

L. Jeff HONG, Weiwei FAN, Jun LUO

Review on ranking and selection: A new perspective

© The Author(s) 2021. This article is published with open access at link.springer.com and journal.hep.com.cn

Abstract In this paper, we briefly review the development of ranking and selection (R&S) in the past 70 years, especially the theoretical achievements and practical applications in the past 20 years. Different from the frequentist and Bayesian classifications adopted by Kim and Nelson (2006b) and Chick (2006) in their review articles, we categorize existing R&S procedures into fixed-precision and fixed-budget procedures, as in Hunter and Nelson (2017). We show that these two categories of procedures essentially differ in the underlying methodological formulations, i.e., they are built on hypothesis testing and dynamic programming, respectively. In light of this variation, we review in detail some well-known procedures in the literature and show how they fit into these two formulations. In addition, we discuss the use of R&S procedures in solving various practical problems and propose what we think are the important research questions in the field.

Keywords ranking and selection, hypothesis testing, dynamic programming, simulation

Received August 1, 2020; accepted November 22, 2020

L. Jeff HONG
School of Management and School of Data Science, Fudan University,
Shanghai 200433, China

Weiwei FAN (✉)
Advanced Institute of Business, Tongji University, Shanghai 200092,
China
E-mail: wfan@tongji.edu.cn

Jun LUO (✉)
Antai College of Economics and Management, Shanghai Jiao Tong
University, Shanghai 200030, China
E-mail: jluo_ms@sjtu.edu.cn

This research was supported in part by the National Natural Science Foundation of China (Grant Nos. 71991473, 71701196, 71722006, and 72031006).

1 Introduction

Decision-making processes often involve comparisons among a set of alternatives regarding certain performance measures. In this study, we consider such comparison problems with the goal of selecting the best alternative, where the best is defined to have the largest (or smallest) mean performance. This aspect is not trivial in the stochastic environment where the mean performances of these alternatives are unknown and have to be inferred via statistical sampling from stochastic systems. Therefore, a selection procedure is required to determine how many samples need to be collected from each alternative and then which alternative should be selected as the best based on the sample information. Such selection problems are often called ranking and selection (R&S) in the literature.

R&S problems date back to the 1950s in agricultural and clinical applications (Bechhofer, 1954; Gupta, 1956). At that time, testing the homogeneity of multiple alternatives was common (e.g., grain yields and drug treatments). For instance, an individual might desire to test whether multiple grains produced the same mean yield or whether multiple drug treatments led to the same mean efficacy. Once the homogeneity of their means was rejected statistically, a natural issue readily arose, that is, which one was the best. This issue was first proposed by Paulson (1949) and triggered the early developments of R&S.

In the 1950s, samples needed to be collected through physical experiments, e.g., agricultural experiments and clinical trials, which might cost a long time to conduct. Thus, the experiments were often conducted in batches. Accordingly, a considerable number of the R&S procedures designed then were stage-wise, where the best one was selected at the end of the last stage. Starting in the 1990s, this paradigm began to change owing to the increasing computing power. An increasing number of experiments were conducted in computer simulation environments because it cost little time to generate samples. Through these simulations, samples were often collected sequentially, especially when the program was executed in a single-processor environment. This sequential nature boosted the development of sequential R&S

procedures. Unlike stage-wise procedures, sequential procedures typically provide a decision rule at each time of the sample collection process and are therefore more efficient in most situations by taking advantage of the interim sample information. Sequential R&S procedures are still prevalent today.

In recent years, another forming paradigm that considers large-scale R&S problems has emerged. For early applications, such as agricultural problems, the number of alternatives was relatively small. Designed for these applications, classic procedures were typically applied to problems with fewer than 500 alternatives. However, in the modern world, we often face problems that may have thousands to tens of thousands of alternatives. For instance, in scheduling problems, one may need to determine multiple components simultaneously, such as the jobs to be scheduled, the values assigned to the jobs, and the time when the scheduling happens. Assuming that 50 choices are available for each component, their combination fairly leads to a total of 125000 alternatives, which is a huge number for classic R&S procedures. Recently, research on how to handle large-scale R&S problems has drawn significant attention. As pioneer works, Luo and Hong (2011), Luo et al. (2015), and Ni et al. (2013; 2017) addressed large-scale problems by adapting the classic procedures into parallel computing environments.

Interested readers may refer to Fu and Henderson (2017) for introduction on the history of R&S. Basically, R&S procedures provide a general tool for solving selection problems. Therefore, they are widely applicable to practical problems. Besides, many of the R&S procedures are also easy to implement, and some of them have been embedded in commercial simulation software packages, such as Arena and Simio.

To organize the R&S procedures, existing review articles often categorize them into frequentist and Bayesian procedures according to the probability models used to describe the collected samples (Chick, 2006; Kim and Nelson, 2006b; Branke et al., 2007). In this work, we take a different perspective and categorize them into fixed-precision and fixed-budget procedures, as in Gabillon et al. (2012) and Hunter and Nelson (2017). Particularly, fixed-precision procedures intend to provide a desired statistical guarantee of the selected alternative being the best (or at least close to the best), while fixed-budget procedures intend to allocate a given sampling budget in various optimal or approximately optimal ways. To explain these two categories of procedures, we show that they essentially follow two different formulations, i.e., the hypothesis-testing and dynamic-programming formulations, respectively. A number of studies in the literature have adopted the same perspective and designed new procedures under the two formulations (Batur and Choobineh, 2012; Peng et al., 2018). Different from

these works, the goal of this review is to construct a unified framework for each formulation and explain how the existing procedures fit in the framework.

This paper only focuses on selecting the best mean. However, some related problems may also be categorized into R&S problems. They essentially have different combinations of goals to achieve and performance measures used for comparisons. For instance, the goals can be ranking all the alternatives, or selecting the top m alternatives, or selecting a subset of alternatives that contains the best. Meanwhile, the performance measures used can be quantile or proportion. These problems are not covered in this study, and interested readers may refer to Bechhofer et al. (1995), Goldsman et al. (1998) and Kim and Nelson (2006b) for comprehensive reviews.

One problem closely related to R&S is the multi-armed bandit (MAB) problem in the machine learning literature. Both problems stemmed from Bechhofer (1954) and Paulson (1964), and they have grown into two branches of research with different goals in designing procedures. R&S procedures typically attempt to optimize the quality of the final selection. In contrast, MAB procedures attempt to balance the tradeoff between exploration (gathering new information on different alternatives) and exploitation (choosing the best alternative) in the sequential sampling process. Therefore, the MAB problem often aims to minimize cumulative regret during the sampling process. Nonetheless, a series of works have considered the pure-exploration version of the MAB problem, which is known as the best-arm identification (BAI) problem (Bubeck and Cesa-Bianchi, 2012). Although BAI and R&S problems have the same goal, they typically make different assumptions on the samples from alternatives. Particularly, the BAI problem assumes the samples to be bounded or sub-Gaussian distributed, whereas the R&S problem typically assumes they are Gaussian distributed with unknown variances. In this study, we will not review the MAB procedures. Interested readers may refer to Even-Dar et al. (2002), Bubeck and Cesa-Bianchi (2012), Gabillon et al. (2012), and Kaufmann and Kalyanakrishnan (2013) for more information on MAB and Ma and Henderson (2017) and Glynn and Juneja (2018) for their connections to R&S procedures.

The rest of paper is organized as follows. In Section 2, we provide a comprehensive description on how R&S problems under fixed-precision and fixed-budget are formulated as hypothesis-testing or dynamic-programming problems, respectively. In Sections 3 and 4, we present several well-known fixed-precision and fixed-budget R&S procedures and explain how they can be derived under two different formulations, respectively. In Section 5, we present the procedures designed for solving large-scale R&S problems. In Section 6, we introduce several emerging R&S problems, followed by the discussion of some interesting future research directions in Section 7.

2 Two formulations for R&S

Suppose that there are $k \geq 2$ alternatives with mean performance $\boldsymbol{\mu} = (\mu_1, \mu_2, \dots, \mu_k)$, and the best alternative is defined to have the largest mean. For simplicity, we assume that the best alternative is unique. The goal of R&S is to select the index of the best alternative, which is unknown a priori. If multiple alternatives have tied best means, choosing any of these alternatives as the best can be viewed as a correct selection.

Evidently, the selection decision should be made based on the information collected from samples. Ideally, we hope to select the best alternative with 100% probability. However, this is impossible unless infinite samples can be collected. Therefore, a tradeoff exists between the sampling budget and the precision of the selection decision. To alleviate this tradeoff, R&S problems are often imposed with two constraints: Fixed precision and fixed budget (Hunter and Nelson, 2017). In particular, the fixed-precision R&S problems intend to achieve a fixed precision of selection when using as few sampling budget as possible, while the fixed-budget R&S problems intend to optimize the precision of the selection given a fixed sampling budget.

In this section, we show that these R&S problems under the two constraints can be formulated as hypothesis-testing (HT) and dynamic-programming (DP) problems, respectively. We also illustrate some key issues in designing corresponding R&S procedures.

2.1 Fixed-precision R&S

To describe the precision of a selection (i.e., the first constraint), one common way is to use the probability that the selected alternative is the true best, which is called the probability of correct selection (PCS). Then, under a fixed precision $1 - \alpha$ ($0 < \alpha < 1 - 1/k$), the goal of R&S is to deliver a PCS guarantee as,

$$\text{PCS}(\boldsymbol{\mu}) = \text{P}\{\text{Select the best alternative} \mid \boldsymbol{\mu}\} \geq 1 - \alpha, \quad \forall \boldsymbol{\mu} \in \Theta, \quad (1)$$

where $\Theta = \{\boldsymbol{\mu} : \mu_{[k]} > \mu_{[k-1]}\}$ and $\mu_{[k]} > \mu_{[k-1]} \geq \dots \geq \mu_{[1]}$ denote the ordered means.

2.1.1 Fixed-precision R&S formulated as hypothesis-testing

Practically, any alternative may be selected as the best. Then, the alternative must be assessed to determine whether it is truly the best. This assessment suffices to detect, for any alternative j , whether it has a larger mean than all the others, i.e., $\mu_j > \mu_i$ for any $i \neq j$. Then, R&S problems essentially involve k simultaneous HTs and are therefore formulated as a multiple HT problem,

$$(HT_j) H_0^j : \mu_j \leq \max_{i \neq j} \mu_i \text{ versus}$$

$$H_1^j : \mu_j > \max_{i \neq j} \mu_i, \quad \forall j = 1, 2, \dots, k. \quad (2)$$

Each single HT_j above regards the comparison between alternative j and all the others.

When H_0^j is rejected, alternative j should be selected as the best. Therefore, to select the best alternative correctly, we only need to avoid committing the Type II error for each HT_j . To make it clear, notice that the PCS guarantee in Eq. (1) can be rewritten as,

$$\begin{aligned} \text{PCS}(\boldsymbol{\mu}) &= \text{P}\{\text{Reject } H_0^j \mid \boldsymbol{\mu} \in H_1^j\} \\ &= 1 - \text{P}\{\text{Type II error in } HT_j\} \geq 1 - \alpha, \\ &\quad \text{for } \boldsymbol{\mu} \in H_1^j, \quad \forall j. \end{aligned}$$

(For simplicity of the notation, we write $\boldsymbol{\mu} \in H_d^j$ ($d = 0, 1$) if $\boldsymbol{\mu}$ satisfies the corresponding hypothesis) This implies that we only need to control the Type II error for all HT_j in Eq. (2) as,

$$\begin{aligned} \text{P}\{\text{Type II error in } HT_j\} &\leq \alpha, \\ \forall \boldsymbol{\mu} \in H_1^j, \quad j &= 1, 2, \dots, k. \end{aligned} \quad (3)$$

The Type I error for each HT_j has been automatically controlled at the same time. Taking the special case when there are only two alternatives for example, when Eq. (2) has two HTs, then the Type I error in one HT essentially corresponds to the Type II error in the other. For the general case, all H_1^j ($j = 1, 2, \dots, k$) compose a disjoint partition of the whole mean space Θ . This partition indicates that any mean vector $\boldsymbol{\mu}$ satisfying H_0^j must satisfy one of H_1^l ($l \neq j$). Then, we are able to show,

$$\begin{aligned} \text{P}\{\text{Type I error in } HT_j\} &\leq \text{P}\{\text{Reject } H_1^l \mid \boldsymbol{\mu} \in H_1^l\} \\ &= \text{P}\{\text{Type II error in } HT_l\} \leq \alpha, \text{ if } \boldsymbol{\mu} \in H_1^l, \end{aligned}$$

or equivalently,

$$\begin{aligned} \text{P}\{\text{Type I error in } HT_j\} &\leq \alpha, \\ \forall \boldsymbol{\mu} \in H_0^j, \quad j &= 1, 2, \dots, k. \end{aligned} \quad (4)$$

Above all, we formulate the fixed-precision R&S problem as a multiple HT problem in Eq. (2) and illustrate that its precision (i.e., PCS guarantee in Eq. (1)) can be delivered by controlling the Type II error for each single HT_j , as presented in Eq. (3).

