems

$$I = \{X_i : \bar{Y}(X_i) \geq \bar{Y}(X_j) - (W-\delta)^+ \text{ for all } j \neq i\},$$

where $W = h_B \sigma \sqrt{2/n_0}$, and $h_B$ is Bechhofer's constant mentioned in Section 4.1. This procedure was developed by [26], except that his procedure neglected to account for the necessary positive-part operator in the term $(W-\delta)^+$. See appendix for a complete proof of the PCS guarantee.

**Screen-to-the-Best** An extension of Gupta's procedure to handle unknown and uncommon variances developed by [19]. Take $n_0$ replications from each system and return the set of systems

$$I = \{X_i : \bar{Y}(X_i) \geq \bar{Y}(X_j) - (W_{ij}-\delta)^+ \text{ for all } j \neq i\},$$

where

$$W_{ij} = t\left(\frac{S_i^2}{n_0} + \frac{S_j^2}{n_0}\right)^{1/2}$$

and $t = t_{(1-\alpha_0)^{1/(k-1)}, n_0-1}$ is the upper-quantile of a $t$ distribution with $n_0 - 1$ degrees of freedom. The Screen-to-the-Best procedure was further extended to unequal sample sizes [5], but we do not consider the extension here.

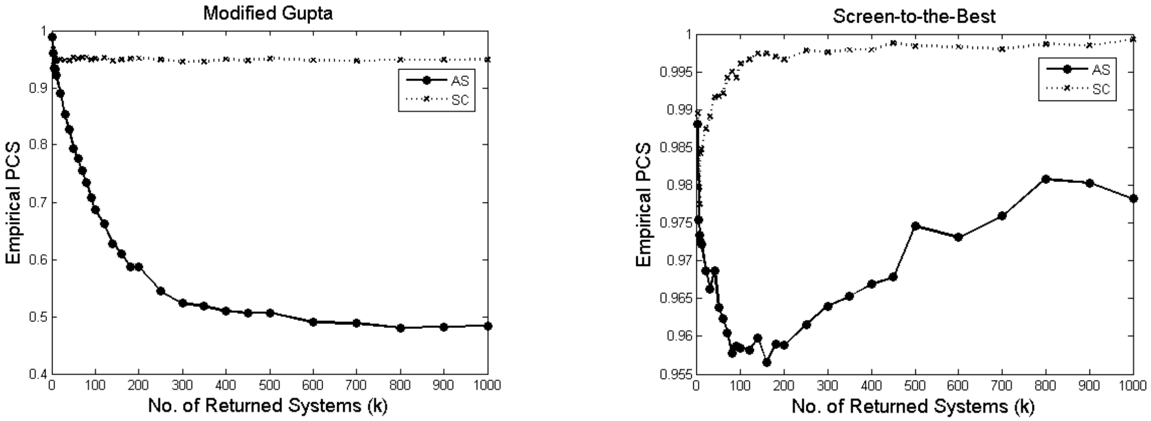
The Modified Gupta procedure is designed to be tight in the slippage configuration while the Screen-to-the-Best procedure is more conservative.

As seen in Figure 2a, the empirical PCS for the Modified Gupta procedure after AS quickly drops below the desired level of 0.95. The empirical PCS further deteriorates as the number of systems increases, even to the point where PCS is nearly half of its guaranteed value!

In Figure 2b the empirical PCS of the Screen-to-the-Best procedure after AS stays well below the empirical PCS of the slippage configuration, dropping to just above its guarantee of 0.95 at around 100 systems, before increasing again. We suspect that there exist problem instances and parameter settings for which the PCS for Screen-to-the-Best will fall below its guaranteed level.

## 4.3 A Realistic Search

The AS framework in Sections 4.1 and 4.2 is clearly unrealistic, yet effective at showing that ranking-and-selection guarantees can suffer from reusing search data. We now consider



