IIE Transactions (2007) **39**, 723–734
Copyright © "IIE"
ISSN: 0740-817X print / 1545-8830 online
DOI: 10.1080/07408170600838415

Selecting the best system when systems are revealed sequentially

L. JEFF HONG[1] and BARRY L. NELSON[2,*]

[1] *Department of Industrial Engineering & Logistics Management, The Hong Kong University of Science and Technology, Clear Water Bay, Hong Kong, China*
E-mail: hongl@ust.hk
[2] *Department of Industrial Engineering & Management Sciences, Northwestern University, 2145 Sheridan Road, Evanston, IL 60208-3119, USA*
E-mail: nelsonb@northwestern.edu

Received June 2004 and accepted March 2005

Statistical Ranking and Selection (R&S) is a collection of experiment design and analysis techniques for selecting the system with the largest or smallest mean performance from among a finite set of alternatives. R&S procedures have received considerable research attention in the stochastic simulation community, and they have been incorporated in commercial simulation software. All existing procedures assume that the set of alternatives is available at the beginning of the experiment. In many situations, however, the alternatives are revealed (generated) sequentially during the experiment. We introduce procedures that are capable of selecting the best alternative in these situations and provide the desired statistical guarantees.

Keywords: Ranking and selection, system design, optimization via simulation

1. Introduction

Ranking-and-Selection (R&S) procedures have been proposed to select the simulated system with the largest or smallest mean performance from among a finite number of alternative systems (see Bechhofer *et al.* (1995) for a summary). These procedures require that all alternative systems are known at the beginning of the experiment. In many cases, however, some alternative systems may not be known at the beginning and they may be revealed during the process of the experiment.

One instance of this situation is iterative system design. System design is naturally a sequential process. New designs are created based on the evaluation of the current designs. The design process typically lasts several rounds and the design selected in the end should at least be the best design among all alternatives that have been evaluated. Another instance is optimization via simulation (see Fu (2002) for a thorough review). The optimization-via-simulation process can be thought of as a system-design process with no human interaction. Optimization algorithms, e.g., the simulated annealing algorithm of Alrefaei and Andradóttir (1999), the nest partitions algorithm of Shi and Ólafsson (2000), and the COMPASS algorithm of Hong and Nelson (2006) decide what new solutions to visit on each iteration based on information on the current best solution. For example, in the COMPASS algorithm new candidate solutions are generated on each iteration based on the location of the selected best solution among all solutions visited through the previous iteration. In iterative system design and optimization via simulation the experiment may terminate when some stopping rule is satisfied, and then the selected system on that iteration is treated as the best system among all alternative systems generated. Therefore, it is important to find the best system (design or solution) on each iteration to help generate better systems on the next iteration, and also to provide a certain (statistical) guarantee that the selected system is the true best whenever the experiment terminates.

This problem is different from the typical R&S problem since the number of alternative systems is not known at the beginning of the experiment, and a selection decision has to be made each time new alternative systems are revealed. The existing selection procedures cannot be applied directly to solve this problem. In this paper, we propose two general approaches to solve the problem, the single-elimination approach and the stop-and-go approach, and design several procedures accordingly.

There are two types of R&S procedures: frequentist procedures and Bayesian procedures. Frequentist procedures

[*] Corresponding author

such as Rinott (1978) and Kim and Nelson (2001) allocate simulation effort to different systems to ensure a probability of correct selection even for the least favorable configuration. They are typically conservative for an average case. Bayesian procedures such as Chen *et al.* (2000) and Chick and Inoue (2001a, 2001b) maximize the posterior probability of correct selection or minimize an opportunity cost given a simulation budget. However, they do not provide a guaranteed probability of correct selection. In this paper we take the frequentist viewpoint and the procedures proposed in this paper all guarantee a prespecified probability of correct selection.

The procedures provided in this paper are also sequential procedures. Among the frequentist procedures, Kim and Nelson (2001) first introduce a fully sequential procedure to solve simulation selection problems. They approximate the sum of differences between two systems as a Brownian motion process and use a triangular continuation region to determine the stopping time of the selection process. Like other frequentist procedures, sequential selection procedures are designed for the least favorable configuration where the differences between the true mean of the best system and the true means of all other systems are exactly the Indifference-Zone (IZ) parameter, δ. However, unlike the popular two-stage procedures such as Rinott (1978), sequential selection procedures terminate faster if the systems are not in this unfavorable configuration.

Boesel *et al.* (2003) and Pichitlamken *et al.* (2006) have applied R&S in the optimization-via-simulation context. Boesel *et al.* (2003) propose efficient R&S procedures that can be used at the end of the optimization process to guarantee that the chosen solution is the true best among all solutions visited by the optimization process with a certain confidence. However, their procedures do not help make correct selection decisions on each iteration of the optimization process. Pichitlamken *et al.* (2006) propose a selection procedure that can be applied on each iteration of the optimization process to provide a statistical guarantee on that iteration. However, their procedure does not provide an overall statistical guarantee on all visited solutions.

The procedures proposed in this paper combine aspects of Boesel *et al.* (2003) and Pichitlamken *et al.* (2006) to help make selection decisions on each iteration and provide an overall statistical guarantee at the end of the design or optimization process. However, there is a price to pay for this very strong inference. Providing a statistical guarantee of correct selection on each iteration, especially when the number of candidate solutions that the optimization process visits is large, is very demanding. Therefore, our proposed procedures are only applicable to iterative system-design problems and small-scale optimization problems. Extensions of these procedures to optimization problems with a very large number of alternatives is a subject of ongoing research.

The paper is organized as follows. The mathematical formulation of the problem is provided in Section 2. We discuss the single-elimination approach and the stop-and-go approach in Sections 3 and 4, respectively. Numerical examples are presented in Section 5, followed by the conclusions and possible future research directions in Section 6.

2. The mathematical formulation

Consider the following generic algorithm for generating new alternative systems.

System Generating Algorithm (SGA)

Step 0. We start with k_0 simulated systems, $\pi_1, \pi_2, \ldots, \pi_{k_0}$, and $k_0 \geq 2$. Let i and K_i denote the iteration count and the total number of alternative systems generated through iteration i, respectively. Set $i = 0$ and $K_0 = k_0$. Go to Step 2.

Step 1. Let $i = i + 1$. Generate $k_i \geq 1$ alternatives, $\pi_{K_{i-1}+1}$, $\pi_{K_{i-1}+2}, \ldots, \pi_{K_{i-1}+k_i}$. Let $K_i = K_{i-1} + k_i$.

Step 2. Select the best system in all K_i alternatives π_1, $\pi_2, \ldots, \pi_{K_i}$.

Step 3. If the stopping rule is satisfied, stop; otherwise, go to Step 1.

Remark 1. In Step 1 of SGA k_i is determined by the designer or the optimization algorithm, and it may depend on the simulation budget (i.e., available time to solve the problem). However, the selection procedures provided in this paper work for any deterministic $k_i \geq 1$.

Clearly, Step 2 of SGA is where R&S should be applied. IZ selection procedures select the best system with a Probability of Correct Selection (PCS) greater than or equal to $1 - \alpha$ from a given set of alternative systems whenever the true mean performance of the best system in the set is at least δ greater than the true mean of the second-best system. The IZ parameter $\delta > 0$ is set by the experimenter to the minimum difference in expected performance that it is important to detect. Differences of less than δ are considered practically insignificant. In this paper, we assume that the best system is the system with the largest mean performance.

IZ selection procedures for SGA are different from typical IZ selection procedures, since the total number of systems for comparison is not fixed. Since SGA generates new alternative systems iteration by iteration, it is natural to design IZ selection procedures that compare the new alternative systems to only the previous best system, and systems determined not to be the best are eliminated and never considered again. For this type of procedure, allocating the Probability of Incorrect Selection (PICS) among the different comparisons can solve the problem of changing the total number of alternatives. When the total number of alternatives that SGA will generate can be bounded, one can allocate the PICS uniformly. When the total number cannot be bounded, one can design an infinite sequence, whose sum converges to PICS, to allocate the PICS. As we show in Section 3, both approaches guarantee the overall PCS whenever SGA terminates.