2.1.2 The indifference-zone assumption

We next consider each HT_j in Eq. (2) individually and notice that its Type I and II errors need to be controlled

either directly or indirectly as discussed in Section 2.1.1. However, for a given set of samples, simultaneously controlling both types of error probabilities might be impossible. To show this, we connect these two error probabilities via the power function of the test, i.e.,

$$\beta_j(\boldsymbol{\mu}) = \text{P}\{\text{Reject } H_0^j \mid \boldsymbol{\mu}\} \\ = \begin{cases} \text{P}\{\text{Type I error in } HT_j\}, & \text{if } \mu_j \leq \max_{i \neq j} \mu_i \\ 1 - \text{P}\{\text{Type II error in } HT_j\}, & \text{if } \mu_j > \max_{i \neq j} \mu_i \end{cases}$$

For most testing procedures, the power function $\beta_j(\boldsymbol{\mu})$ is continuous with respect to $\boldsymbol{\mu}$. Then,

$$\text{P}\{\text{Type I error in } HT_j\} = 1 - \text{P}\{\text{Type II error in } HT_j\}, \\ \text{when } \mu_j = \max_{i \neq j} \mu_i. \quad (5)$$

Obviously, this equation conflicts with the constraints stated in Eqs. (3) and (4). Therefore, the testing procedure satisfying Eq. (3) may not exist. It further reveals that in R&S problems, we may not be able to select the best with the desired precision when the means are sufficiently close to each other.

To overcome this obstacle, Bechhofer (1954) introduced a so-called indifference-zone (IZ) parameter $\delta > 0$, which refers to the smallest mean difference worth detecting. Given the IZ, the R&S problems are modified to select the best alternative when all the inferior alternatives are outside the IZ of the best. Accordingly, the PCS guarantee in Eq. (1) is rewritten as,

$$\text{PCS-IZ}(\boldsymbol{\mu}) = \text{P}\{\text{Select the best alternative} \mid \boldsymbol{\mu}\} \geq 1 - \alpha, \\ \forall \boldsymbol{\mu} \in \Theta^\delta, \quad (6)$$

where $\Theta^\delta = \{\boldsymbol{\mu} : \mu_{[k]} - \delta > \mu_{[k-1]}\}$ is called the IZ. Following the same logic in Section 2.1.1, this R&S problem can be reformulated as a multiple HT problem, that is,

$$(HT_j^\delta) H_0^{j,\delta} : \mu_j + \delta \leq \max_{i \neq j} \mu_i \text{ versus } H_1^{j,\delta} : \mu_j - \delta > \max_{i \neq j} \mu_i, \\ \forall j = 1, 2, \dots, k. \quad (7)$$

We remark here that, for any mean vector $\boldsymbol{\mu} \in \Theta^\delta$ of interest, either $H_0^{j,\delta}$ or $H_1^{j,\delta}$ is true, which ensures the test above is well-defined.

Given the IZ parameter δ , the corresponding power function is defined in two non-adjacent sets, i.e., $\{\boldsymbol{\mu} : \mu_j + \delta \leq \max_{i \neq j} \mu_i\}$ and $\{\boldsymbol{\mu} : \mu_j - \delta > \max_{i \neq j} \mu_i\}$. This frees us from facing the adjacent point, at which the Type I and II error probabilities cannot be controlled as desired because their sum is forced to be one. Therefore, in the presence of the IZ parameter, we can control both types of errors for each HT_j^δ or the Type II errors for

HT_j^δ ($j = 1, 2, \dots, k$). Accordingly, the R&S problems with PCS-IZ guarantee can also be tackled. In Section 3, we will explain in detail how several representative R&S procedures are derived along this line.

2.1.3 PCS and PGS

As stated in Section 2.1.2, the PCS guarantee in Eq. (1) is difficult to deliver. Therefore, the IZ parameter is introduced, and the R&S problems are restricted to a smaller mean vector space. As a consequence, the PCS-IZ guarantee in Eq. (6) is delivered whenever the best mean is at least δ larger than the others. However, in practice, several alternatives may have means that fall into the indifference zone, and these alternatives are called good alternatives. According to the definition of IZ, we should be indifferent if one of these good alternatives is selected as the best. Hence, we may care about the probability of good selection (PGS) rather than the original PCS, where the PGS guarantee is represented as,

$$\text{PGS}(\boldsymbol{\mu}) = \text{P}\{\text{Select a good alternative} \mid \boldsymbol{\mu}\} \geq 1 - \alpha, \\ \forall \boldsymbol{\mu} \in \Theta. \quad (8)$$

In the area of multi-armed bandits, a good selection is viewed as an approximately correct selection. Accordingly, the PGS guarantee is also called the probably approximately correct (PAC) selection guarantee (Even-Dar et al., 2006; Ma and Henderson, 2017).

Notice that for the R&S procedures with PCS-IZ guarantee, it is natural to expect that they could also deliver the PGS guarantee. Unfortunately, several counterexamples have been provided (Eckman and Henderson, 2018a).

In the following, we attempt to explain this phenomenon from the hypothesis-testing perspective. Similar to Section 2.1.1, to select a good alternative, it suffices to test, for any given alternative j , whether it is a good alternative, i.e., $\mu_j + \delta > \max_{i \neq j} \mu_i$. Therefore, we formulate R&S problems with PGS guarantee as a multiple HT problem, that is,

$$(HT_j^G) H_0^{j,G} : \mu_j + \delta \leq \max_{i \neq j} \mu_i \\ \text{versus } H_1^{j,G} : \mu_j + \delta > \max_{i \neq j} \mu_i, \forall j. \quad (9)$$

Suppose that a procedure with PCS-IZ guarantee of Eq. (6) exists, and we want to know whether it can deliver the PGS guarantee in Eq. (8). According to the previous analysis, an easy way is by checking the Type II error constraints presented in Eq. (3). In Table 1, we summarize the R&S problems with different probability guarantees and their corresponding HT formulations. Table 1 shows that $H_0^{j,G} = H_0^{j,\delta}$. However, $H_1^{j,G}$ refers to a larger mean vector space than $H_1^{j,\delta}$. Therefore, the Type II error probability in

Table 1 R&S problems and their HT formulations

Goal of R&S	Means	HT formulations
PCS	$\mu_{[k]} > \mu_{[k-1]}$	$H_0^j : \mu_j \leq \max_{i \neq j} \mu_i$ v.s. $H_1^j : \mu_j > \max_{i \neq j} \mu_i, \forall j$
PCS-IZ	$\mu_{[k]} - \delta > \mu_{[k-1]}$	$H_0^{j,\delta} : \mu_j + \delta \leq \max_{i \neq j} \mu_i$ v.s. $H_1^{j,\delta} : \mu_j - \delta > \max_{i \neq j} \mu_i, \forall j$
PGS	$\mu_{[k]} > \mu_{[k-1]}$	$H_0^{j,G} : \mu_j + \delta \leq \max_{i \neq j} \mu_i$ v.s. $H_1^{j,G} : \mu_j + \delta > \max_{i \neq j} \mu_i, \forall j$

HT_j^G may not satisfy Eq. (3) even though it is satisfied in HT_j^δ . In other words, the PGS guarantee cannot be guaranteed. To overcome this drawback, Eckman and Henderson (2020) constructed several sufficient conditions under which the PCS-IZ guarantee can imply the PGS guarantee.

On the opposite side, Table 1 depicts that the PGS guarantee implies the PCS-IZ guarantee. Thus, interest has recently emerged in developing the procedures with the PGS guarantee (Fan et al., 2016; Eckman and Henderson, 2018a).

2.2 Fixed-budget R&S

In this section, we consider the R&S procedures under a fixed sampling budget. By its nature, one can always select the alternative with the largest sample mean as the best when the sampling budget is exhausted. Therefore, the key issue here is how to allocate the budget efficiently. When the allocation can be made multiple times, one effective method is to re-determine the allocation adaptively at each stage based on the sampling information collected so far. Thus, a dynamic-programming (Bellman, 1966; Bertsekas, 1995) formulation looks proper to derive an optimal allocation policy.

Under the DP formulation, R&S problems turn into finding a sequence of sampling allocation decisions to optimize the precision of the final selection. Besides the PCS used in Section 2.1, another popular measure to describe the precision of selection is the expected opportunity cost (EOC). In fact, the PCS is related to the so-called 0–1 loss, i.e., only a correct selection acquires a reward, while the EOC describes the precision of selection by its opportunity cost. Particularly, when the EOC is used, a non-best selection also obtains a reward proportional to the discrepancy in the mean from the best one, which corresponds to a linear loss function. Instead of focusing on the final selection, some researchers have chosen to optimize the way information has been collected, e.g., by maximizing the expected value of information (EVI) collected at each stage.

2.2.1 Fixed-budget R&S formulated as dynamic-programming

Suppose that a total sampling budget N is allocated to the k alternatives progressively along T stages, each endowed

with a budget of $\tau = N/T$ (in the special case when $\tau = 1$, the samples are allocated one by one.). We assume that the τ samples are collected according to some sampling allocation policy at each stage t ($t = 1, 2, \dots, T$), termed by π_t . The information about the alternatives is revealed gradually along the sequential sampling. To track the process, we denote \mathcal{E}_0 as the initial information on the alternatives and \mathcal{E}_t as the information collected up to the end of stage t , for $t = 1, 2, \dots, T$. The inter-stage updating rule of the information can be defined by a transition function f_t , i.e., $\mathcal{E}_t = f_t(\mathcal{E}_{t-1}, \pi_t, \zeta_t)$, where ζ_t refers to the randomness of the samples collected at stage t . After the final stage, the selection decision is made based on all the information (i.e., \mathcal{E}_T) that is collected.

Let $V(\mathcal{E}_T)$ denote the terminal value function we want to optimize. For instance, when our objective is to minimize the probability of incorrect selection (i.e., 1–PCS), the value function can be set as the 0–1 function, which is 1 if the selected alternative is not the best and 0 otherwise. Then, the R&S procedures are formulated as a DP, which is,

$$\min_{\pi} E^{\pi}[V(\mathcal{E}_T)], \quad (10)$$

where the decision is a sequence of allocation policies, i.e., $\pi = (\pi_1, \pi_2, \dots, \pi_T)$. In the literature, the DP problem is often handled recursively through the associated Bellman equation,

$$V_t^*(\mathcal{E}_t) = \min_{\pi_{t+1}} E[V_{t+1}^*(\mathcal{E}_{t+1})],$$

$$t = T-1, T-2, \dots, 0, \quad (11)$$

where the value function $V_t^*(\mathcal{E}_t)$ defines the optimal expected cost-to-go from current stage t to the terminal and the terminal cost $V_T^*(\mathcal{E}_T) = V(\mathcal{E}_T)$.

Notice that the Bellman equation builds the relationship between the value functions in the current and next stages. As a consequence, the original DP is broken into a series of static optimization problems although in a stage-by-stage and recursive form. However, in practice, the Bellman equation is typically difficult to solve, and the difficulty is illustrated as follows. To solve the Bellman equation, the next-to-terminal cost-to-go $V_{t+1}^*(\mathcal{E}_{t+1})$ in Eq. (11) has to be calculated by backward iterations. Unfortunately, these calculations tend to be increasingly difficult, as the number of stages increases due to the ‘‘curse of dimensionality’’. In Section 4, we will explain in detail how existing studies

have resolved this problem and obtained the corresponding sample allocation rules (or R&S procedures).