(a) Modified Gupta

(b) Screen-to-the-Best

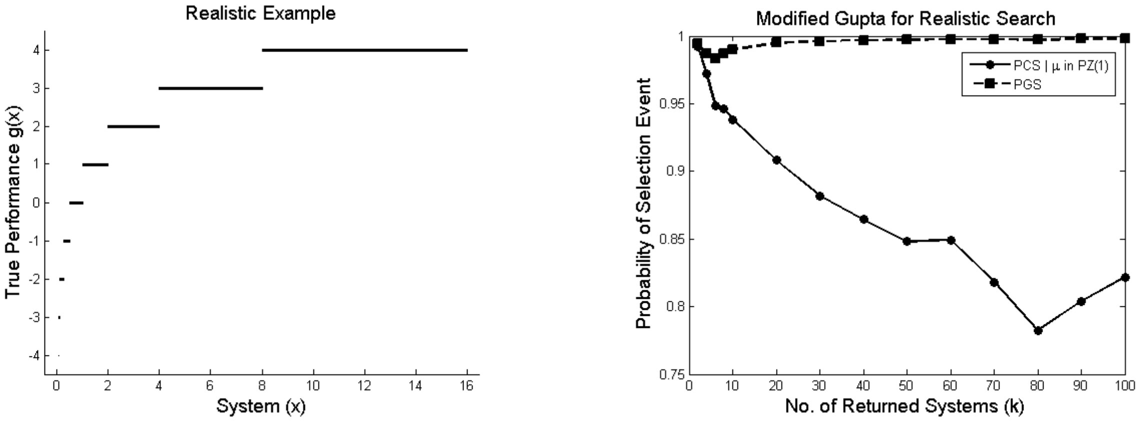
Figure 2: Based on the usual normal confidence intervals, the empirical PCSs of the Modified Gupta and Screen-to-the-Best procedures are accurate to within ±0.01 of the true PCSs.

a more realistic optimization problem and search method and study the performance of a subset-selection procedure that reuses the replications from search. The optimization problem we test is to maximize $g(x) = \lceil \log_2 x \rceil$ on the interval $[1/16, 16]$, a step-like function with ever-widening steps whose values range on the integers from $-4$ to 4, as plotted in Figure 3a.

The search method we consider first evaluates the system $x_1 = 0.75$ by taking $n_0 = 10$ samples. Each new system is chosen uniformly at random from an interval of width 2 centered around the system with the highest sample mean among the explored systems, but otherwise independently of past observations. When the search attempts to return a system from outside the interval $[1/16, 16]$, the nearest endpoint is returned as the next system. In this way, the search finds new systems from within a neighborhood of the system with the best observed performance.

We tested the Modified Gupta subset-selection procedure for 100 000 macroreplications and calculated the overall PGS and PCS of Guarantees (3) and (4), respectively, at a range of values of $k$. The empirical PCS of Guarantee (4) was averaged over only the macroreplications for which the configuration of the returned systems was in the preference zone. For $k < 10$, this amounted to roughly 1 in 4 macroreplications, while for $k = 100$, it was roughly 1 in 100. As seen in Figure 3b the empirical PGS remains well above its guaranteed level of 0.95 while the empirical PCS in the preference zone falls below this threshold.

The experimental results suggest that the event of the returned configuration being in the preference zone makes correct decisions less likely. And indeed, the structure of the objective function is responsible. When the search returns a configuration in the preference zone, it means that only one system was explored from the highest explored "step" of the objective function. Because the search looks for new systems in an interval centered around the best-looking system, returning a configuration in the preference zone likely means the true best system did not have the highest sample mean immediately after it was evaluated. For this reason, the probability of correct selection suffers in these instances.



(a) Objective function.

(b) Modified Gupta

Figure 3: Based on the usual normal confidence intervals, the empirical PCS of the Modified Gupta is within ±0.003 for $k \leq 10$ and within ±0.03 for $k \geq 80$.

# 5 Conclusions

In this paper, we considered the framework of applying a ranking-and-selection procedure on a random set of systems returned by a sampling or search method. We formulated purposeful guarantees on PCS and PGS and showed how they can immediately follow from the traditional guarantees of ranking-and-selection procedures. We then argued that, in this setting, PGS guarantees are superior to PCS guarantees based on the indifference-zone condition.

We studied issues that arise from reusing replications taken during search as input to a ranking-and-selection procedure. We explained how search methods inherently produce replications that are conditionally dependent on the sequence of returned systems, an aspect that had been overlooked in the proofs of existing procedures that reuse search data. Furthermore, we designed a search-like method that exploited this dependence to demonstrate how the PCS and PGS of ranking-and-selection procedures can suffer as a consequence.

In our experiments, we observed empirical PCS well below the guaranteed level for ranking-and-selection procedures with tight guarantees, e.g. Bechhofer and Modified Gupta. For more conservative procedures that handle unequal and unknown variances, e.g. Rinott and Screen-to-the-Best, we also observed lower-than-expected PCS, but not enough to violate the guarantees. These results suggest that for practical simulation optimization problems, the procedures proposed in the literature that reuse search replications can likely be performed with little fear of encountering violated guarantees. Since realistic simulation optimization searches return systems with ever-similar performance, we believe that PGS guarantees will in practice be robust to reusing search data.