Another approach is to solve a R&S problem with K_i alternatives on the ith iteration. We can apply an IZ selection procedure to all alternatives, whether or not they have been eliminated during previous iterations. In Section 4, we show that this approach can be efficient in terms of sampling; however, it does require more switches among the simulated alternative systems, which can be costly for computer simulations.

Let $X_{i\ell}, \ell = 1, 2, \ldots$, denote the observations taken from system $\pi_i, i = 1, 2, \ldots$. In this paper, we assume the following:

Assumption 1. The $X_{i\ell}$ are normally distributed with unknown mean μ_i and unknown variance σ_i^2 for all $i = 1, 2, \ldots$ and $X_{i\ell}$ is independent of X_{jm} if either $i \neq j$ or $\ell \neq m$ or both.

Assumption 1 is a standard assumption used in R&S. Since most simulation models are used to study the average behavior of a system, it is reasonable to make the normality assumption because of the Central Limit Theorem. Independence of $X_{i\ell}, \ell = 1, 2, \ldots$, is a direct result of making replications. When simulation observations are obtained within a single replication of a steady-state simulation, then techniques such as batch means allow assumption 1 to hold approximately (see, for instance, Law and Kelton (2000)). Assuming $X_{i\ell}$ independent of $X_{j\ell}$ implies that we do not use common random numbers. Although we expect our procedures to work, in the sense of delivering at least the desired probability of correct selection in the presence of common random numbers, we do not exploit them.

3. Single-elimination approach

An ideal selection procedure for SGA is a single-elimination procedure. Whenever a system is declared inferior on any iteration, it never needs to be evaluated again. In the iterative system-design context, this means that system designers always compare the newly generated designs to the best design from previous rounds. In the optimization-via-simulation context, this is analogous to deterministic optimization where solutions can be evaluated without noise. Two single-elimination sequential-selection procedures, called SEB and SEU, are provided in this section. SEB assumes the existence of a bound on the total number of alternatives SGA generates. SEU does not need this assumption; however, it requires an infinite sequence of positive numbers whose sum equals the PICS.

3.1. *Single elimination with bound*

In many cases one may have an upper bound on how many systems will be evaluated before the selection process starts. In the iterative system-design context, for instance, the system designers may know that they have time to evaluate at most 30 designs, or that there are at most 60 possible designs. Similarly, in the optimization-via-simulation context one may be able to determine the maximum number of solutions that could be simulated in advance. Although the system designer or optimization algorithm *could* exhaust the set of all possible designs, they likely will not, making it more efficient to do iterative design or use an optimization via simulation algorithm rather than run a R&S procedure on all feasible designs from the beginning.

We assume there exists a known bound on the total number of alternatives SGA will generate, say K. On iteration i there are $k_i + 1$ systems in comparison, $\pi_{K_{i-1}+1}, \pi_{K_{i-1}+2}, \ldots, \pi_{K_{i-1}+k_i}$, and $\hat{\pi}_{i-1}^*$, where $\hat{\pi}_{i-1}^*$ is the selected best on iteration $i-1$. If we use π_i^* to denote the true best system through iteration i, then $\hat{\pi}_{i-1}^*$ may not be π_{i-1}^* because of selection error. Let μ_i^* denote the mean performance of π_i^*. The following SEB procedure is called from Step 2 of SGA on iteration i:

Single-Elimination with Bound (SEB)

Input: Let I denote the set of systems in contention. If the iteration counter $i = 0$, then set $I = \{\pi_1, \pi_2, \ldots, \pi_{k_1}\}$ and $K_{-1} = 0$; otherwise, set $I = \{\pi_{K_{i-1}+1}, \pi_{K_{i-1}+2}, \ldots, \pi_{K_{i-1}+k_i}, \hat{\pi}_{i-1}^*\}$, and input the overall sample size, overall sample mean and first-stage sample variance of system $\hat{\pi}_{i-1}^*$.

Return: The system $\hat{\pi}_i^*$ along with its overall sample size, overall sample mean and first-stage sample variance.

Procedure:

Setup: Select the PCS $1/2 < 1 - \alpha < 1$, IZ parameter $\delta > 0$, and first-stage sample size $n_0 \geq 2$. Let $\lambda = \delta/2$.

Initialization: For all $p = K_{i-1} + 1, K_{i-1} + 2, \ldots, K_{i-1} + k_i$, take n_0 observations from π_p, set the total number of observations for π_p to $r_p = n_0$, and calculate the first-stage sample mean $\bar{X}_p(n_0)$ and the first-stage sample variance:

$$S_p^2(n_0) = \frac{1}{n_0 - 1}\left[\sum_{\ell=1}^{n_0} X_{p\ell}^2 - n_0(\bar{X}_p(n_0))^2\right]. \tag{1}$$

Let $r = n_0$ denote the current check point.

Computing parameters: Let $\beta = \alpha/(K-1)$. For any π_p and π_q in I and $p \neq q$, calculate:

$$a_{pq} = \frac{\eta(n_0 - 1)\left[S_p^2(n_0) + S_q^2(n_0)\right]}{2\delta}, \tag{2}$$

where

$$\eta = (2\beta)^{-\frac{2}{n_0-1}} - 1. \tag{3}$$

Elimination: Set $I^{\text{old}} = I$, and update I to be

$$I = \{\pi_p : \pi_p \in I^{\text{old}} \text{ and } r[\bar{X}_p(r_p) - \bar{X}_q(r_q)] \\ \geq -\max(0,\ a_{pq} - r\lambda),\ \forall \pi_q \in I^{\text{old}},\ p \neq q\}.$$

Stopping rule: If $|I| = 1$, then let $\hat{\pi}_i^*$ be the system in I and Return. Otherwise, for every $\pi_p \in I$ such that $r_p = r$, take an observation of system π_p, let $r_p = r_p + 1$, and update $\bar{X}_p(r_p)$. Set $r = r + 1$ and go back to Elimination.

Suppose the second-best system among $\pi_1, \pi_2, \ldots, \pi_{K_i}$ is denoted as π_i^{**} with mean μ_i^{**}. Then we have the following theorem.

Theorem 1. *Suppose that assumption* 1 *holds, and that the total number of alternatives SGA will generate is bounded by K. Then at the end of any iteration i of SGA, SEB guarantees that*:

$$\Pr\left\{\pi_i^* \text{ is selected}\right\} \geq 1 - \alpha,$$

if $\mu_i^* - \mu_i^{**} \geq \delta$.

To prove Theorem 1 we need the following definitions and lemmas.

We use $\mathcal{B}(t; \delta, \sigma^2)$ to denote the Brownian motion process with drift such that $E[\mathcal{B}(t; \delta, \sigma^2)] = \delta t$ and $\text{Var}[\mathcal{B}(t; \delta, \sigma^2)] = \sigma^2 t$. We define a triangular continuation region by $L(t) = -a + \lambda t$ and $U(t) = a - \lambda t$, where $a > 0$ and $\lambda > 0$, and let T be the first time when $\mathcal{B}(t; \delta, \sigma^2)$ exits the triangular continuation region (Fig. 1).

For any two systems π_p and π_q, let $\delta_{pq} = \mu_p - \mu_q$ and $\sigma_{pq}^2 = \sigma_p^2 + \sigma_q^2$. For any positive integers n_p and n_q, define the stochastic process:

$$\begin{aligned}\mathcal{A}\big(t; \delta_{pq}, \sigma_{pq}^2, n_p, n_q\big) \equiv\ & \mathcal{I}\{t \leq n_p\}\frac{t\mathcal{B}_p\big(n_p; \mu_p, \sigma_p^2\big)}{n_p} \\ & + \mathcal{I}\{t > n_p\}\mathcal{B}_p\big(t; \mu_p, \sigma_p^2\big) \\ & - \mathcal{I}\{t \leq n_q\}\frac{t\mathcal{B}_q\big(n_q; \mu_q, \sigma_q^2\big)}{n_q} \\ & - \mathcal{I}\{t > n_q\}\mathcal{B}_q\big(t; \mu_q, \sigma_q^2\big),\end{aligned}$$

where $\mathcal{B}_p(t; \mu_p, \sigma_p^2)$ and $\mathcal{B}_q(t; \mu_q, \sigma_q^2)$ are two independent Brownian motion processes with drift, and $\mathcal{I}(\cdot)$ is the indicator function. The process $\mathcal{A}(t; \delta_{pq}, \sigma_{pq}^2, n_p, n_q)$ corresponds to having already observed $\mathcal{B}_p(t; \mu_p, \sigma_p^2)$ and $\mathcal{B}_q(t; \mu_q, \sigma_q^2)$ up to times n_p and n_q, respectively. The process $\mathcal{A}(\cdot)$ can be thought of as a continuous-time counterpart of the process $r[\bar{X}_p(r_p) - \bar{X}_q(r_q)]$ that SEB considers during the elimination step. Let T^* be the first time when $\mathcal{A}(t; \delta_{pq}, \sigma_{pq}^2, n_p, n_q)$ exits the triangular continuation. Then we have the following two lemmas.