2.2.2 Consistency of fixed-budget procedures

With a fixed sampling budget, the DP R&S procedures provide no probability guarantee on the correctness of the selection. Alternatively, they usually process another appealing property of consistency. A procedure is said to be consistent if its selected alternative converges to the true best as the total budget goes to infinity.

The consistency of a DP procedure is generally difficult to show directly. As long as all the alternatives receive infinite sampling budget in the limit, we will always have the exact information on the ranking of their true means to select the best correctly. Hence, asymptotically infinite samples on all the alternatives often work as a sufficient condition to verify the consistency of a procedure in the literature.

2.3 Connection to the frequentist and Bayesian formulations

Before this paper, the R&S procedures under fixed-precision and fixed-budget were often classified into the frequentist and Bayesian procedures in the literature (Kim and Nelson, 2006b). The main reason is that the precision of a selected alternative or generally the value function in DP is often described under the corresponding frequentist or Bayesian probability models. However, some exceptions exist. For instance, Frazier (2014) proposed a R&S procedure with a PCS guarantee under a Bayes-inspired framework, and Chen et al. (2000) suggested a R&S procedure with a fixed budget under a frequentist framework.

Moreover, given that the R&S problems under fixed-precision can be formulated as a hypothesis test, any testing rule, frequentist or Bayesian, can ideally be used to derive the corresponding R&S procedures. Similarly, more sample allocation (or R&S) procedures can be derived under either a frequentist or Bayesian framework for the R&S problems under a fixed sampling budget. Therefore, in our view, R&S procedures can be properly classified based on their underlying methodological formulations (i.e., HT or DP).

3 Fixed-precision procedures

Considering the fixed-precision constraint, most of the existing R&S procedures are designed under the IZ formulation and deliver the PCS-IZ guarantee in Eq. (6). These procedures are often called IZ procedures. Following the discussion in Section 2.1, we will first show in detail how the stage-wise and sequential IZ procedures are derived by addressing the corresponding HT problem in Eq. (7). Then, we move to the newly designed IZ-free

procedure, which is able to deliver both the PCS and PGS guarantees.

Before moving to the next part, we first set up some notations. Let X_{ij} denote the j th observation from alternative i , for $i = 1, 2, \dots, k$ and $j = 1, 2, \dots$. Unless specifically stated, we assume these observations are independent across alternatives and $\{X_{ij} : j = 1, 2, \dots\}$ are independent and identically distributed (i.i.d.) Gaussian distribution with mean μ_i and variance σ_i^2 . Let $\bar{X}_i(n)$ and $S_i^2(n)$ denote, respectively, the sample mean and sample variance calculated based on the first n samples from alternative i .

3.1 Stage-wise R&S procedures

We start by deriving Bechhofer's procedure (Bechhofer, 1954), which is probably known as the first R&S procedure in the literature. It considers a special case where the variances across all alternatives are common and known, i.e., $\sigma_1^2 = \sigma_2^2 = \dots = \sigma_k^2 = \sigma^2$, and the goal is to deliver the PCS-IZ guarantee. In this case, one natural procedure for its corresponding HT problem in Eq. (7) works as follows: for $j = 1, 2, \dots, k$, reject $H_0^{j,\delta}$ (i.e., select alternative j), if $\bar{X}_j(n) - \max_{i \neq j} \bar{X}_i(n) \geq z$, and accept $H_0^{j,\delta}$ otherwise. Here, the constant z and the common sample size n of all alternatives need to be carefully chosen.

Only a single alternative is expected to be returned as the best. Straightforwardly, it occurs if only one $H_0^{j,\delta}$ is rejected. This suffices to require that the rejection regions for $H_0^{j,\delta}$ ($j = 1, 2, \dots, k$) compose the disjoint partition of the whole space \mathbb{R}_+^k . One way to achieve this goal is setting $z = 0$. In doing so, the alternative with the largest sample mean is selected as the best. Moreover, the common sample size n is chosen such that the Type II error probability for each HT_j^δ satisfies Eq. (3), and specifically,

$$\begin{aligned}
 & \mathbb{P}\{\text{Type II error in } HT_j^\delta\} \\
 &= \mathbb{P}\left\{\bar{X}_j(n) - \max_{i \neq j} \bar{X}_i(n) < 0 \mid H_1^{j,\delta}\right\} \\
 &= \mathbb{P}\left\{\max_{i \neq j} \frac{\sqrt{n}[\bar{X}_i(n) - \bar{X}_j(n) - (\mu_i - \mu_j)]}{\sqrt{2\sigma^2}} > \right. \\
 &\quad \left. - \max_{i \neq j} (\mu_i - \mu_j) \sqrt{\frac{n}{2\sigma^2}} \mid H_1^{j,\delta}\right\} \\
 &\leq \mathbb{P}\left\{\max_{i \neq j} Z_i > \delta \sqrt{\frac{n}{2\sigma^2}}\right\} \leq \alpha, \tag{12}
 \end{aligned}$$

where Z_i ($i \neq j$) is a $(k-1)$ -dimensional multivariate Gaussian random variable with means 0, variances 1, and common pairwise correlations $1/2$. Let h denote the $(1-\alpha)$ quantile of the maximum of Z_i ($i \neq j$). The common sample size n is chosen as,

$$n = \left\lceil \frac{2h^2 \sigma^2}{\delta^2} \right\rceil, \quad (13)$$

where $\lceil x \rceil$ denotes the smallest integer no smaller than x .

Following the testing procedure above, a R&S procedure can be constructed. It first determines the common sample size allocated to each alternative as Eq. (13). Then, it selects the alternative with the largest sample mean as the best. This is exactly Bechhofer's procedure.

Regarding Bechhofer's procedure, we make two remarks here.

(i) From Eq. (12), we see that the worst-case of Type II error probabilities is attained when the best mean is exactly δ better than all the others, i.e., $\mu_{[k]} - \delta = \mu_{[k-1]} = \dots = \mu_{[1]}$. Thus, this configuration of means is the most difficult situation in Θ^δ , and Bechhofer (1954) named it the least favorable configuration (LFC) of means.

(ii) Bechhofer's procedure is also able to deliver the PGS guarantee in Eq. (8). To verify this statement, we only need to prove that the Type II error constraint in Eq. (3) can be achieved while applying the procedure to address the HT_j^G for all j . This proof is easily accomplished and therefore omitted in this study.

Rinott (1978) extends Bechhofer's procedures to the situation where the variances across alternatives are unknown and unequal. To handle this situation, Bechhofer's procedure is modified in three aspects. First, an initial stage is included in which a small number of samples are generated to estimate the unknown variances. Second, the total sample sizes allocated to each alternative are not the same any more but are set to be positively proportional to its sample variance. Third, the constant h_R in the total sample size N_i needs to be modified accordingly. Finding this constant needs to solve a root-find problem with integration, i.e.,

$$\int_{-\infty}^{\infty} \Psi_{n_0-1}^{k-1}(t + h_R) \psi_{n_0-1}(t) dt = (1-\alpha),$$

where Ψ_{n_0-1} and ψ_{n_0-1} denote the cumulative distribution function and probability density function of a standard student- t distribution with n_0-1 degrees of freedom, respectively. Historically, given the limited computational capacity, it is considered difficult to solve; hence, tables are provided (Wilcox, 1984; Bechhofer et al., 1995; Goldsman et al., 1998). The new two-stage procedure (named as the Rinott's procedure) is presented as follows.

Procedure 1 Rinott's procedure

Require: Number of alternatives k , common first-stage sample size $n_0 \geq 2$, PCS $1-\alpha$, IZ parameter δ , and a constant h_R

1: Generate n_0 samples for each alternative i , and calculate the sample variance $S_i^2(n_0)$

2: **for** $i \leftarrow 1 : n$ **do**

3: Let

$$N_i \leftarrow \max \left\{ n_0, \left\lceil \frac{h_R^2 S_i^2(n_0)}{\delta^2} \right\rceil \right\}, \quad (14)$$

4: Generate $N_i - n_0$ samples from alternative i , and calculate the sample mean $\bar{X}_i(N_i)$

5: **end for**

6: Select $\arg \max_{i=1,2,\dots,k} \bar{X}_i(N_i)$ as the best

As the simplest and most popular IZ procedure, there are a lot of variations of Rinott's procedure. For instance, to avoid the complexity in calculating h_R , some procedures (Clark and Yang, 1986) adopt Bonferroni's inequality and set it approximately as the $1-\alpha/(k-1)$ quantile of a t -distribution with n_0-1 degrees of freedom (Banerjee, 1961). As a price, it often leads to more conservativeness, which means that a larger sample size is needed for the procedure. Another variation of Rinott's procedure worth mentioning is the use of common random numbers (CRNs) (Clark and Yang, 1986; Nelson and Matejcek, 1995). CRNs artificially introduce a positive correlation between the observations from each pair of alternatives, thus decreasing the variance of their sample mean difference. In doing so, the R&S process becomes much easier, and the sample size required is ultimately reduced.

3.2 Sequential R&S procedures

Paulson's procedure is one of the early sequential R&S procedures, and this subsection will start from re-deriving this procedure from the hypothesis-testing perspective. Same as Bechhofer's procedure, Paulson's procedure also considers the special case with common and known variances, i.e., $\sigma_1^2 = \sigma_2^2 = \dots = \sigma_k^2 = \sigma^2$.

Similar to Section 3.1, we first consider each HT_j^δ individually and our task is to design a sequential testing procedure for it. However, such sequential procedure is not trivial because it involves multiple pairwise comparisons between alternatives. As a remedy, we break down HT_j^δ into a group of HT problems, each of which considers a pairwise comparison between alternative j and one of the other alternatives. Particularly, HT_j^δ is decomposed into:

$$(HT_{ji}^\delta) H_0^{ji,\delta} : \mu_j + \delta \leq \mu_i$$

$$\text{versus } H_1^{ji,\delta} : \mu_j - \delta > \mu_i, \forall i \neq j. \quad (15)$$

Meanwhile, to control the Type II error in HT_j^δ at most α as desired in Eq. (3), we adopt Bonferroni's inequality and require:

$$P\{\text{Type II error in } HT_{ji}^\delta\} \leq \alpha/(k-1), \quad \forall i \neq j. \quad (16)$$

A sequential procedure for HT_{ji}^δ is noticeably easy to obtain while satisfying Eq. (16), and a vast volume of literature supports it. Specifically, we may use Wald's sequential probability ratio test (SPRT) (Wald 1945; 1947), which,

$$\text{rejects } H_0^{ji,\delta}, \text{ if } n(\bar{X}_j(n) - \bar{X}_i(n)) \geq a - \lambda n,$$

$$\text{accepts } H_0^{ji,\delta}, \text{ if } n(\bar{X}_j(n) - \bar{X}_i(n)) \leq -a + \lambda n,$$

and continues to take samples otherwise. Here, $0 < \lambda < \delta$ and a is chosen as $a = \ln\left(\frac{k-1}{\alpha}\right) \frac{\sigma^2}{\delta - \lambda}$.

The original R&S problem is reformulated as $k(k-1)$ simultaneous HT problems, i.e., HT_{ji}^δ , for $j \neq i$. Each HT_{ji}^δ considers that the pairwise comparison between alternatives j and i is resolved by a sequential procedure as mentioned above. Intuitively, at any time of the sampling process, we should select alternative j as the best if all the $H_0^{ji,\delta}$ ($i \neq j$) are rejected. We eliminate alternative j from consideration if one of the $H_0^{ji,\delta}$ ($i \neq j$) is accepted. Otherwise, we continue to take samples. Once an alternative is eliminated, we should stop taking samples from this alternative and abandon all the HT_{ji}^δ regarding it. For clarity, $I(n)$ denotes the set of surviving alternatives right before stage n . Then, a sequential procedure is designed as,

$$\text{selecting alternative } j, \text{ if } n(\bar{X}_j(n) - \bar{X}_i(n)) \geq a - \lambda n,$$

$$\forall i \in I(n) \text{ and } i \neq j,$$

$$\text{eliminating alternative } j, \text{ if } n(\bar{X}_j(n) - \bar{X}_i(n)) \leq -a + \lambda n,$$

$$\exists i \in I(n) \text{ and } i \neq j.$$

It continues to take samples from the surviving alternatives otherwise. This sequential procedure is known as Paulson's procedure.