How ranking-and-selection procedures might be designed to safely reuse search data while offering provable guarantees remains an open question. One way forward may be procedures with asymptotic PCS and PGS guarantees as $\delta \to 0$, e.g. $\mathcal{KN}$+ and $\mathcal{KN}$++ of [16]. For such procedures, as the required number of replications increases, the influence of the replications taken during search will diminish.

# Acknowledgments

This work is supported by the National Science Foundation under grants DGE-1144153 and CMMI-1537394.

# APPENDIX

In this appendix we prove the validity of the Modified Gupta procedure, namely, that it satisfies

$$\mathbb{P}_{(\mu,\Sigma)}\{x_{[k]} \in I\} \geq 1-\alpha \quad \text{for all } \mu \in PZ(\delta),$$

where $\Sigma$ is a vector whose elements are all $\sigma^2$.

Fix an arbitrary set of systems $\{x_1, \ldots, x_k\}$ with configuration $\mu \in PZ(\delta)$ and a common, known variance $\sigma^2 < \infty$. The procedure takes $n_0$ i.i.d. replications $Y_1(x_i), \ldots, Y_{n_0}(x_i)$ for $i = 1, \ldots, k$. Let $\bar{Y}(x_i)$, $i = 1, \ldots, k$, be the corresponding sample means. The Modified Gupta procedure preserves a set of systems $I$ such that for each system $i \in I$,

$$\bar{Y}(x_i) \geq \max_{j \neq i} \bar{Y}(x_j) - (W-\delta)^+,$$

where $W = h_B \sigma \sqrt{2/n_0}$ and $h_B$ is Bechhofer's constant. Note that the positive part operator on the term $(W-\delta)^+$ is needed to ensure that at least one system, the one with the largest sample mean, is included in the set $I$.

We follow the proof of Gupta's procedure presented by [17] and modify where necessary:

$$\begin{aligned}
\mathbb{P}_{(\mu,\Sigma)}\{x_{[k]} \in I\} &= \mathbb{P}_{(\mu,\Sigma)}\left\{\bar{Y}(x_{[k]}) \geq \max_{j\neq k} \bar{Y}(x_{[j]}) - (W-\delta)^+\right\} \\
&= \mathbb{P}_{(\mu,\Sigma)}\left\{\bar{Y}(x_{[k]}) \geq \max_{j\neq k} \bar{Y}(x_{[j]}) - \left(h_B\sigma\sqrt{2/n_0} - \delta\right)^+\right\} \\
&= \mathbb{P}_{(\mu,\Sigma)}\left\{\bar{Y}(x_{[k]}) \geq \bar{Y}(x_{[j]}) - \left(h_B\sigma\sqrt{2/n_0} - \delta\right)^+ \quad \text{for all } j \neq k\right\} \\
&= \mathbb{P}_{(\mu,\Sigma)}\left\{\frac{\bar{Y}(x_{[j]}) - \bar{Y}(x_{[k]}) - \big(g(x_{[j]}) - g(x_{[k]})\big)}{\sigma\sqrt{2/n_0}}\right. \\
&\qquad\qquad \left. \leq \left(h_B - \frac{\delta}{\sigma\sqrt{2/n_0}}\right)^+ - \frac{\big(g(x_{[j]}) - g(x_{[k]})\big)}{\sigma\sqrt{2/n_0}} \quad \text{for all } j \neq k\right\} \\
&\geq \mathbb{P}_{(\mu,\Sigma)}\left\{\frac{\bar{Y}(x_{[j]}) - \bar{Y}(x_{[k]}) - \big(g(x_{[j]}) - g(x_{[k]})\big)}{\sigma\sqrt{2/n_0}}\right. \\
&\qquad\qquad \left. \leq h_B - \frac{\delta + \big(g(x_{[j]}) - g(x_{[k]})\big)}{\sigma\sqrt{2/n_0}} \quad \text{for all } j \neq k\right\} \\
&\geq \mathbb{P}_{(\mu,\Sigma)}\left\{Z_j \leq h_B \quad \text{for all } j \neq k\right\} = 1-\alpha\,,
\end{aligned}$$

where the variables $Z_1, \ldots, Z_{k-1}$ are distributed according to a multivariate normal with means 0, variances 1, and common pairwise correlations 1/2. The last equality follows from the definition of Bechhofer's constant $h_B$.