Lemma 1. (Fabian, 1974) *For a fixed triangular region defined by a and* λ, *if the triangular-region parameter* $\lambda = \delta/2$ *and* $\delta > 0$, *then*:

$$\Pr\left\{\mathcal{B}\big(T; \delta, \sigma_{pq}^2\big) < 0\right\} = \frac{1}{2}e^{-a\delta/\sigma_{pq}^2}.$$

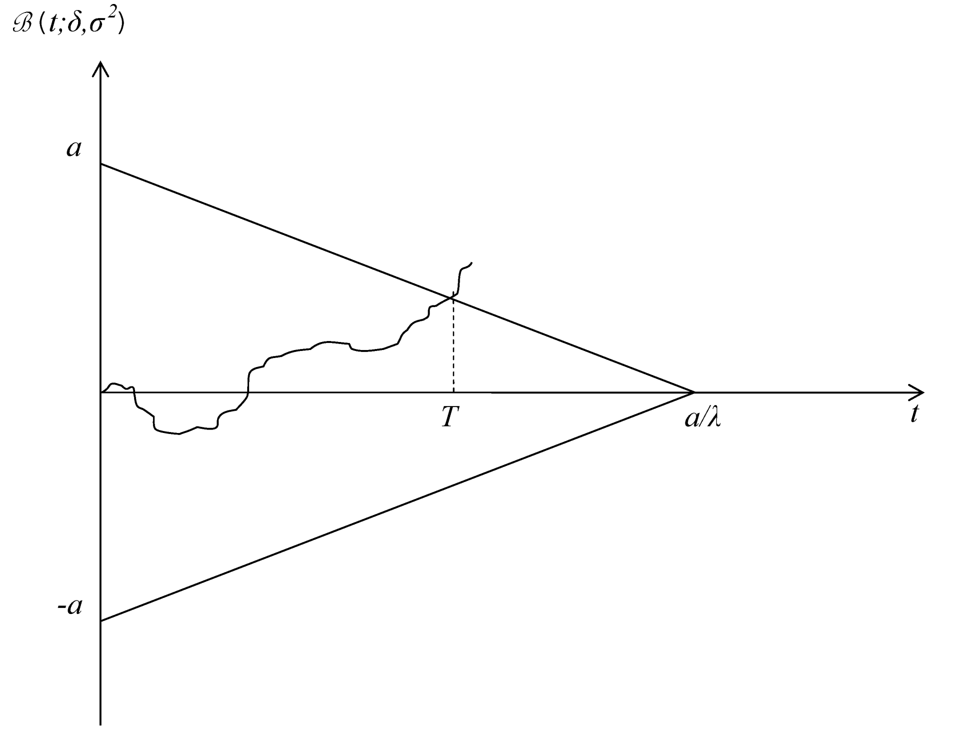

Fig. 1. Triangular continuation region with a drifted Brownian motion process.

Lemma 2. (Pichitlamken *et al.*, 2006) *For a fixed triangular region defined by a and* λ, *if* $\delta_{pq} \ge \delta > 0$, *then for positive* n_p, n_q *and* σ_{pq}^2:

$$\Pr\left\{\mathcal{A}\left(T^*;\delta_{pq},\sigma_{pq}^2,n_p,n_q\right)<0\right\}\le \Pr\left\{\mathcal{B}\left(T;\delta,\sigma_{pq}^2\right)<0\right\}.$$

Remark 2. In both Lemmas 1 and 2 we require that the triangular region is fixed. Notice that in the SEB procedure λ is fixed, but a_{pq} is not fixed. The value of a_{pq} depends on $S_p^2(n_0)$ and $S_q^2(n_0)$. However, if we condition on $S_p^2(n_0)$ and $S_q^2(n_0)$, then a_{pq} is fixed and we can apply Lemmas 1 and 2.

Now we can prove Theorem 1.

Proof of Theorem 1. Let $Y_{pq}(r) = r[\bar{X}_p(r_p) - \bar{X}_q(r_q)]$. Let r_{pq}^* be the first integer time that $Y_{pq}(r)$ exits the triangular continuation region. If systems π_p and π_q are compared and if $\mu_p - \mu_q \ge \delta$, then:

$$\Pr\{\pi_q \text{ eliminates } \pi_p\} = \Pr\{Y_{pq}\left(r_{pq}^*\right)<0\} = E\left[\Pr\left\{Y_{pq}(r_{pq}^*) < 0 \middle| S_p^2(n_0), S_q^2(n_0)\right\}\right].$$

Notice that $Y_{pq}(r)$ is independent of $S_p^2(n_0)$ and $S_q^2(n_0)$ (e.g., Stein (1945)). Therefore, when we condition on $S_p^2(n_0)$ and $S_q^2(n_0)$, the triangular region is fixed and $Y_{pq}(r)$ is independent of the region. By Jennison *et al.* (1980), for a fixed triangular region:

$$\Pr\{Y_{pq}(r_{pq}^*)<0\} \le \Pr\left\{\mathcal{A}\left(T^*;\delta_{pq},\sigma_{pq}^2,n_p,n_q\right)<0\right\}. \quad (4)$$

Then,

$$\begin{aligned}
&\Pr\{\pi_q \text{ eliminates } \pi_p\}\\
&\le E\left[\Pr\left\{\mathcal{A}\left(T^*;\delta_{pq},\sigma_{pq}^2,n_p,n_q\right)<0\middle|S_p^2(n_0),S_q^2(n_0)\right\}\right]\\
&\qquad\qquad\qquad\qquad\text{by Equation (4)}\\
&\le E\left[\Pr\left\{\mathcal{B}\left(T;\delta,\sigma_{pq}^2\right)<0\middle|S_p^2(n_0),S_q^2(n_0)\right\}\right] \text{ by Lemma 2}\\
&= E\left\{\frac{1}{2}\exp\left[-\frac{\eta(n_0-1)\left(S_p^2(n_0)+S_q^2(n_0)\right)}{2\left(\sigma_p^2+\sigma_q^2\right)}\right]\right\} \text{ by Lemma 1}\\
&= \frac{1}{2}E\left\{\exp\left[-\frac{\eta}{2}\frac{\sigma_p^2}{\sigma_p^2+\sigma_q^2}\frac{(n_0-1)S_p^2(n_0)}{\sigma_p^2}\right]\right\}\\
&\quad E\left\{\exp\left[-\frac{\eta}{2}\frac{\sigma_q^2}{\sigma_p^2+\sigma_q^2}\frac{(n_0-1)S_q^2(n_0)}{\sigma_q^2}\right]\right\}. \quad (5)
\end{aligned}$$

Notice that $(n_0-1)S_p^2(n_0)/\sigma_p^2$ and $(n_0-1)S_q^2(n_0)/\sigma_q^2$ are independent $\chi_{n_0-1}^2$ random variables, where χ_f^2 denotes a chi-square random variable with f degrees of freedom, and $\mathrm{E}[e^{t\chi_f^2}] = (1-2t)^{-f/2}$. Then,

$$\begin{aligned}
&\text{Right-hand side of Equation (5)}\\
&= \frac{1}{2}\left\{\left[1+\frac{\eta\sigma_q^2}{\sigma_p^2+\sigma_q^2}\right]\left[1+\frac{\eta\sigma_p^2}{\sigma_p^2+\sigma_q^2}\right]\right\}^{-(n_0-1)/2}\\
&\le \frac{1}{2}(1+\eta)^{-(n_0-1)/2}\\
&= \beta.
\end{aligned}$$

If π_1 is the best system, then the PICS is the largest, since π_1 has to survive the comparisons on all iterations to be selected at the end of iteration i. So in this case, if $\mu_i^* - \mu_i^{**} \ge \delta$, then $\mu_1 - \mu_p \ge \delta$ for all $p = 2, 3, \ldots, K_i$. Thus,

$$\begin{aligned}
&\Pr\{\text{incorrect selection}\}\\
&\le \Pr\{\pi_1 \text{ is eliminated}\}\\
&\le \sum_{p=2}^{K_i}\Pr\{\pi_1 \text{ is eliminated by } \pi_p\} \quad \text{Bonferroni inequality}\\
&\le \sum_{p=2}^{K_i}\beta = \frac{K_i-1}{K-1}\alpha \le \alpha.
\end{aligned}$$

This concludes the proof. ■

One disadvantage of SEB is the requirement of an upper bound on the number of alternatives. Ideally we want the upper bound to be close to the number of alternatives actually evaluated. In many cases, however, such an upper bound is difficult to obtain. Even if it can be estimated, it is often much larger than the actual number. When the bound is larger than the actual number of alternatives, β is smaller than need be, which implies a larger continuation region. Therefore, many more observations than necessary will be taken to make the elimination decisions.