Kim and Nelson (2001) extended Paulson's procedure to the case of unknown and unequal variances. Similar to the previous two-stage procedures, Kim and Nelson's (\mathcal{KN}) procedure also uses an additional initial stage of sampling to estimate the unknown variances. After the variances are estimated, it then starts screening alternatives just as Paulson's procedure does. In addition, replacing Paulson's bound by a tighter bound of Fabian (1974) and considering the estimated variances that are random variables, the \mathcal{KN} procedure re-assigns the values of λ and a to ensure the same PCS guarantee. The detailed \mathcal{KN} procedure is presented in Procedure 2.

Procedure 2 \mathcal{KN} procedure

Require: Number of alternatives k , common first-stage sample size $n_0 \geq 2$, PCS $1 - \alpha$, IZ parameter δ , and a constant h

$$1: \text{ Set } \eta = \frac{1}{2} \left[\left(\frac{2\alpha}{k-1} \right)^{-2/(n_0-1)} - 1 \right]$$

$$2: I \leftarrow \{1, 2, \dots, k\}, h^2 = 2\eta(n_0 - 1), n \leftarrow n_0$$

$$3: \text{ Generate } n_0 \text{ samples to each alternative } j \text{ and calculate } \bar{X}_j(n_0). \text{ For } i, j \in I, S_{ji}^2 = \frac{1}{n_0 - 1} \sum_{l=1}^{n_0} [X_{jl} - X_{il} - (\bar{X}_j(n_0) - \bar{X}_i(n_0))]^2$$

4: **while** $|I| > 1$ **do**

$$5: \text{ Set } W_{ji} = \max \left\{ 0, \frac{\delta}{2n} \left(\frac{h^2 S_{ji}^2}{\delta^2} - n \right) \right\} \text{ and}$$

$$I \leftarrow \{j : j \in I \text{ and } \bar{X}_j(n) - \bar{X}_i(n) \geq -W_{ji}(n), \forall i \in I, i \neq j\}$$

6: Take an additional observation from each alternative $j \in I$, and set $n \leftarrow n + 1$

7: **end while**

8: Select the alternative in I as the best

An intuitive way to understand the \mathcal{KN} procedure is presented in Fig. 1. For each pair of alternatives j and i , it constructs the partial-sum process of their mean difference $\{n(\bar{X}_j(n) - \bar{X}_i(n)) : n = 1, 2, \dots\}$. At each stage n , \mathcal{KN} checks whether this partial-sum process exits from the triangular region and makes decisions accordingly.

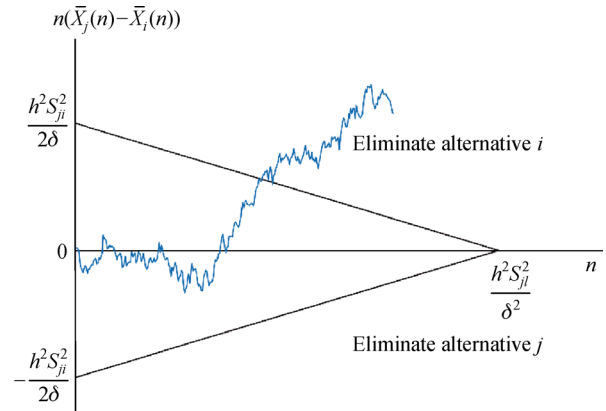


Fig. 1 Triangular region for the \mathcal{KN} procedures.

The \mathcal{KN} procedure has numerous variations, and this family of procedures is shown to be effective among IZ procedures (Kim and Nelson, 2006b; Branke et al., 2007). All these variations are classified into two categories. The first category intends to enhance the efficiency of the \mathcal{KN} procedure. For instance, Hong (2006) designed a variance-dependent sampling rule. Moreover, Tsai and Nelson (2009) and Tsai et al. (2017) adopted the control-variate technique. In another study, Nelson et al. (2001) took advantage of the first-stage samples to screen out alternatives that are unlikely to be the best. The second category intends to address different practical situations.

For instance, Hong and Nelson (2005) considered the cost of switching between alternatives to take samples and designed a new procedure to balance the tradeoff between sampling and switching costs. In a follow-up study, Hong and Nelson (2007b) noticed a situation where alternatives may be revealed sequentially, thus designing a new procedure for this situation. Meanwhile, Kim and Nelson (2006a) studied the steady-state experiments and designed a new procedure achieving the PCS guarantee asymptotically.

3.3 Indifference-zone-free R&S procedures

In Sections 3.1 and 3.2, we have seen how the IZ formulation (i.e., $\boldsymbol{\mu} \in \Theta^\delta = \{\boldsymbol{\mu} : \mu_{[k]} - \delta > \mu_{[k-1]}\}$) helps to achieve the PCS guarantee. However, the problem remains on whether a R&S procedure with the PCS guarantee can be developed for all possible mean vectors in Θ .

To solve this problem, Fan et al. (2016) proposed an IZ-free procedure. We call it the FHN procedure and present it as Procedure 3. Similar to the \mathcal{KN} procedure, it decomposes a R&S problem into a group of pairwise comparisons and designs a procedure for each pairwise comparison. When $\boldsymbol{\mu} \in \Theta$, the pairwise mean differences might be arbitrarily close to zero. Then, the desired procedure is intended to detect whether these mean differences are zero or not. Motivated by the law of iterated logarithm, this IZ-free procedure adopts a new continuation region whose boundary function grows to infinity at a rate between $\mathcal{O}(\sqrt{n \log \log n})$ and $\mathcal{O}(n)$. For instance, a boundary function $\sqrt{[c + \log(n+1)](n+1)}$ is used, as shown in Procedure 3.

Procedure 3 FHN procedure

Require: Number of alternatives k , common first-stage sample size $n_0 \geq 2$, and PCS $1 - \alpha$

1: Set $c = -2 \log\left(\frac{2\alpha}{k-1}\right)$

2: $I \leftarrow \{1, 2, \dots, k\}$, $n \leftarrow n_0$

3: Generate n_0 samples to each alternative j , and calculate $\bar{X}_j(n_0)$. For $i, j \in I$,

$$S_{ji}^2(n_0) = \frac{1}{n_0 - 1} \sum_{l=1}^{n_0} [X_{jl} - X_{il} - (\bar{X}_j(n_0) - \bar{X}_i(n_0))]^2$$

4: **while** $|I| > 1$ **do**

5: Set $t_{ji}(n) = n/S_{ji}^2(n)$ and

$$g_{ji}(t_{ji}(n)) = \sqrt{[c + \log(t_{ji}(n) + 1)](t_{ji}(n) + 1)}, \text{ and}$$

let $I \leftarrow \{j : j \in I \text{ and } t_{ji}(n)[\bar{X}_j(n) - \bar{X}_i(n)] \geq -g_{ji}(t_{ji}(n)), \forall i \in I, i \neq j\}$

6: Take an additional observation from each alternative $j \in I$, and set $n \leftarrow n + 1$

7: **end while**

8: Select the alternative in I as the best

Now, we illustrate from the HT perspective why this IZ-free procedure is able to achieve the PCS guarantee in

Eq. (1). As mentioned in Section 2.1.2, the challenge for the conventional IZ procedures is how to control the Type I and II errors in each HT_j simultaneously when the second-best mean is arbitrarily close to the best. Specifically, Eq. (5) shows that we might lose such control at the point $\boldsymbol{\mu}^0$ with $\mu_j^0 = \max_{i \neq j} \mu_i^0$, which is caused by the continuity of the power function. The FHN procedure resolves this challenge by forcing its power function $\beta_j(\cdot)$ to be discontinuous at $\boldsymbol{\mu}^0$.

The FHN procedure addresses HT_j ($j = 1, 2, \dots, k$) by,

$$\text{rejecting } H_0^j, \text{ if } t_{ji}(n)[\bar{X}_j(n) - \bar{X}_i(n)] \geq g(t_{ji}(n)),$$

$$\forall i \in I(n) \text{ and } i \neq j,$$

$$\text{accepting } H_0^j, \text{ if } t_{ji}(n)[\bar{X}_j(n) - \bar{X}_i(n)] \leq -g(t_{ji}(n)),$$

$$\exists i \in I(n) \text{ and } i \neq j,$$

and continues sampling otherwise. Here $I(n)$ denotes the set of surviving alternatives right before stage n . Then, a careful derivation yields that,

$$\beta_j(\boldsymbol{\mu}) \geq 1 - \alpha, \text{ for } \boldsymbol{\mu} \text{ with } \mu_j > \max_{i \neq j} \mu_i,$$

$$\text{and } \beta_j(\boldsymbol{\mu}) \leq \alpha, \text{ for } \boldsymbol{\mu} \text{ with } \mu_j \leq \max_{i \neq j} \mu_i,$$

thereby demonstrating a discontinuous power function $\beta_j(\boldsymbol{\mu})$. The inequalities above also show that the FHN procedure satisfies the constraints of error probability in Eqs. (3) and (4), thus implying that the desired PCS guarantee in Eq. (1) can be achieved.

Fan et al. (2016) also extended the FHN procedure to incorporate an IZ parameter when it is available. Particularly, a stopping condition based on the IZ parameter is embedded into the original FHN procedure. The new procedure is shown to be able to achieve not only the PCS guarantee in Eq. (1), but also the PGS guarantee in Eq. (8).

4 Fixed-budget procedures

In this section, we review the existing fixed-budget R&S procedures related to the DP formulation. With a fixed sampling budget, the main task of R&S procedures is to determine a sample allocation policy, which is formulated as a DP problem in Eq. (10) as introduced in Section 2.2. This DP problem is essentially a finite-horizon stochastic DP and can be solved exactly in principle by backward induction through Bellman equation of Eq. (11). However, this exact procedure is often impossible to execute due to the curse of dimensionality. This limitation motivates the researchers to consider the suboptimal solutions generated

by easily implementable approximation procedures. In particular, all the procedures reviewed in this section can be regarded as approximate dynamic programming (ADP) procedures.

4.1 Static-allocation based procedures

As a practically acceptable DP procedure is impossible to obtain, one possible approach would be developing a good heuristic procedure instead. Intuitively, a superior DP procedure “optimizes” the way of collecting information about the mean of each alternative. Hypothetically, if we have perfect information at the beginning but still have to make a selection based on the samples, a simple static allocation policy that maximizes the precision of the selection will be proper. For example, assuming that the precision of selection is measured by the PCS guarantee in Eq. (1), the optimal allocation policy can be determined by solving the following static optimization problem:

$$\max_{n_{[1]}+\dots+n_{[k]}=N} \mathbb{P}\left\{\bar{X}_{[k]}(n_{[k]}) > \max_{[j] \neq [k]} \bar{X}_{[j]}(n_{[j]})\right\}, \quad (17)$$

where $n_{[i]}$ denotes the sample size allocated to alternative $[i]$, for $i = 1, 2, \dots, k$.

Based on the static allocation policy, several procedures have been developed. The optimal computing budget allocation (OCBA) procedure initiated by Chen (1996) and Chen et al. (2000) is among the most famous static-allocation-based procedures. Moreover, the OCBA procedure has also been extended to sequential settings, and the basic idea is to approximate the static allocation policy dynamically based on the sample information.