# References

[1] M. A. Ahmed and T. M. Alkhamis. Simulation-based optimization using simulated annealing with ranking and selection. *Computers and Operations Research*, 29(4):387–402, 2002.

[2] R. E. Bechhofer. A single-sample multiple decision procedure for ranking means of normal populations with known variances. *The Annals of Mathematical Statistics*, 25(1):16–39, 1954.

[3] R. E. Bechhofer, T. J. Santner, and D. M. Goldsman. *Design and analysis of experiments for statistical selection, screening, and multiple comparisons.* Wiley, 1995.

[4] J. Boesel, B. L. Nelson, and N. Ishii. A framework for simulation-optimization software. *IIE Transactions*, 35(3):221–229, 2003.

[5] J. Boesel, B. L. Nelson, and S.-H. Kim. Using ranking and selection to “clean up” after simulation optimization. *Operations Research*, 51(5):814–825, 2003.

[6] E. J. Dudewicz and S. R. Dalal. Allocation of observations in ranking and selection with unequal variances. *The Indian Journal of Statistics*, 37(1):28–78, 1975.

[7] E. Even-Dar, S. Mannor, and Y. Mansour. Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems. *Journal of Machine Learning Research*, 7(June):1079–1105, 2006.

[8] P. Frazier. A fully sequential elimination procedure for indifference-zone ranking and selection with tight bounds on probability of correct selection. *Operations Research*, 62(4):926–942, 2014.

[9] M. Fu. *Handbook of simulation optimization*, volume 216. Springer, 2015.

[10] P. Glynn and S. Juneja. Ordinal optimization – empirical large deviations rate estimators, and stochastic multi-armed bandits, July 2015.

[11] S. S. Gupta. On some multiple decision (selection and ranking) rules. *Technometrics*, 7(2):225–245, 1965.

[12] S. G. Henderson and B. L. Nelson, editors. *Handbooks in operations research and management science: Simulation*, volume 13. Elsevier, 2006.

[13] L. J. Hong and B. L. Nelson. Selecting the best system when systems are revealed sequentially. *IIE Transactions*, 39(7):723–734, 2007.

[14] S. R. Hunter and R. Pasupathy. Large-deviation sampling laws for constrained simulation optimization on finite sets. In *Proceedings of the 2010 Winter Simulation Conference*, pages 995–1002, 2010.

[15] S.-H. Kim and B. L. Nelson. A fully sequential procedure for indifference-zone selection in simulation. *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, 11(3):251–273, 2001.

[16] S.-H. Kim and B. L. Nelson. On the asymptotic validity of fully sequential selection procedures for steady-state simulation. *Operations Research*, 54(3):475–488, 2006.

[17] S.-H. Kim and B. L. Nelson. Selecting the best system. In S. G. Henderson and B. L. Nelson, editors, *Simulation*, volume 13 of *Handbooks in operations research and management science*. Elsevier, 2006.

[18] B. L. Nelson and F. Matejcik. Using common random numbers for indifference-zone selection and multiple comparisons in simulation. *Management Science*, 41(12):1935–1945, 1995.

[19] B. L. Nelson, J. Swann, D. Goldsman, and W. Song. Simple procedures for selecting the best simulated system when the number of alternatives is large. *Operations Research*, 49(6):950–963, 2001.

[20] S. Olafsson. Iterative ranking-and-selection for large-scale optimization. In *Proceedings of the 1999 Winter Simulation Conference*, pages 479–485, 1999.

[21] J. Pichitlamken and B. L. Nelson. Selection-of-the-best procedures for optimization via simulation. In *Proceedings of the 2001 Winter Simulation Conference*, pages 401–407, 2001.

[22] J. Pichitlamken and B. L. Nelson. A combined procedure for optimization via simulation. *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, 13(2):155–179, 2003.

[23] J. Pichitlamken, B. L. Nelson, and L. J. Hong. A sequential procedure for neighborhood selection-of-the-best in optimization via simulation. *European Journal of Operational Research*, 173(1):283–298, 2006.

[24] Y. Rinott. On two-stage selection procedures and related probability inequalities. *Communications in Statistics – Theory and Methods*, 7(8):799–811, 1978.

[25] T. A. Sriver, J. W. Chrissis, and M. A. Abramson. Pattern search ranking and selection algorithms for mixed variable simulation-based optimization. *European Journal of Operations Research*, 198(3):878–890, 2009.

[26] P. van der Laan. Subset selection of an almost best treatment. *Biometrical Journal*, 34(6):647–656, 1992.