3.2. *Single-elimination unbounded*

To keep the advantages of SEB and avoid the requirement of knowing K, we design a new procedure called Single-Elimination Unbounded (SEU). To do so we require an infinite sequence of positive numbers α_i, $i = 0, 1, \ldots$, such that $\sum_{i=0}^{\infty}\alpha_i = \alpha$. For instance, $\alpha_i = 2^{-(i+1)}\alpha$. SEB requires that the selection decision among $\pi_{K_{i-1}+1}, \pi_{K_{i-1}+2}, \ldots, \pi_{K_i}$ and $\hat{\pi}_{i-1}^*$ on iteration i is correct with probability at least $1-\alpha_i$ if π_i^* is among them and $\mu_i^* - \mu_i^{**} \ge \delta$.

The SEU procedure is the same as the SEB procedure except for the Computing Parameters step. The following is the Computing Parameters step of SEU.

Computing parameters: Let $\beta_i = \alpha_i/k_i$. For any π_p and π_q in I and $p \ne q$, calculate:

$$a_{pq} = \frac{\eta_i(n_0-1)\left[S_p^2(n_0)+S_q^2(n_0)\right]}{2\delta},$$

where

$$\eta_i = (2\beta_i)^{-\frac{2}{n_0-1}} - 1.$$

Theorem 2. *Suppose that assumption* 1 *holds, and the infinite sequence* α_i, $i = 0, 1, \ldots$, *satisfies* $\alpha_i > 0$ *and* $\sum_{i=0}^{\infty}\alpha_i = \alpha$. *Then at the end of any iteration i of SGA*, *SEU guarantees that*:

$$\Pr\left\{\pi_i^* \text{ is selected}\right\} \ge 1-\alpha,$$

if $\mu_i^* - \mu_i^{**} \ge \delta$.

Proof. By the proof of Theorem 1, for any system π_p and on any iteration i, if $\mu_i^* - \mu_p \geq \delta$ and π_i^* and π_p are compared on iteration i, then:

$$\Pr\{\text{system } \pi_p \text{ eliminates system } \pi_i^*\} \leq \beta_i.$$

Similar to the proof of Theorem 1, PICS is largest if π_1 is the best system. So if $\mu_i^* - \mu_i^{**} \geq \delta$, then $\mu_1 - \mu_p \geq \delta$ for all $p = 2, 3, \ldots, K_i$. Thus,

$$\begin{aligned}
&\Pr\{\text{incorrect selection through iteration } i\}\\
&\leq \sum_{j=0}^{i} \Pr\{\pi_1 \text{ is eliminated on iteration } j\}\\
&= \sum_{p=2}^{k_0} \Pr\{\pi_p \text{ eliminates } \pi_1\}\\
&\quad + \sum_{j=1}^{i} \left\{ \sum_{p=K_{j-1}+1}^{K_{j-1}+k_j} \Pr\{\pi_p \text{ eliminates } \pi_1\} \right\}\\
&\leq (k_0 - 1)\beta_0 + \sum_{j=1}^{i} k_j\beta_j \leq \sum_{j=0}^{i} k_j\beta_j \leq \sum_{j=0}^{i} \alpha_j \leq \alpha.
\end{aligned}$$

This concludes the proof, since the result holds for any i. ■

Although SEU does not require an upper bound on the number of alternatives, finding a good scheme to allocate PICS is not easy. For instance, if $\alpha_i = 2^{-(i+1)}\alpha$ is used and α equals 0.1, then after ten iterations (including iteration 0), $\alpha_9 < 0.0001$, which causes the elimination decision to be very difficult to make. A good strategy is to first estimate the number of iterations that SGA may conduct, say I, and allocate a fixed proportion of α, e.g., 0.9α, evenly to the first I iterations. Then we can use a power series, e.g., $2^{-(i-I)}$, to allocate the rest of α among the remaining iterations.

Single-elimination procedures eliminate alternatives permanently. This makes SGA simple and easy to implement. They also provide a statistical guarantee whenever SGA terminates. However, neither single-elimination procedure is efficient in total sample size. In both the iterative system-design and optimization-via-simulation contexts, better systems tend to be discovered on later iterations. Therefore, systems generated by SGA on earlier iterations can easily be eliminated by good systems generated on later iterations. In both single-elimination procedures, however, systems that are generated on the same iteration or consecutive iterations are often used to eliminate each other. Since the alternatives generated on the same iteration or consecutive iterations often are close in mean performance, the procedures may require an excessive number of observations to make the selection decision. We address these shortcomings in the next section.

4. Stop-and-Go procedures

A shortcoming in single-elimination procedures is in the allocation of PICS, specifically that all of PICS is allocated before starting SGA. However, it seems natural to allocate PICS according to the number of alternatives SGA generates. Stated differently, we will let the number of alternatives compared on iteration i be the cumulative number of alternatives generated through iteration i of SGA.

Notice that the boundary of the continuation region for comparing two systems, π_p and π_q, is determined by a_{pq}, which depends on the PICS assigned to the comparison. Given the total PICS, the PICS assigned to each paired comparison is a function of the total number of alternatives through the Bonferroni inequality. One can easily check that, given the first-stage sample variances of the two systems under comparison, the size of the continuation region increases as the total number of alternatives increases. Therefore, if system π_p eliminates system π_q on iteration i, the mean difference process $r[\bar{X}_p(r_p) - \bar{X}_q(r_q)]$ stops at the boundary crossing point, say $r_{pq}^*(i)$. Then on iteration $i+1$, since more alternatives are added to the set of alternatives by SGA, the boundary of the continuation region for comparison between π_p and π_q widens. To confirm the elimination decision between system π_p and π_q, more observations from them may be needed. Therefore, we let the mean difference process take off again from the previous boundary crossing point. The elimination decision is made when the process crosses the new boundary at, say $r_{pq}^*(i+1)$. See Fig. 2 for an illustration. For this reason the procedures presented here are called Stop-and-Go (SaG) procedures.

Suppose SGA tends to generate better systems on later iterations. If system π_p is eliminated by system π_q on iteration i, and a better system, say system π_s, is generated on a later iteration, then π_s can eliminate π_p more easily than π_q can. Therefore, fewer observations are needed for π_p. SaG is in general more efficient than single elimination, even if we know exactly how many alternatives SGA will generate, since better systems generated on later iterations are used to eliminated inferior systems generated on earlier iterations. Compared to single-elimination procedures, SaG procedures also have disadvantages. In SaG procedures, an alternative that has previously been eliminated needs to be considered again whenever a new alternative is generated; this complicates the iterative system-design/optimization-via-simulation process, and it may not be intuitive for users.

4.1. *SaG with fixed first-stage sample size*

If we fix the first-stage sample size n_0, as in the SEB and SEU procedures, then the following SaG-F procedure is called at Step 2 on iteration i of SGA.