Taking the sequential algorithm of OCBA proposed by Chen et al. (2000) as an example, a total budget of N is allocated to T stages sequentially with each stage endowed with $\tau = N/T$. Perfect information is assumed in developing the OCBA procedure at first. Particularly, it assumes the information given at stage t as $\mathcal{E}_t = \{(\mu_j, \sigma_j^2), j = 1, 2, \dots, k\}$ for $0 \leq t \leq T$. For any intermediate stage t , the allocation policy is determined by a static allocation problem as of Eq. (17), in which the budget for the first t stages are reallocated for a myopic objective of maximizing PCS as if the selection is made at the end of the current stage.

$$V_t^{\text{OCBA}}(\mathcal{E}_t) = \max_{n_{[1],t}+\dots+n_{[k],t}=\tau t} \mathbb{P}\left\{\bar{X}_{[k]}(n_{[k],t}) > \max_{[j] \neq [k]} \bar{X}_{[j]}(n_{[j],t})\right\}.$$

Here, $n_{[i],t}$ is the total sample size that is allocated to alternative $[i]$ up to the end of stage t , for $i = 1, 2, \dots, k$ and $t = 1, 2, \dots, T$. The allocation rule is then derived by approximating the PCS with Bonferroni’s inequality and

Procedure 4 OCBA procedure

Require: Number of alternatives k , common first-stage sample size $n_0 \geq 5$, total sampling budget N , and sampling budget τ per stage

1: Generate n_0 samples from each alternative i

2: Set $t \leftarrow 0$, $n_{i,t} \leftarrow n_0$, $b_t \leftarrow \sum_{i=1}^k n_{i,t}$

3: **while** $b_t < N$ **do**

4: Update the sample mean \bar{x}_i ¹⁾ and the sample variance $\hat{\sigma}_i^2$; $(k) \leftarrow \arg \max_i \bar{x}_i$ and $d_{(i)(k)} \leftarrow \bar{x}_{(k)} - \bar{x}_i$

5: Set $b_{t+1} \leftarrow b_t + \tau$. Calculate the new budget allocation $n_{1,t+1}, n_{2,t+1}, \dots,$

$n_{k,t+1}$ satisfying $\sum_i n_{i,t+1} = b_{t+1}$ according to $\frac{n_{i,t+1}}{n_{j,t+1}} = \left(\frac{\hat{\sigma}_i/d_{(i)(k)}}{\hat{\sigma}_j/d_{(j)(k)}}\right)^2$, for

$i \neq j \neq (k)$ and $n_{(k),t+1} = \hat{\sigma}_{(k)} \sqrt{\sum_{i \neq (k)} n_{i,t+1}^2 / \hat{\sigma}_i^2}$

6: Generate $\max\{0, n_{i,t+1} - n_{i,t}\}$ samples from each alternative i . Set $t \leftarrow t + 1$

7: **end while**

8: Select $\arg \max_i \bar{x}_i$ as the best

letting the budget per stage go to infinity. The resulting allocation rule is presented in Step 5 in Procedure 4.

Moreover, using the large deviation theory, Glynn and Juneja (2004) derived the asymptotic optimal allocation policy for Eq. (17) that maximizes the exponential decay rate of the probability of incorrect selection as $N \rightarrow \infty$. Specially, they showed that the optimal allocation satisfies:

$$\frac{n_{[i]}^*}{n_{[j]}^*} \approx \frac{\sigma_{[i]}^2 / (\mu_{[k]} - \mu_{[i]})^2}{\sigma_{[j]}^2 / (\mu_{[k]} - \mu_{[j]})^2}, \text{ for } [i] \neq [j] \neq [k],$$

$$\text{and } n_{[k]}^* = \sigma_{[k]} \sqrt{\sum_{[j] \neq [k]} \left(\frac{n_{[j]}^*}{\sigma_{[j]}}\right)^2}. \quad (18)$$

This equation provides a theoretical benchmark on the optimality of static allocation policy. Careful investigation reveals that the optimal allocation coincides with the one in the OCBA procedure. Thus, the OCBA policy is asymptotically efficient.

In practice, we do not have perfect information about the means and variances of the alternatives, and the OCBA procedure suggests to use sample estimates instead based on the available data at the beginning of each stage (see Step 4).

Some variations of the above OCBA procedure have been proposed. He et al. (2007) adopted the linear loss function to measure the quality of the selection and designed an OCBA-type procedure; Gao et al. (2017a) also considered the case of linear loss function but designed an OCBA-type procedure based on the large-deviation theory; Branke et al. (2007) addressed the issue of

1) Most of DP procedures are described as Bayesian procedures. Therefore, they are used to represent random variables with upper-case letters and their observations with lower-case letters. To keep in line with the existing literature, we use \bar{x} in Section 4 to denote the observation of the sample mean.

unknown variances and proposed to use student- t approximation; Peng et al. (2018) directly approximated the objective function in Eq. (17) using some feature functions; moreover, Fu et al. (2007) considered the case of correlated samples across alternatives and showed that the optimal policy agrees with that of the independent case as the correlation vanishes.

4.2 Two-stage approximation based procedures

Static-allocation-based procedures like OCBA are developed by assuming that the means and variances are known. In addition, these procedures use the corresponding sample estimators to replace the unknown parameters in practical implementations. In contrast, another stream of research takes account of the unknown means and variances in developing the procedures. These procedures often contain two stages, which include a first-stage sampling to collect some information about the unknown parameters and then use the information to guide the second-stage allocation decision.

As a representative, we shall review one famous two-stage procedure proposed by Chick and Inoue (2001), which is known as the expected value of information (EVI) procedure. In particular, we consider the one with linear loss and a budget constraint, namely, Procedure $\mathcal{LL}(\mathcal{B})$. The procedure adopts the Bayesian approach for updating the information collected about the mean performance of any alternative i , which is assumed to be a random variable W_i . At the first stage, it takes n_0 samples from alternative i and computes the sample means and variances $(\bar{x}_{i,0}, \hat{\sigma}_{i,0}^2)$. By Bayes' rule, it indicates the prior distribution of $W_i \sim \text{St}(\bar{x}_{i,0}, n_0/\hat{\sigma}_{i,0}^2, n_0-1)$, where $\text{St}(\mu, \kappa, \nu)$ denotes the student- t distribution with mean μ , precision κ , and degrees of freedom ν . If additional $n_i - n_0$ samples are allocated to alternative i and the overall sample mean and variance are $(\bar{x}_i, \hat{\sigma}_i^2)$, then the posterior distribution W_i becomes $\text{St}(\bar{x}_i, n_i/\hat{\sigma}_i^2, n_i-1)$. The final selection will go to the alternative with the largest sample mean, i.e., $(k) = \arg \max_i \bar{x}_i$. A false selection will incur a linear loss, which is $\max_i W_i - W_{(k)}$. Therefore, the problem for the second stage is to choose (n_1, n_2, \dots, n_k) to minimize the expected linear loss¹⁾, i.e.,

$$\min_{n_1+n_2+\dots+n_k=N} \mathbb{E} \left[\mathbb{E} \left[\max_i W_i - W_{(k)} \mid (X_1, X_2, \dots, X_k) \right] \right]. \quad (19)$$

Notice that (n_1, n_2, \dots, n_k) are determined before the second stage. Therefore, to calculate the expected linear loss, we need to take the expectation with respect to the second-stage samples (X_1, X_2, \dots, X_k) , which are random. As the problem in Eq. (19) has no closed-form solution,

Chick and Inoue (2001) derived their allocation policy by asymptotically minimizing a bound of the expected loss.

Chick and Inoue (2001) then adapted this two-stage procedure to the dynamic setting. At each intermediate stage t , a set of observations is collected from each alternative in the previous stages, based on which the current-stage allocation policy needs to be determined. This issue is essentially what the two-stage procedure above attempts to address. Therefore, this allocation policy should be determined by applying the two-stage procedure. Particularly, a myopic perspective is taken as if the selection is made at the end of stage t , and the current-stage allocation policy is then obtained by solving Eq. (19),

$$V_t^{\text{EVI}}(\mathcal{E}_t) = \min_{\sum_i n_{i,t} = \sum_i n_{i,t-1} + \tau} \mathbb{E} \left[\mathbb{E} \left[\max_i W_i - W_{(k)} \mid (X_{1,t}, X_{2,t}, \dots, X_{k,t}) \right] \right], \quad (20)$$

where $X_{i,t}$ denotes the random samples that will be taken at stage t . The extension from the above two-stage procedure to a sequential procedure encounters an obstacle, which is caused by the unbalanced samples from different alternatives. Technically, it involves the subtraction of two student- t random variables with different degrees of freedom. Chick and Inoue (2001) overcame this difficulty by using the Welch approximation.

The process is documented in Procedure 5, where we assume the sampling cost from each alternative is the same and set as one. The optimal sampling allocation policy in Step 6 looks very similar to that of the OCBA procedure (Step 5 in Procedure 4) because these two procedures are derived in a similar way as mentioned before.

In the same work, Chick and Inoue (2001) also considered the problem with unconstrained budget. Thus, they proposed an EVI procedure to determine the number of replications to balance the replication costs against the reduction in the expected opportunity cost. They also presented analogous procedures with the 0–1 loss function. Chick et al. (2010) developed a variation of the EVI procedure in which the sampling budget is allocated to only one alternative at each stage. For this special case, they showed that most of the approximations in solving this optimal allocation policy can be avoided and therefore derived a procedure with better performance, especially in the small-budget problems.

4.3 One-step look-ahead procedures

In this section, we review the group of DP procedures that are derived using the one-step look-ahead approximation. Specifically, we consider the knowledge-gradient (KG) procedure proposed by Frazier et al. (2008).

The KG procedure also adopts a Bayesian approach to

1) In the original paper of Chick and Inoue (2001), they also considered the sampling cost, which results in an additional linear term in the objective function.

Procedure 5 EVI procedure for linear loss

Require: Number of alternatives k , common first-stage sample size $n_0 \geq 2$, total sampling budget N , and sampling budget τ per stage

- 1: Generate n_0 samples from each alternative i
- 2: Set $t \leftarrow 0$, $n_{i,t} \leftarrow n_0$, $b_t \leftarrow \sum_{i=1}^k n_{i,t}$
- 3: **while** $b_t < N$ **do**
- 4: Update \bar{x}_t and $\hat{\sigma}_t^2$. Set $\bar{x}_{(1)} \leq \bar{x}_{(2)} \leq \dots \leq \bar{x}_{(k)}$ and $\mathcal{L} = \{1, 2, \dots, k\}$
- 5: Let $\lambda_{(i)(j)}^{-1} \leftarrow \hat{\sigma}_{(i)}^2/n_{(i),j} + \hat{\sigma}_{(j)}^2/n_{(j),i}$, $d_{(i)(k)} \leftarrow \bar{x}_{(k)} - \bar{x}_{(i)}$. Set $b_{t+1} \leftarrow b_t + \tau$
- 6: For each alternative $(i) \in \mathcal{L}$, calculate

$$n_{(i),t+1} = \frac{\left(\tau + \sum_{(j) \in \mathcal{L}} n_{(j),t}\right) \left(\hat{\sigma}_{(i)}^2 \eta_{(i)}\right)^{1/2}}{\sum_{(j) \in \mathcal{L}} \left(\hat{\sigma}_{(j)}^2 \eta_{(j)}\right)^{1/2}}, \text{ where } \eta_{(i)} =$$

$$\lambda_{(i)(k)}^{1/2} \frac{n_{i,t} - 1 + \lambda_{(i)(k)} d_{(i)(k)}^2}{n_{i,t} - 2} \psi_{n_{i,t}-1} \left[\lambda_{(i)(k)}^{1/2} d_{(i)(k)} \right] \text{ for } (i) \neq (k) \text{ and}$$

$$\eta_{(k)} = \sum_{(j) \neq (k)} \eta_{(j)}, \text{ and } \psi_s(\cdot) \text{ denotes the probability density function of a}$$

standard student- t distribution with s degrees of freedom
- 7: **while** $\min_{(i) \in \mathcal{L}} (n_{(i),t+1} - n_{(i),t}) < 0$ **do**
- 8: **if** $n_{(i),t+1} - n_{(i),t} < 0$ **then**
- 9: Set $\mathcal{L} \leftarrow \mathcal{L} \setminus (i)$ and $n_{(i),t+1} \leftarrow n_{(i),t}$
- 10: **end if**
- 11: For each alternative $(i) \in \mathcal{L}$, update

$$\lambda_{(i)(k)}^{-1} = \begin{cases} \hat{\sigma}_{(i)}^2/n_{(i),t} + \hat{\sigma}_{(k)}^2/n_{(k),t}, & \text{if } (k) \in \mathcal{L} \\ \hat{\sigma}_{(i)}^2/n_{(i),t}, & \text{if } (k) \notin \mathcal{L} \end{cases}$$
- 12: Go back to Step 6
- 13: **end while**
- 14: Generate $n_{i,t+1} - n_{i,t}$ samples from each alternative $i \in \mathcal{L}$. Set $t \leftarrow t + 1$
- 15: **end while**
- 16: Select $\arg \max \bar{x}_t$ as the best

solve the R&S problem. Unlike the EVI, it determines the optimal sampling allocation policy by maximizing the expected terminal reward. Suppose that there is a budget of N samples for selecting the best from k alternatives, and information collected from the samples is summarized in the posterior distribution of the unknown mean for each alternative. In such case, μ_i^t and $(\sigma_i^t)^2$ are the mean and variance of the posterior distribution for alternative i , respectively, after observing the first t samples. Then, the problem is to determine the allocation of the $(t+1)$ th sample $z^{t+1} \in \{1, 2, \dots, k\}$ for $t = 0, 1, \dots, N-1$, in order to maximize the expected terminal reward $E[\max_i \mu_i^N | \mathcal{E}_N]$, and the alternative with the largest μ_i^N is selected as the best. Here, the information set \mathcal{E}_t records the posterior mean and variance after the t th sample and is updated according to the Bayes rule.