SaG with Fixed First-stage Sample Size (SaG-F)

Input: If the iteration counter $i = 0$, then set $I = \{\pi_1, \pi_2, \ldots, \pi_{k_1}\}$ and $K_{-1} = 0$; otherwise, set $I =$

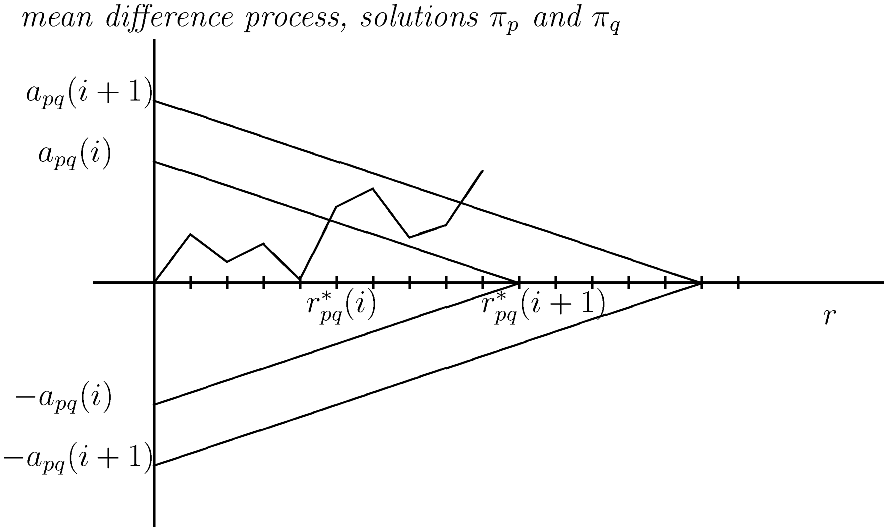

Fig. 2. Mean difference process in the Stop-and-Go procedure.

$\{\pi_1, \pi_2, \ldots, \pi_{K_{i-1}}, \pi_{K_{i-1}+1}, \pi_{K_{i-1}+2}, \ldots, \pi_{K_{i-1}+k_i}\}$, and input the overall sample sizes, overall sample means and first-stage sample variances of systems $\pi_1, \pi_2, \ldots, \pi_{K_{i-1}}$.

Return: System $\hat{\pi}^*_i$ and the overall sample sizes, overall sample means and first-stage sample variances of systems $\pi_1, \pi_2, \ldots, \pi_{K_i}$.

Procedure:

Setup: Select the PCS $1/2 < 1-\alpha < 1$, IZ parameter $\delta > 0$, and first-stage sample size $n_0 \geq 2$. Let $\lambda = \delta/2$.

Initialization: For all $p = K_{i-1}+1, K_{i-1}+2 \ldots, K_i$, take n_0 observations from π_p, calculate the first-stage sample mean $\bar{X}_p(n_0)$ and the first-stage sample variance $S^2_p(n_0)$ using Equation (1), and set $r_p = n_0$. Let $r = n_0$.

Computing parameters: Let $\beta_i = \alpha/(K_i - 1)$. For any π_p and π_q in I and $p \neq q$, calculate:

$$a_{pq} = \frac{\eta_i(n_0-1)\left[S^2_p(n_0) + S^2_q(n_0)\right]}{2\delta},$$

where

$$\eta_i = (2\beta_i)^{-\frac{2}{n_0-1}} - 1.$$

Elimination: Set $I^{\text{old}} = I$, and update I to be

$$I = \{\pi_p : \pi_p \in I^{\text{old}} \text{ and } r[\bar{X}_p(r_p) - \bar{X}_q(r_q)] \geq -\max(0,\ a_{pq} - r\lambda),\ \forall \pi_q \in I^{\text{old}},\ p \neq q\ \}.$$

Stopping rule: If $|I| = 1$, then let $\hat{\pi}^*_i$ be the system in I, and Return. Otherwise, for every $\pi_p \in I$ such that $r_p = r$, take an observation of system π_p, let $r_p = r_p + 1$, and update $\bar{X}_p(r_p)$. Set $r = r + 1$ and go back to Elimination.

Theorem 3. *Suppose that assumption* 1 *holds. Then at the end of iteration i of SGA, SaG-F guarantees that*:

$$\Pr\left\{\pi^*_i \text{ is selected}\right\} \geq 1 - \alpha,$$

if $\mu^*_i - \mu^{**}_i \geq \delta$.

Proof. By the proof of Theorem 1, for any system π_p and on any iteration i, if $\mu^*_i - \mu_p \geq \delta$, then:

$$\Pr\{\pi_p \text{ eliminates } \pi^*_i\} \leq \beta_i.$$

If $\mu^*_i - \mu^{**}_i \geq \delta$, then $\mu^*_i - \mu_p \geq \delta$ for all $\pi_p \in \{\pi_1, \pi_2, \ldots, \pi_{K_i}\} \setminus \{\pi^*_i\}$. Thus,

$$\Pr\{\text{incorrect selection}\} \leq \sum_{\pi_p \in \{\pi_1, \ldots, \pi_{K_i}\} \setminus \{\pi^*_i\}} \Pr\{\pi_p \text{ eliminates } \pi^*_i\} \leq (K_i - 1)\beta_i = \alpha.$$

This concludes the proof of Theorem 3. ■

When SaG-F is applied in SGA, the PICS is allocated according to the current cumulative number of systems. On earlier iterations of SGA, since the total number of generated systems is small, the sizes of the continuation regions are relatively small due to the fixed overall PCS. Therefore, the elimination decisions are relatively easy to make even if the differences between the systems are small. When the total number of the generated systems gets larger on later iterations, the better systems generated on later iterations can eliminate the inferior systems generated on earlier iterations easily. The following analysis shows why SaG-F on average needs fewer observations than the single-elimination procedures.

Let $B(i) = \{p : \pi_p = \pi^*_i\}$. Then $\pi_{B(i)} = \pi^*_i$ on iteration i. For any π_q that is not equal to $\pi_{B(i)}$, the probability of incorrect selection, β_i, between $\pi_{B(i)}$ and π_q goes to zero as $K_i \to \infty$. If $\beta_i \to 0$, then $a_{B(i)q} \to \infty$. The number of observations required to make an elimination decision also

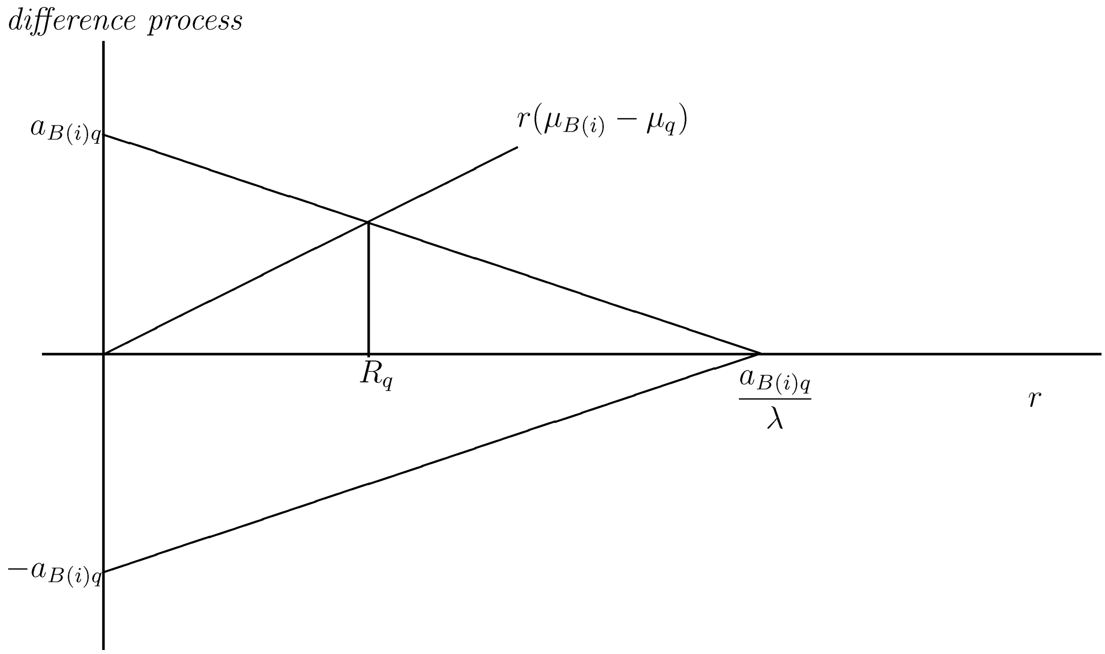

Fig. 3. Optimistic analysis of $\pi_{B(i)}$ vs. π_q.