From the view of dynamic-programming formulation, we solve:

$$V_t^{\text{KG}}(\mathcal{E}_t) = \max_{(z^{t+1}, \dots, z^N)} E[V_N(\mathcal{E}_N) | \mathcal{E}_t]$$

$$= \max_{(z^{t+1}, \dots, z^N)} E \left[\max_i \mu_i^N | \mathcal{E}_t \right].$$

The key idea of KG procedure is to approximate $V_t(\mathcal{E}_t)$ by:

$$V_t^{\text{KG}}(\mathcal{E}_t) \approx \sum_{j=t}^{N-1} \max_{z^{j+1}} E \left[\max_i \mu_i^{j+1} - \max_i \mu_i^j | \mathcal{E}_j \right] + \max_i \mu_i^t,$$

and the problem reduces to solve the one-step optimization problem,

$$\max_{z^{t+1}} E \left[\max_i \mu_i^{t+1} - \max_i \mu_i^t | \mathcal{E}_t \right]. \quad (21)$$

Intuitively, it maximizes the increment (e.g., gradient) in the “knowledge” gained from the next sample, which explains the name “knowledge gradient”.

We assume that the samples across different alternatives are independent and have a common and known variance. In this special structure, the optimal solution of Eq. (21) has a closed form (see Steps 4–5 in Procedure 5 or Theorem 1 in Frazier et al. (2008)). This structure is highly attractive from the implementational point of view. Moreover, the procedure possesses other favorable properties. For instance, it is consistent, i.e., the selected alternative converges to the true best as the total sampling budget N grows to the infinity, and the suboptimality of the KG policy is bounded for any finite budget N .

Procedure 6 KG procedure

Require: Number of alternatives k , total sampling budget N , common and known variance σ^2 , prior predictive mean μ_i , and variance σ_i^2 for each alternative

- 1: Set $t \leftarrow 0$. Let $\mu_i^t \leftarrow \mu_i$, $\beta_i^t \leftarrow 1/\sigma_i^2$ and $\beta = 1/\sigma^2$
- 2: **while** $t < N$ **do**
- 3: Calculate the variance of the change in predictive mean by taking a sample from alternative i , $\tilde{\sigma}_i^2 = (\beta_i^t)^{-1} - (\beta_i^t + \beta)^{-1}$
- 4: Calculate

$$\zeta_i = - \left| \frac{\mu_i^t - \max_{j \neq i} \mu_j^t}{\tilde{\sigma}_i} \right|, \quad (22)$$
- 5: Choose $z^{t+1} = \arg \max_{i=1, 2, \dots, k} \tilde{\sigma}_i \left(\zeta_i \Phi(\zeta_i) + \phi(\zeta_i) \right)$, where $\Phi(\cdot)$ and $\phi(\cdot)$ denote the cumulative distribution function and probability density function of the standard Gaussian distribution, respectively
- 6: Take a sample $y_{z^{t+1}}^{t+1}$ from alternative z^{t+1} . Update $\beta_{z^{t+1}}^{t+1} \leftarrow \beta_{z^{t+1}}^t + \beta$, $\mu_{z^{t+1}}^{t+1} \leftarrow (\beta_{z^{t+1}}^t \mu_{z^{t+1}}^t + \beta y_{z^{t+1}}^{t+1}) / \beta_{z^{t+1}}^{t+1}$
- 7: Set $t \leftarrow t + 1$
- 8: **end while**
- 9: Select $\arg \max \mu_i^t$ as the best

The original KG procedure of Frazier et al. (2008) has several variations. For instance, Frazier et al. (2009) and Xie et al. (2016) extended the procedure to the case of correlated sampling and correlated Gaussian beliefs on the mean vectors. Ryzhov (2016) adopted a different way to define the value of information functions and then derived the corresponding optimal sampling allocation rule. In this rule, the allocation ratios among the non-best alternatives are quite similar to that of the OCBA procedure. However, the total proportion of samples allocated to these non-best alternatives vanishes as the total sampling budget grows to infinity. To understand the connection between Ryzhov's procedure and the OCBA procedure, Peng and Fu (2017) showed that the allocation rules of the OCBA procedure can be achieved by slightly modifying the function used to describe the value of information in Ryzhov (2016).

5 Large-scale R&S procedures using parallel computing

As mentioned before, many existing R&S procedures under either the fixed-precision or the fixed-budget formulations are designed to solve small- or medium-scale problems, with a total number of alternatives typically less than 500, which is largely due to the limited computing resource. On one hand, numerous large-scale R&S problems in practice have thousands to millions of alternatives, which are traditionally solved by optimization-via-simulation (OvS) algorithms (see, for instance, Hong and Nelson (2009) and Hong et al. (2015b) for comprehensive reviews of OvS). On the other hand, the fast development of computer technology and parallel computing (e.g., either from the multi-core personal computers to many-core servers or from smart phones to cloud services) are prevalent and ready for ordinary users to access. Then, using parallel computing to solve a large-scale R&S problem directly becomes an interesting research topic. It has even been labeled as one of the three central developments in the past 15 years by Fu and Henderson (2017).

Researchers begin investigating parallel computing for R&S problems by asking the following questions: (i) Can existing R&S procedures be easily implemented in a parallel fashion? (ii) If not, how are these procedures modified to suit for parallel computing environments? (iii) In the process of parallelization, what kind of substantial issues need to be addressed? To the best of our knowledge, Yücesan et al. (2001) and Chen (2005) are the two earliest works in the literature that attempted to answer the first question. In particular, the former implemented the OCBA procedure in a web-based parallel environment, and the latter executed a multi-stage procedure by distributing the simulation tasks to multiple processors. However, both studies tested their procedures only for a small-scale problem with 10 alternatives. Hence,

whether their procedures are suitable for handling large-scale problems is unclear. Luo et al. (2015), Luo and Hong (2011) and Ni et al. (2013; 2014; 2015; 2017) are works that intended to answer the three questions. They demonstrated that redesigned procedures can be used to solve large-scale problems with thousands to millions of alternatives in different parallel computing environments.

Various parallel computing environments are suitable for R&S problems, and they can be classified into three categories in general, i.e., Message-Passing Interface (MPI), Hadoop MapReduce, and Apache Spark (Ni et al., 2017). The MPI (Gropp et al., 1999) is a standardized and portable message-passing protocol for parallel programming on several parallel computing architectures, which is equipped with C/C++ and Fortran libraries. Hadoop (Dean and Ghemawat, 2008) is an open-source framework designed for distributed storage and processing of large amounts of data and computation using the MapReduce programming architecture. Apache Spark (Zaharia et al., 2010) is also an open-source framework for general-purpose parallel computing. MapReduce and Spark are supported by various commercial clouds including Amazon EC2, Google Cloud Platform, and Microsoft Azure (Zhong and Hong, 2020). All three frameworks can be implemented using the Master/Worker parallel structure. MPI allows more flexibility of parallel implementation but does not detect or manage core failures automatically compared with MapReduce and Spark.

In the following, we first briefly describe both the theoretical and implementational challenges as modifying existing R&S procedures to suit for parallel computing environments in Section 5.1. Then, we introduce some different performance measures and new frameworks that are developed for large-scale R&S problems in Section 5.2. Two representative procedures are also presented in Sections 5.1 and 5.2, respectively.

5.1 Extending existing procedures to parallel

In traditional R&S problems, the efficiency of a procedure can often be measured by the total running time, which is approximately the total simulation time of generating observations from different alternatives. This method is reasonable because the operations of all other calculations and comparisons are quite fast. In addition, the total time of these operations could be negligible compared with the total simulation time as solving small-scale problems in a single-processor environment. However, when handling large-scale problems in a parallel computing environment, the situation becomes complicated, as the comparison operations may become the bottleneck. The communications and synchronizations among different processors may also need to be taken into consideration. In other words, to measure the efficiency of a procedure in parallel computing environments, we shall evaluate the running time from four aspects, i.e., the simulation time, the

comparison time, the communication time (i.e., the time to transfer information between different processors), and the synchronization time (i.e., the time to wait for the ready state of all processors). For the sake of presentation, we take the stage-wise and fully sequential procedures in the fixed-precision formulation to illustrate the tradeoff among the four aspects.

Stage-wise procedures are easy to parallelize, and no communications occur among processors until the comparison operation at the end of each stage, which means they are efficient in comparison and communication. The synchronization is also not an issue if the simulation tasks are distributed evenly among different processors. Compared with fully sequential procedures, however, stage-wise procedures are typically not efficient in the total sample size, i.e., inefficient in simulation. For fully sequential procedures, they conduct all-pairwise comparisons (i.e., $k(k-1)/2$ in the worst-case) among all alternatives still in contention at each round when all alternatives add one observation, thus implying frequent communications and synchronizations among different processors. Therefore, they are inefficient in comparison, communication, and synchronization.

The total sample size is inherently determined by the theoretical framework of a procedure, which could be hardly reduced even using parallel computing. Therefore, the efficiency of stage-wise procedures has little room for improvement. Moreover, several works in the literature have focused on improving the efficiency of fully sequential procedures by redesigning them to be fit for parallel computing to reduce the times for comparison, communication, and synchronization. For instance, Luo et al. (2015) addressed the synchronization issue by proposing an asynchronization scheme to achieve a high simulation efficiency of sampling. They pointed out the potential issues caused by all-pairwise comparisons and frequent communications. Later, Ni et al. (2017) and Zhong et al. (2020) addressed the comparison issue using two different approaches, namely, a “divide-and-conquer” scheme by distributing the all-pairwise comparisons and a new comparison scheme by defining the “best” alternative differently. They further mitigated the communication burden by using batching techniques and boosting the sample size of all surviving alternatives to a maximum number afterward. Notably, different batching techniques and boosting methods are proposed in Ni et al. (2017) and Zhong et al. (2020), thus resulting in different theoretical foundations of their procedures. Before introducing more details, we first briefly describe the aforementioned Master/Worker parallel structure, which has been used in Luo et al. (2015), Ni et al. (2017), and Zhong et al. (2020).

Suppose that the parallel computing environment has $m + 1$ processors, in which one processor serves as the

master and the other m processors serve as the workers, which are denoted by workers 1, 2, ..., m . The master is the controller who determines the start and stop of the program, creates m job tasks for the workers, manages the data information, and performs all other necessary calculations. Workers 1, 2, ..., m function in a simple way: Taking the task from the master, processing the task, submitting the result to the master, and requesting the next task. The communications occur only between the master and workers, and no communication occurs among workers.

To address the synchronization issue, Luo et al. (2015) defined each job task as generating one observation from one alternative that is still in contention. In addition, all alternatives in contention are queued in a round-robin order in front of the master. This one-by-one task assignment scheme requires no synchronization among workers and can indeed fairly balance the workloads of different workers. However, given the random processing time of each task on different workers, the sequence of the simulation results sent back to the master is also random, which is likely to be different from the round-robin assigning order. Therefore, it may cause unexpected statistical issues as implementing existing fully sequential procedures based on the output sequence. One straightforward way is to restore all the outputs exactly in the same order as in the original input sequence and perform comparisons according to the input sequence, which is the basic idea for the vector-filling \mathcal{KN} (VKN) procedure of Luo et al. (2015). We omit the details about the VKN procedure but summarize two disadvantages of VKN. First, it needs to create a vector to store these outputs, which may require a large amount of memory. Second, it does not allow the failure of any worker or communication interruption, as the vector may be incomplete if some of the simulation results are lost in these situations.