goes to infinity. Therefore, when the number of systems generated by SGA is large (K_i is large) we can approximate the difference process as $r(\mu_{B(i)} - \mu_q)$ by the Strong Law of Large Numbers (see Fig. 3), and approximate $E[R_q]$, the expected number of observations from π_q that are required to eliminate π_q, as

$$E[R_q] \approx \frac{E[a_{B(i)q}]}{\lambda + \mu_{B(i)} - \mu_q} = \frac{E[a_{B(i)q}]}{\delta/2 + \mu_{B(i)} - \mu_q}. \qquad (6)$$

Since $\delta/2$ is usually small compared to the difference between mean values, $E[R_q]$ is approximately inversely proportional to $\mu_{B(i)} - \mu_q$. Therefore, when better systems (higher $\mu_{B(i)}$) are generated on later iterations of SGA, the number of observations required to eliminate an inferior system can be significantly reduced. The observations obtained on early iterations for a system (say π_q) may be enough for it to be eliminated by a good system generated on later iterations even though the PICS allocated to the comparison is smaller due to the larger number of systems. This explains why the SaG procedures are on average more efficient than the single-elimination procedures.

The disadvantage of SaG-F is that it may need to go back to a large number of previously simulated systems on each iteration to acquire more observations. This leads to high switching costs between simulations of different systems. Switching between complicated simulation models can be computationally cumbersome. For an example of controlling switching cost in R&S, see Hong and Nelson (2005).

4.2. *SaG with variable first-stage sample size*

SaG-F uses a fixed first-stage sample size, n_0, for all systems. At the beginning of iteration i, systems $\pi_1, \pi_2, \ldots, \pi_{K_{i-1}}$ usually have more than n_0 observations. Since a larger n_0 often results in a smaller continuation region because of the greater degrees of freedom of the variance estimators, one may ask: why we do not treat all the observations acquired before iteration i as the first-stage samples for the purpose of computing variance estimates? The answer is that if we do this, then the numbers of observations obtained on previous iterations depend on the variance estimators. If we use the observations as the first-stage samples, then the independence between the overall sample mean and the first-stage sample variance is destroyed. To prove the validity of the SaG procedures, we have to maintain the independence between the variance estimator and the overall sample mean.

If we let n_0 be a function $n_0(K_i)$, where K_i is the total number of systems generated through the ith iteration, and choose the appropriate $n_0(K_i)$ before running SGA, then the independence property still holds. Now at Step 2 of each iteration of SGA, one first updates $n_0(K_i)$ based on the total number of the generated alternatives. If the previously generated systems have more than $n_0(K_i)$ observations, then only the first $n_0(K_i)$ observations are considered as the first-stage samples in the variance calculation. If the previously generated systems have fewer than $n_0(K_i)$ observations, then we take additional observations from the systems to get $n_0(K_i)$ observations. We then take $n_0(K_i)$ observations for the newly generated systems, and run an SaG procedure. The following is the new SaG procedure.

SaG procedure with variable first-stage sample size (SaG-V)

Input: Systems $\pi_1, \pi_2, \ldots, \pi_{K_i}$ and all sample information on systems $\pi_1, \pi_2, \ldots, \pi_{K_{i-1}}$. If iteration counter i equals zero, let $K_{-1} = 0$ and let the identities and all sample information on previously generated systems be null.

Return: System $\hat{\pi}_i^*$ and all the sample information on systems $\pi_1, \pi_2, \ldots, \pi_{K_i}$.

Procedure:

Setup: Select the PCS $1/2 < 1 - \alpha < 1$, IZ parameter $\delta > 0$ and first-stage sample size function $n_0(\cdot)$. Let $\lambda = \delta/2$.

Initialization: Take $\max\{0, n_0(K_i) - r_p\}$ observations from π_p and set $r_p = r_p + \max\{0, n_0(K_i) - r_p\}$ for all $p = 1, 2, \ldots, K_{i-1}$; also take $n_0(K_i)$ observations from π_p and set $r_p = n_0(K_i)$ for all $p = K_{i-1} + 1, \ldots, K_i$. Calculate the first-stage sample mean $\bar{X}_p(n_0(K_i))$ and the first-stage sample variance $S_p^2(n_0(K_i))$ using Equation (1) for all $p = 1, 2, \ldots, K_i$. Let $r = n_0(K_i)$ and $I = \{\pi_1, \pi_2, \ldots, \pi_{K_i}\}$.

Computing parameters: Same as the step in SaG-F, but using $n_0(K_i)$ instead of n_0.

Elimination: Same as the step in SaG-F.

Stopping rule: Same as the step in SaG-F.

Theorem 4. *Suppose that assumption* 1 *holds, then at the end of iteration i of SGA, SaG-V guarantees that*:

$$\Pr\left\{\pi_i^* \text{ is selected}\right\} \geq 1 - \alpha,$$

if $\mu_i^* - \mu_i^{**} \geq \delta$.

Proof. The proof of Theorem 4 is the same as the proof of Theorem 3, except that we change n_0 to $n_0(K_i)$. ■

One natural question when using the SaG-F procedure is how to choose $n_0(k)$, where k is the number of systems in comparison. Ideally, we want to choose $n_0(k)$ to minimize the expected total number of observations needed to make the selection decision. However, this is a difficult, if not impossible, task since it requires knowing the true means and variances of all generated alternatives before starting the experiment. Instead we try to find a family of functions $n_0(k)$ to minimize the growth rate (as $k \to \infty$) of an upper bound on the expected sample size. In the following analysis, we assume that there exists a finite positive number M such that $\sigma_p^2 \leq M$ for all $p = 1, 2, \ldots$.

If systems π_p and π_q are in comparison, then the maximum number of observations required for both of them is $\lceil a_{pq}/\lambda \rceil$, which is the end of the triangular continuation region, if it is not smaller than n_0. At the end of the ith iteration, if $E[a_{pq}/\lambda] \geq n_0$, then

$$\eta_i = \left(\frac{K_i - 1}{2\alpha}\right)^{2/(n_0(K_i)-1)} - 1,$$

$$E\left[\frac{a_{pq}}{\lambda}\right] = E\left\{\frac{\eta_i(n_0(K_i) - 1)\left[S_p^2(n_0(K_i)) + S_q^2(n_0(K_i))\right]}{2\delta\lambda}\right\}$$
$$\leq \frac{2M}{\delta^2}[n_0(K_i) - 1]\left[\left(\frac{K_i - 1}{2\alpha}\right)^{2/(n_0(K_i)-1)} - 1\right].$$

Since $a_{pq}/\lambda \to \infty$ as $K_i \to \infty$, the difference between $\lceil a_{pq}/\lambda \rceil$ and a_{pq}/λ can be omitted. Therefore, the expected total sample size of SaG-V at the end of ith iteration, $E[R(K_i)]$, is bounded by

$$E[R(K_i)] \leq K_i \max_{p,q \in \{1,2,\ldots,K_i\}} E\left[\frac{a_{pq}}{\lambda}\right]$$
$$\leq \frac{2M}{\delta^2} K_i[n_0(K_i) - 1]\left[\left(\frac{K_i - 1}{\alpha}\right)^{2/(n_0(K_i)-1)} - 1\right]. \tag{7}$$