To resolve these problems, Luo et al. (2015) proposed the asymptotic parallel selection (APS) procedure, which performs all-pairwise comparisons directly based on the output sequence and introduces a phantom alternative to determine the time points for comparisons.¹⁾ However, the desired finite-time PCS guarantee is no longer achieved, as the sample sizes of different alternatives in the output sequence are random and perhaps unequal; these observations from the same alternatives are not even i.i.d. Fortunately, the innovative idea of introducing the phantom alternative, which serves as a drumbeat process with predetermined time points, allows to establish a finite lag of the difference between the input and output sequences that finally vanishes in an asymptotic regime. By doing so, the APS procedure of Luo et al. (2015) provides an asymptotic PCS guarantee. We present the APS procedure in Procedure 7.

1) Note that the phantom alternative does not need to be processed by any worker but immediately returns to the master for requiring the master to perform eliminations by conducting all-pairwise comparisons.

Procedure 7 APS procedure

Require: Number of alternatives k , PCS $1-\alpha$, IZ parameter $\delta > 0$, the first-stage sample size $n_0 \geq 2$, and the number of processors $m+1$ (i.e., one master and m workers)

- 1: Let $a = -\log[2\alpha/(k-1)]$ and $I \leftarrow \{1, 2, \dots, k\}$
- 2: Let p denote the phantom alternative queued after each round-robin cycle, and let the stage r denote the current sample size of p
- 3: For all $i \in I$, record the triple $(N_{ir}, \sum_{\ell=1}^{N_{ir}} Y_{i\ell}, \sum_{\ell=1}^{N_{ir}} Y_{i\ell}^2)$, where $Y_{i\ell}$ is the ℓ th completed observation from alternative i and N_{ir} is the total sample size obtained from alternative i at stage r . Set $r \leftarrow 1$, and set $N_{ir} \leftarrow 0$, $\sum_{\ell=1}^{N_{ir}} Y_{i\ell} \leftarrow 0$, $\sum_{\ell=1}^{N_{ir}} Y_{i\ell}^2 \leftarrow 0$ for all $i \in I$
- 4: Use the round-robin order to start simulations on workers 1, 2, ..., m
- 5: **while** $|I| > 1$ **do**
- 6: Wait for the next observation Y_h .
- 7: **if** $h \in I$ **then**
- 8: Update $\sum_{\ell=1}^{N_{hr}} Y_{h\ell} \leftarrow \sum_{\ell=1}^{N_{hr}} Y_{h\ell} + Y_h$, $\sum_{\ell=1}^{N_{hr}} Y_{h\ell}^2 \leftarrow \sum_{\ell=1}^{N_{hr}} Y_{h\ell}^2 + Y_h^2$ and $N_{hr} \leftarrow N_{hr} + 1$
- 9: **else if** $h \notin I$ and $h \neq p$ **then**
- 10: Drop the observation
- 11: **else if** $h = p$ **then**
- 12: **for** all $i, j \in I$ and $i \neq j$ **do**
- 13: **if** $N_{ir} \geq n_0$ and $N_{jr} \geq n_0$ **then**
- 14: Let $\tau_{ij,r} = \left[\frac{S_i^2(N_{ir})}{N_{ir}} + \frac{S_j^2(N_{jr})}{N_{jr}} \right]^{-1}$, where $S_i^2(N_{ir})$ and $S_j^2(N_{jr})$ are the sample variance of alternatives i and j , respectively
- 15: **else**
- 16: Let $\tau_{ij,r} = 0$
- 17: **end if**
- 18: **end for**
- 19: Set $I \leftarrow \left\{ i : i \in I \text{ and } \tau_{ij,r} [\bar{Y}_i(N_{ir}) - \bar{Y}_j(N_{jr})] \geq \min \left\{ 0, -\frac{a}{\delta} + \frac{\delta}{2} \tau_{ij,r} \right\}, \forall j \in I, j \neq i \right\}$, where $\bar{Y}_i(N_{ir})$ and $\bar{Y}_j(N_{jr})$ are the sample mean of alternatives i and j , respectively
- 20: Set $r \leftarrow r + 1$
- 21: **end if**
- 22: **end while**
- 23: Select the alternative in I as the best

As mentioned but not addressed by Luo et al. (2015), all-pairwise comparisons conducted in the master could overwhelm the workload of the master, and the frequent communications between the master and workers could become a bottleneck for solving large-scale R&S problems. To reduce the comparisons, the good selection procedure (GSP) of Ni et al. (2017) proposes a “divide-and-conquer” approach to distributing all-pairwise comparisons among the workers. The master initially divides k alternatives into m groups and asks each worker to conduct local all-pairwise comparisons for eliminations within the

assigned group. Then, the computational complexity of comparisons at each round is reduced from $\mathcal{O}(k^2)$ to $\mathcal{O}(k^2/m^2)$. To improve the elimination efficiency further, the master retrieves the m local bests from the m groups at the beginning of each local comparison round to find the global best. Then, the master sends the global best to the m groups for additional comparisons.

To reduce the frequent communications, the GSP introduces a batching technique of samples. In particular, it suggests each worker to simulate a batch of 100 or 200 samples from one alternative at a time. In addition, when a surviving alternative takes enough samples, i.e., reaching a threshold, the procedure requires all surviving alternatives to take a maximum number of samples. Then, it selects the one with the largest sample mean as the best. By doing so, the GSP can significantly reduce the communication frequency. In fact, the maximum sample size in the boosting stage (i.e., Stage 3 in their procedure) is constructed based on the Rinott’s result. Moreover, the sequential elimination rule in Stage 2 is built on the results of Hong (2006). Taking advantage of both sequential and stage-wise frameworks, GSP is an excellent hybrid procedure that not only improves the comparison and communication efficiency but also provides a finite-time PGS guarantee of Eq. (8).

One potential drawback of GSP is its conservativeness in terms of total sample size due to the batching technique of samples and the error separation in the hybrid structure. In other words, GSP sacrifices a certain level of sampling efficiency to achieve lower computational complexities of comparisons and communications as well as the finite-time statistical guarantee. This drawback motivated Zhong et al. (2020) to design the parallel Paulson’s procedure (PPP), which takes a different approach to achieving simulation, comparison, and communication efficiency.

In terms of the simulation efficiency, PPP adopts the well-known sequential procedure of Paulson (1964), which often requires a smaller sample size than the stage-wise and the hybrid procedures in the same desired guarantee level. In terms of the comparison efficiency, PPP breaks all-pairwise comparisons into comparisons with the “best”, which reduces the computational complexity of comparisons from $\mathcal{O}(k^2)$ to $\mathcal{O}(k)$ at each comparison round. The “best” involves both the sample mean and sample variance information, which is different from the global best involving only the mean information defined in GSP. In terms of the communication efficiency, PPP uses a different batching technique, i.e., batching alternatives instead of samples, which can reduce the communication frequency without increasing the sample size. PPP also allows to boost the sample size to a maximum number in Paulson’s sequential framework. In addition, by incorporating the result of Kao and Lai (1980), PPP can also achieve the PGS guarantee.

We omit the presentation of both GSP and PPP.

Interested readers may refer to the works of the above-mentioned authors for more details. The CRNs technique is difficult to implement for VKN, APS, and GSP because of the asynchronized simulation scheme in Luo et al. (2015) and Ni et al. (2017). Moreover, CRNs are not suitable for PPP because of the newly designed comparison scheme in Zhong et al. (2020).

5.2 New parallel framework with sample-size optimality

The aforementioned procedures for parallel computing environments are all built on the paradigm of existing stage-wise or fully sequential R&S procedures. They have even successfully solved R&S problems with up to millions of alternatives. Therefore, we must ask whether they are fundamentally suitable for handling large-scale problems. More precisely, we would like to know how the expected total sample size $E[N]$ increases as the number of alternatives k increases.

To answer the question, Zhong and Hong (2020) first proved that the growth rate of $E[N]$ for any procedure with the PCS guarantee is lower bounded by $\mathcal{O}(k)$. Then, they defined the sample-size optimality of a procedure if the upper bound of the growth rate of $E[N]$ can achieve the lower bound rate, that is, $E[N] = \mathcal{O}(k)$. Intuitively speaking, the lower bound of $E[N]$ is easy to understand, as each alternative requires at least one observation to estimate the unknown mean, thus resulting in a total sample size growing at least linearly in k . The lower bound is universal for all stage-wise and fully sequential procedures in either the IZ or IZ-free framework (Zhong and Hong, 2020).

However, the upper bound is typically higher than the order of k for all existing stage-wise and fully sequential procedures. For instance, the expected total sample size of each alternative in Paulson's procedure grows proportionally to the ending point of the continuation region, which grows in the order of $\log k$, thus leading to $\mathcal{O}(k \log k)$ in the total sample size. This is because it needs to compare the best with all other $k-1$ alternatives in pairs in theoretical formulation, which is also true for other fully sequential procedures, such as the \mathcal{KN} procedure. Similarly, stage-wise procedures, e.g., one-stage procedure of Bechhofer (1954) and two-stage procedure of Rinott (1978), also need to compare the best with all other $k-1$ alternatives in a joint formulation, as in Eq. (12). Thus, the sample size of each alternative in Eq. (13) grows as k increases, which inevitably leads to a higher order of k in the total sample size. In other words, neither fully sequential nor stage-wise procedures can achieve the sample-size optimality due to the nature of comparisons between the best alternative and all others.

Inspired by the knockout tournament arrangement of tennis Grand Slam tournaments, Zhong and Hong (2020) proposed a novel parallel selection framework in which the champion (i.e., the unknown best) does not have to play

with all others to be declared as the best. In particular, the knockout tournament (\mathcal{KT}) procedures of Zhong and Hong (2020) divide all alternatives into pairs and construct a "match" between two alternatives in pairs, thus keeping the winner for the next round "matches" while knocking out the loser after the current round. By doing so, \mathcal{KT} procedures eliminate approximately half of the alternatives at each round and therefore achieve the theoretical lower bound of $E[N]$. The sample-size optimality is achieved regardless of whether the variances of the alternatives are known or not. However, it will only change the constants on the optimal upper bounds.

This structure of the \mathcal{KT} procedures is perfect for parallel computing in terms of synchronization, communication, and comparison efficiency. The \mathcal{KT} procedures can divide all alternatives into m subsets and assign each subset to one processor to conduct selections simultaneously among the alternatives in that subset. Then, neither synchronization nor communication is necessary among different processors until each processor produces a local best alternative. Given that all "matches" in each processor are conducted independently and locally, and CRNs technique can be easily implemented into \mathcal{KT} procedures. In addition, as comparisons are only made within "matches", Zhong and Hong (2020) demonstrated that the comparison time in the procedures is negligible compared with the simulation time. In fact, the number of comparisons is only half of the total sample size, as simulating two observations (i.e., one for each alternative) is coupled with just a single comparison.

In each "match", any existing R&S procedures can be used to determine the winner, and \mathcal{KT} procedures adopt the \mathcal{KN} procedure to achieve the PCS guarantee and to gain sampling efficiency on the total sample size. The sampling efficiency can be further improved by assigning more than two alternatives into one "match". The PGS guarantee can also be obtained by allocating the IZ parameter in different rounds.

For the simplicity of presentation, we adopt the notation of $\mathcal{KN}(\mathcal{C}, \alpha_r, \delta, n_0)$ in Zhong and Hong (2020) to denote the output of executing the \mathcal{KN} procedure in each "match", which provides a PCS of $1-\alpha_r$ among the alternatives with unknown means and unknown variances in the set \mathcal{C} when the first stage sample size is n_0 and the IZ parameter is δ . In the following, we present the \mathcal{KT}^+ , one of the \mathcal{KT} procedures for parallel computing environments of Zhong and Hong (2020), in Procedure 8.