Since $x^{1/\log(x)} = e$, the Right-Hand Side (RHS) of Equation (7) equals:

$$\frac{2M}{\delta^2} K_i[n_0(K_i) - 1]\left[e^{2\log((K_i-1)/2\alpha)/[n_0(K_i)-1]} - 1\right],$$

which increases to infinity at the same rate as

$$K_i n_0(K_i)\left[e^{\log(K_i)/n_0(K_i)} - 1\right],$$

as i goes to infinity (implying $K_i \to \infty$). To avoid oversampling we want $n_0(K_i)$ to satisfy $\lim_{K_i \to \infty} K_i n_0(K_i)/E[R(K_i)] < \infty$, which means that the first-stage sample size should not dominate the expected total sample size in the growth rate. Therefore, we need $\lim_{K_i \to \infty} \log(K_i)/n_0(K_i) > 0$. Let $n_0(K_i) = \log(K_i)/f(K_i)$, where $f(K_i) > 0$ for all $K_i > 0$ and $\lim_{K_i \to \infty} f(K_i) > 0$. Then the RHS of Equation (7) grows at the same rate as

$$\frac{1}{f(K_i)} e^{f(K_i)} K_i \log(k_i).$$

Therefore, the RHS of Equation (7) achieves the minimum growth rate when $f(K_i)$ is a constant. Thus $n_0(K_i)$ grows at the rate of $\log(K_i)$ and $E[N(K_i)]$ grows at the rate of $K_i \log(K_i)$ (often written as $k \log((k-1)/\alpha)$ in the R&S literature where k denotes the total number of alternatives), which is an efficient growth rate common to many IZ selection procedures with known variances (see Dudewicz (1969)). An example of $n_0(k)$ that satisfies these conditions is

$$n_0(k) = n_0\lceil \log(k) \rceil \quad k = 2, 3, \ldots. \tag{8}$$

The implementation of SaG-V is more complicated than SaG-F. In SaG-F, only the first-stage sample variances and the current sample means need to be stored. But in SaG-V the first-stage sample size is increasing and it typically does not equal to the current sample sizes of the systems in comparison. Therefore, to implement SaG-V, the previous samples need to be stored such that the first-stage sample variances of all systems can be calculated given a first-stage sample size.

5. Empirical evaluation

In this section we give two examples to compare different procedures and to illustrate the usefulness of the proposed procedures. In the first example, we mimic the iterative system-design/optimization-via-simulation context and compare the performances of the proposed procedures. We also compare the procedures to a fully sequential procedure assuming that the set of evaluated alternatives is known in advance. In the second example, we give a realistic iterative system-design/optimization-via-simulation problem and show the advantages of our procedures.

5.1. *A simple test problem*

In both iterative system design and optimization via simulation, we expect to find better solutions on every iteration. But we will also generate some not-so-good solutions on every iteration. As the number of iterations increases, the improvement in solutions may diminish, and the solutions that are generated on each iteration may get closer to each other in terms of mean performance. Moreover, several more iterations are often needed after finding the optimal solution to conclude that the optimal solution has been found.

To incorporate all these considerations, we designed the following test problem. Consider the functions:

$$\begin{aligned} y_1(x) &= -|x-16|^{1.25} + 68, \\ y_2(x) &= -|x-16|^{1.5} + 65, \\ y_3(x) &= \frac{1}{3}y_1(x) + \frac{2}{3}y_2(x), \\ y_4(x) &= \frac{2}{3}y_1(x) + \frac{1}{3}y_2(x), \end{aligned}$$

where $0 \le x \le 20$. Suppose that we start with $x = 0$ and four initial systems each normally distributed with mean $y_p(0)$ and standard deviation $0.1y_p(0)$, $p = 1, 2, 3, 4$. Also suppose that on iteration i, we have $x = i \times \Delta$, where $\Delta > 0$, and four new systems, each normally distributed with mean $y_p(x)$ and standard deviation $0.1y_p(x)$, $p = 1, 2, 3, 4$, are generated (see Fig. 4). The goal is to select the system with the largest mean performance when we stop at $x = 20$. This example mimics the iterative system-design/optimization-via-simulation process in the way that systems are generated on each iteration. What is missing here is the search process, in which the quality of the systems generated on an iteration depends on the quality of the selection decisions made on the previous iterations.

In this example, we let the IZ parameter be $\delta = 1$ so that the difference in the mean performance between the best system and the second-best system is exactly δ. We let the first-stage sample size be $n_0 = 10$ and a PCS value of 0.9.

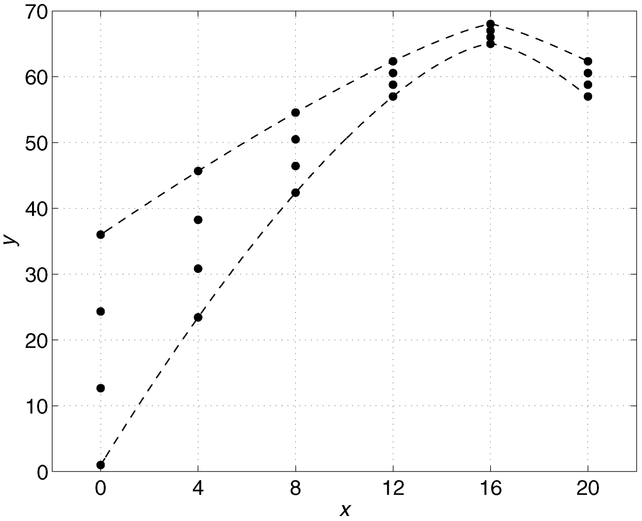

Fig. 4. The test problem used in Section 5.1 with $\Delta = 4$.

We consider three configurations where $\Delta = 4$, 2 or 1. The total numbers of alternatives (K) of the three configurations are 24, 44 and 84, respectively.

The results are shown in Table 1. KN is the fully sequential procedure from Kim and Nelson (2001). KN is used only as a benchmark for comparison; it cannot be applied to this problem because it requires all systems to be available from the beginning. SEB1 is the SEB procedure with the bound equal to exactly the total number of systems, and SEB2 is the SEB procedure with the bound equal to 100. In SEU we use $\alpha_i = 0.025(4/5)^{i+1}$, and in SaG-V we use $n_0(k) = 10\lceil \log_2(k/2) \rceil$. The results reported in the table are the averages over 100 macroreplications of the entire experiment.

We see from Table 1 that the SaG procedures consume fewer observations than the single-elimination procedures. When the total number of systems that SGA will visit is known, SEB works better than SEU; however, when the total number of systems is not known and we only have a large bound, then SEB may not be as good as SEU. When the total number of systems that SGA will visit is large, SEU may be less efficient since the PICS allocated to the later iterations can be very small. For the SaG procedures, SaG-V is more efficient than SaG-F as we expect. We also see that the differences between KN and SaG-V are not very

Table 1. Observed PCS and average total number of observations

	Δ = 4/K = 24		*Δ = 2/K = 44*		*Δ = 1/K = 84*	
Procedures	*PCS*	*Avg. #*	*PCS*	*Avg. #*	*PCS*	*Avg. #*
KN	0.99	2285.1	1.00	4213.7	1.00	8790.5
SEB1	1.00	4019.7	1.00	9349.8	1.00	25 393.5
SEB2	1.00	6309.6	1.00	11 955.2	1.00	26 885.7
SEU	1.00	4977.7	1.00	11 934.5	1.00	40 301.8
SaG-F	1.00	3753.5	1.00	8 562.7	1.00	22 720.0
SaG-V	1.00	2585.7	1.00	5 384.5	1.00	13 373.8

large when the total number of alternatives SGA generates is not large.

Table 2. The routings for various job types

Job type	*Work station routing*
1	3, 1, 2, 5
2	4, 1, 3
3	2, 5, 1, 4, 3

5.2. *A job shop example*

In this section we study a job-shop improvement problem (see Section 2.7.1 of Law and Kelton (2000) for the simulation model). The job shop consists of five work stations, and at present stations $1, 2, \ldots, 5$ consist of three, two, four, three and one parallel machines, respectively. Assume that jobs arrive to the system with interarrival times that are independent and identically distributed exponential random variables with mean 0.25 hour. There are three types of jobs, and arriving jobs are of type 1, 2 and 3 with respective probabilities 0.3, 0.5 and 0.2. Job types 1, 2 and 3 require four, three and five tasks to be done, respectively, and each task must be done at a specified station and in a prescribed order. The routings for the different job types are as listed in Table 2.

If a job arrives at a particular station and finds all machines in that station already busy, the job joins a single first-in-first out queue at the station. The service times are Erlang-2 random variables whose mean depends on the job type and the station to which the machine belongs, and they are independent. The mean service times for each job type and each task are as listed in Table 3.

Given these parameters we build a simulation model to estimate the long-run mean job waiting time, regardless of job type. For each replication, we start with the empty job shop and simulate it for 365 days. The length of the simulaton is long enough to overcome the initial-condition bias and to ensure that the average waiting time is approximately normally distributed.

Suppose that the owner of the job shop is concerned that the job waiting times are too long and is considering improving the performance of the shop. Each machine costs approximately \$200 000 and the budget for improvement is \$1000 000. The owner estimates that 1 hour of waiting time is equivalent to a present value of \$100 000 of loss of goodwill or potential sales in the future, and wants to improve the performance of the job shop to minimize the expected total cost, including both machine cost and waiting-time cost.

Table 3. The mean service times for the various job types

Job type	*Mean service time for successive tasks (hours)*
1	0.50, 0.60, 0.85, 0.50
2	1.10, 0.80, 0.75
3	1.20, 0.25, 0.70, 0.90, 1.00

We can solve this problem using (at least) three different approaches. The first is to enumerate all system designs and use KN to select the system with the lowest cost. For this approach, we need to consider the cases where up to five machines can be purchased; this yields 256 potential designs. We ran KN with confidence level $1 - \alpha = 0.9$ and an IZ parameter $\delta = \$200\ 000$, which is the cost of one machine or the present value of 2 hours of waiting time. At the end of the process, the optimal solution was to add one machine to work stations 1, 2 and 4, respectively.

The second approach is to use a myopic optimization-via-simulation algorithm. In this approach, we add one machine at a time. In each iteration, we simulate all five options of adding an additional machine to each workstation of the best design from the previous iteration, compare these new options to the previous best design to determine where to add the machine, and stop if the best design on the current iteration is same as the best design on the previous iteration. When using this algorithm, we could simulate at most 26 designs; that is, the initial design with five new designs on each of the five possible iterations. So we set $K = 26$, $\delta = \$200\ 000$ and $1 - \alpha = 0.9$ and ran the optimization in conjunction with the SEB procedure. After the first iteration, the best option was to add a machine to work station 2; on the second iteration, the best alternative was to add another machine to work station 4; while on the third iteration, the best choice was to add another machine to work station 1. On the fourth iteration, the best design was still the best design from the third iteration, so the algorithm stopped and did not perform a fifth iteration. The optimal design was same as the design found by the first approach, but only 21 designs were evaluated as compared to the 256 designs of using the KN procedure, a huge saving.

In the third approach, the designers are involved in decision making, and they use SEB with the same parameters as the second approach. They use the myopic algorithm on the first iteration, but then select the system designs to evaluate on the subsequent iterations based on the results they have seen and stop when they believe that an optimal design has been found. The following is one realization of this approach. After the first iteration, adding one machine to work station 2 yielded the best design with average cost \$1160 000, and adding one machine to work stations 4, 1, 3 and 5 had average total costs of \$1440 000, \$1680 000, \$1800 000 and \$1990 000, respectively. Therefore, on the second iteration they evaluated two new designs: adding a total of two machines to work station 2, and adding one machine to each of work stations 2 and 4. The latter design was the best on this iteration with an average total cost of \$870 000. On the third iteration, they evaluated three new designs: adding a total of two machines to work station 2 and one machine to work station 4; adding one machine to work station 2 and two machines to work station 4; and adding one machine to each of work stations 2, 4 and 1. The third design was the best on this iteration with an average total cost of $750 000. Notice that adding a fourth or a fifth machine requires a total investment of $800 000 or $1000 000, then the total expected cost will be at least $800 000 or $1000 000. Therefore, better solutions cannot be found by adding a fourth or fifth machine. Then the designers stopped and accepted the current design as the best. In this approach, only 11 designs were evaluated, demonstrating the value of using human intelligence to guide the search with SEB to help guarantee that the best design is actually retained on each iteration they made.

This example demonstrates the usefulness of having R&S procedures for situations where systems are revealed sequentially.

6. Conclusions and future research

In this paper we present two approaches, the single-elimination approach and the SaG approach to solve R&S problems where the systems are revealed sequentially during the experiment. The single-elimination procedures, including SEB and SEU, are easier to understand and implement, but they often require a large number of observations to provide the statistical guarantee. The SaG procedures, including SaG-F and SaG-V, are more difficult to implement, but they typically require fewer observations. The empirical evaluation shows that the proposed procedures are useful in solving practical problems.

In terms of future research we suggest to consider the following two possible directions. First, one may consider to use the variance-dependent sampling of Hong (2006) to further improve the overall efficiency of the procedures proposed in this paper. Second, the design of efficient selection procedures for large-scale optimization-via-simulation algorithms is certainly an interesting and important research issue that deserves further study.

Acknowledgements

The authors thank Dr. Sigrún Andradóttir of Georgia Tech for her ideas and assistance leading to the development of the SEU procedure, and the department editor and two referees for their insightful comments that significantly improved the quality of this paper. This research was partially supported by National Science Foundation grant number DMI-0217690, Hong Kong Research Grants Council grant number CERG 613305, and General Motors R&D.

References

Alrefaei, M.H. and Andradóttir, S. (1999) A simulated annealing algorithm with constant temperature for discrete stochastic optimization. *Management Science*, **45**, 748–764.

Bechhofer, R.E., Santner, T.J. and Goldsman, D.M. (1995) *Design and Analysis of Experiments for Statistical Selection, Screening, and Multiple Comparisons*, Wiley, New York, NY.

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

Chen, C.H., Lin, J., Yücesan, E. and Chick, S.E. (2000) Simulation budget allocation for further enhancing the efficiency of ordinal optimization. *Discrete Event Dynamic Systems: Theory & Applications*, **10**, 251–270.

Chick, S.E. and Inoue, K. (2001a) New two-stage and sequential procedures for selecting the best simulated system. *Operations Research*, **49**, 732–743.

Chick, S.E. and Inoue, K. (2001b) New procedures to select the best simulated system using common random numbers. *Management Science*, **47**, 1133–1149.

Dudewicz, E.J. (1969) An approximation to the sample size in selection problems. *The Annals of Mathematical Statistics*, **40**, 492–497.

Fabian, V. (1974) Note on Anderson's sequential procedures with triangular boundary. *Annals of Statistics*, **2**, 170–176.

Fu, M.C. (2002) Optimization for simulation: theory vs. practice. *INFORMS Journal on Computing*, **14**, 192–215.

Hong, L.J. (2006) Fully sequential indifference-zone selection procedures with variance-dependent sampling. *Naval Research Logistics*, **53**, 464–476.

Hong, L.J. and Nelson, B.L. (2005) The trade-off between sampling and switching: new sequential procedures for indifference-zone selection. *IIE Transactions*, **37**, 623–634.

Hong, L.J. and Nelson, B.L. (2006) Discrete optimization via simulation using COMPASS. *Operations Research*, **54**, 115–129.

Jennison, C., Johnston, I.M. and Turnbull, B.W. (1980) Asymptotically optimal procedures for sequential adaptive selection of the best of several normal means. Technical report, Department of Operations Research and Industrial Engineering, Cornell University, Ithaca, NY 14853.

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

Law, A.M. and Kelton, W.D. (2000) *Simulation Modeling and Analysis*, 3rd edn., McGraw-Hill, New York, NY.

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

Rinott, Y. (1978) On two-stage selection procedures and related probability-inequalities. *Communications in Statistics*, **A7**, 799–811.

Shi, L. and Ólafsson, S. (2000) Nested partitions method for stochastic optimization. *Methodology and Computing in Applied Probability*, **2**, 271–291.

Stein, C. (1945) A two-sample test for a linear hypothesis whose power is independent of the variance. *Annals of Mathematical Statistics*, **16**, 243–258.

Biographies

L. Jeff Hong is an Assistant Professor in the Department of Industrial Engineering and Logistics Management at the Hong Kong University of Science and Technology. He obtained his Ph.D. in Industrial Engineering and Management Sciences from Northwestern University. His research interests include design and analysis of simulation experiments, optimization via simulation and robust optimization. He is currently an Associate Editor of *Naval Research Logistics*.

Barry L. Nelson is the Charles Deering McCormick Professor in the Department of Industrial Engineering and Management Sciences at Northwestern University, and is the Editor in Chief of *Naval Research Logistics*. He is interested in the design and analysis of computer simulation experiments, particularly statistical efficiency, multivariate output analysis, and input modeling. He is a Fellow of the Institute for Operations Research and the Management Sciences.