We conclude this section by briefly reviewing some other recent works on parallel R&S problems. Hunter and Nelson (2017) argued that different performance measures and different formulations are needed for large-scale R&S problems. As a response, Pei et al. (2018) proposed a different objective function, i.e., the expected false elimination rate (EFER) rather than the PCS/PGS. They argued that having a sizeable number of alternatives is more reasonable. Ma (2018) extended the Envelop

Procedure 8 \mathcal{KT}^+ procedure

Require: Number of alternatives k , PCS $1-\alpha$, IZ parameter $\delta > 0$, the first-stage sample size $n_0 > 2$, the parameter λ ($0 < \lambda < \delta$), number of alternatives $g \geq 2$ within a “match”, and the number of processors $m + 1$ (i.e., one master and workers)

- 1: Let I_r^s be the set of alternatives in contention at the beginning of round r in processor s for $s = 1, 2, \dots, m$
- 2: Equally allocate k alternatives to m processors so that each processor handles the selection of approximately k/m alternatives, e.g., for $i = 1, 2, \dots, k$, let $I_1^{(i \bmod m)+1} = I_1^{(i \bmod m)+1} \cup \{i\}$
- 3: **for** all $s = 1, 2, \dots, m$ **do**
- 4: Set $r = 1$
- 5: **while** $|I_r^s| > 1$ **do**
- 6: Let $I_{r+1}^s = \emptyset$. Group alternatives in I_r^s with the size of g . In the case of leftover ones, let them form a group. After grouping, a total of $\lceil |I_r^s|/g \rceil$ groups are formed. Let $I_{r,q}^s$ denote the set of the alternatives in group q for $q = 1, 2, \dots, \lceil |I_r^s|/g \rceil$ of processor s at round r
- 7: Let $\alpha_r = \alpha/2^r$. For each group $q = 1, 2, \dots, \lceil |I_r^s|/g \rceil$, set $C = I_{r,q}^s$ and compute $I_{r+1}^s = I_{r,q}^s \cup \{\mathcal{KN}(\mathcal{C}, \alpha_r, \delta, n_0)\}$
- 8: Set $r = r + 1$
- 9: **end while**
- 10: Let I_s denote the index of the alternative in I_r^s
- 11: Take n_0 observations from alternative I_s . Calculate its sample variance $S_{I_s}^2$ based on these n_0 observations. Set $r = \left\lceil \log_g \frac{k}{m} \right\rceil + 1$, $\alpha_r = \alpha/2^r$, and $h(\alpha_r, m, n_0)$, which is the Rinott's constant determined by α_r, m and n_0
- 12: Set $N_{\max, I_s} = \max \left\{ n_0, \left\lceil \left(\frac{h(\alpha_r, m, n_0) S_{I_s}}{\delta} \right)^2 \right\rceil \right\}$. Then, take additional $N_{\max, I_s} - n_0$ observations from alternative I_s
- 13: Compute the sample mean $\bar{X}_{I_s}(N_{\max, I_s})$
- 14: **end for**
- 15: Let I denote the set of alternatives containing all the best alternatives produced by m processors. Select the alternative with the largest sample mean $\bar{X}_{I_s}(N_{\max, I_s})$ for $I_s \in I$

procedure of Ma and Henderson (2017) to parallel computing environments, which established a new error bound of Brownian motion processes inspired by the multi-armed bandit problem of Jamieson et al. (2014). Notice that these abovementioned procedures using parallel computing are fixed-precision procedures. Some of the well-known fixed-budget R&S procedures have also been adapted to parallel computing environments. For instance, Kamiński and Szufel (2018) considered the asynchronization issue as extending both the OCBA and KG procedures in parallel computing environments and discussed the efficiency of these procedures for small- and large-scale problems.

6 Emerging R&S problems

Besides fixed-precision and fixed-budget R&S problems,

some research problems that expend classical R&S from different perspectives have emerged, e.g., considering multiple performance measures, taking the input uncertainty into account, and treating the performance measure as a function of the underlying contexts. In the following, we briefly discuss recent achievements in these topics without presenting the detailed procedures.

6.1 Constrained R&S

Traditional R&S problems often focus on only one performance measure. However, in various practical situations, we may be interested in multiple performance measures. For instance, in service centers, managers are concerned about the expected cost and customer waiting times. In production systems, managers care about not only the expected throughput but also the associated product quality. One way to deal with multiple performance measures is to model the primary one as the objective and to model others as constraints. This leads to constrained R&S problems considered in the simulation literature.

The study by Andradóttir and Kim (2010) is one of the first works in this area. The authors indicated that the primary and secondary simulation outputs are i.i.d. bivariate Gaussian random vectors with unknown mean vector and covariance matrix. They extended the IZ formulation to this problem and designed fixed-precision procedures that are capable of solving the problem. Healey et al. (2014) further developed the work of Andradóttir and Kim (2010) to handle multiple secondary performance measures. Then, Healey et al. (2015) reconsidered the problem by taking the switching cost into consideration. Rather than modeling the secondary performance as a Gaussian random vector as all previous works, Hong et al. (2015a) perceived the secondary performance as a Bernoulli random variable and the secondary performance measure as a probability. Hence, they called the problem a chance-constrained R&S problem. They built a hypothesis test on the chance constraint, thus resulting in an efficient two-stage procedure that performs the feasibility check in the first stage and selects the best among all the sample feasible alternatives in the second stage.

Different from the IZ formulation of the constrained R&S mentioned above, Lee et al. (2012) addressed the problem under the OCBA framework. Meanwhile, Hunter and Pasupathy (2013) and Pasupathy et al. (2015) solved the problem based on the large deviations theory by analyzing the asymptotic rate of identifying the optimal feasible solution, which is later known as Sampling Criteria for Optimization using Rate Estimators (SCORE) simulation framework. Recently, Gao et al. (2019a) considered constrained R&S problems in the OCBA formulation, in which they used large deviations theory and incorporated a quadratic regression metamodel to improve the efficiency further.

6.2 Multi-objective R&S

Except for the constrained R&S formulation, another way to handle multiple performance measures is to treat them as simultaneous objectives to optimize, thereby giving rise to the multi-objective R&S formulation. Multi-objective R&S problems are different from the classic single-objective R&S problems mainly in two aspects. First, in the multi-objective problems, an alternative “dominating” another alternative means that it is better on all objectives. Therefore, the single “best” alternative that dominates all others may not exist, and the goal of multi-objective R&S turns to select the set of all non-dominated alternatives, as termed by the Pareto set. Second, two types of probabilities are defined to measure the errors when a dominated alternative is included into the Pareto set and a non-dominated alternative is excluded from the set.

Multi-objective R&S procedures are often designed based on traditional R&S procedures. Interested readers may see Hunter et al. (2019) for a detailed review. Lee et al. (2010) extended the OCBA procedures to find the sampling allocation rule that minimizes the weighted sum of the two types of error probabilities. Alternatively, Feldman and Hunter (2018) and Applegate et al. (2020) derived the optimal sampling allocation rule by adopting the SCORE framework. Branke and Zhang (2015) supplemented the EVI procedure. At each stage, the new procedure allocates samples to the alternative that most likely changes the observed Pareto set. Branke et al. (2016) further developed the KG procedures and allocated the sample at each stage such that the estimated expected Pareto set is the closest to the true Pareto set.

The multi-objective procedures above are all designed for fixed-budget. Meanwhile, the literature has also suggested several fixed-precision multi-objective procedures. Batur et al. (2018) formulated the mean-variance portfolio analysis as a bi-objective R&S problem and proposed a bi-objective procedure under the IZ formulation that controls both types of error probabilities. Wang and Wan (2017) designed a sequential IZ-free procedure for the multi-objective procedure based on the generalized sequential likelihood ratio test.

6.3 R&S with input uncertainty

In simulation studies, the input distributions are often estimated from the data and other information, thus having uncertainty in them. This uncertainty is called input uncertainty. For instance, when modeling the arrival process of an online service system, multiple plausible distributions can fit the input historical data, especially when the data set is not sufficiently large. When specifying the demand curve of a newly launched product, different managers may have varying beliefs on it. Any distribution of such items can be used as a proxy of the true input distribution. However, the corresponding “best” alternative

may be different. In other words, no single alternative may be the best for all the possible scenarios of the input distributions. Then, taking the uncertainty of the simulation model into consideration in making R&S decisions is an interesting problem in R&S.

Song et al. (2015) studied the impact of input uncertainty on the classic IZ procedures and found that a straightforward application of IZ procedures may fail to deliver the desired PCS in the presence of uncertainty. They further proposed an adjustment to provide an average PCS. However, this average PCS guarantee cannot be delivered for some configurations of the competing alternatives. Therefore, it is still necessary to design procedures that are able to address the R&S with input uncertainty.

To design such procedures, Fan et al. (2013) innovatively proposed a robust selection-of-the-best (RSB) framework. Particularly, the RSB formulation includes all the possible scenarios of input distribution into an ambiguity set and then takes a robust perspective to define the best alternative with respect to the worst-case mean performance measures over the ambiguity set. Fan et al. (2020) further improved their work and proposed both two-stage and sequential procedures that can achieve the user-specified PCS. These RSB procedures are also tested by a healthcare queueing system with both synthetic and real hospital data. Gao et al. (2017b) considered the RSB formulation under the OCBA framework, in which the approximated PCS is also measured by the worst-case performance. Besides the PCS guarantee, Wu and Zhou (2019) took the RSB formulation from the fixed-budget viewpoint into account, in which a joint budget for both collecting input data and running simulations is given in advance.

6.4 R&S with covariates

In some problems, the performance measure of an alternative may vary as a function of some observable random covariates, which are also known as side information, auxiliary quantities, or contextual variables. For instance, in healthcare management, the treatment outcome of a particular drug may depend on the patient’s biometric characteristics. In revenue management, the best assortment could vary according to customer segmentation. Then, the best alternative is not universal but depends on the value of underlying covariates (e.g., patient’s biometric characteristics or customer segmentation). This type of selection of the best problem is called R&S with covariates (R&S-C) or contextual R&S.

Reasonably defining and measuring a correct selection of the best is the first question that needs to be addressed. Shen et al. (2017) were the first to introduce several definitions of correct selection for R&S-C from the frequentist perspective. They first defined the conditional PCS, which is denoted by $PCS(x)$, as the probability of selecting the best alternative (more precisely, the good

alternative within IZ) for an individual whose random covariates (denoted by X) take the value x . Then, two forms of unconditional PCS are introduced. One is the average PCS, i.e., $E[\text{PCS}(X)]$. The other is the worst PCS, i.e., $\min_{x \in \Omega} \text{PCS}(x)$, where Ω is the support of X . In Shen et al. (2017; 2019), they assumed a linear model between the mean performance of an alternative and the corresponding covariates. Moreover, they developed fixed-precision procedures that can produce selection policies (mapping from covariates to alternative index) to achieve the desired targets of unconditional PCS. The IZ formulation in R&S-C is natural and critical, as the mean performance surfaces of alternatives may intersect somewhere and the performance of different alternatives at the neighborhood of intersection points can be arbitrarily close or equal. Li et al. (2018) adopted the R&S-C framework developed by Shen et al. (2017). However, they designed new selection procedures to accommodate the high-dimensional covariates and the general (nonlinear) dependence between the mean performance of alternatives and the covariates.

Fixed-budget R&S-C problems have also been tackled under the Bayesian framework, with the aim of adaptively allocating a given sampling budget to the alternatives and over the domain of covariates to find the best response across the entire domain of covariates efficiently. Hu and Ludkovski (2017) proposed to model the performance functions of alternatives as Gaussian random fields and used the expected improvement criteria to develop Bayesian procedures. Pearce and Branke (2017) followed the same setting in Hu and Ludkovski (2017) and proposed a KG-based sampling policy with a focus on efficiently estimating the expected improvement over a continuous domain. Zhang et al. (2020) extended the problem to a more general setting where the sampling noise can be heteroscedastic and the sampling cost at different locations can be different. More importantly, they provided a theoretical analysis of the asymptotic behavior of the KG based policy and proved that the best alternative as a function of the covariates will be identified almost surely as the number of samples grows. Gao et al. (2019b) considered the case where the covariates only take discrete values and designed an OCBA-based sampling policy that converges to the asymptotic optimal budget allocation rule.

7 Important research questions on R&S

In this section, we outline six R&S problems that we think are important and yet to be solved. We will explain why we believe these problems are important. Some of these problems have also been considered in the literature. However, we feel that they deserve more research attention.

Problem 1: Besides the knockout-tournament procedures and the median-elimination procedures introduced in

Section 5.2, are there other types of rate-optimal fixed-precision large-scale R&S procedures?

Reason: Large-scale R&S is at the center stage of today's R&S research because small-scale problems have been studied extensively in the literature, and moreover, they are typically easy to solve with today's computing resources. The sample-size optimality result of Zhong and Hong (2020) shows that large-scale problems are fundamentally different from small-scale problems, and many R&S procedures that are efficient for small-scale problems are not efficient for large-scale problems. Therefore, more procedures need to be proposed under different parallel computing frameworks to solve various large-scale R&S problems.

Problem 2: Are there rate-optimal fixed-budget large-scale R&S procedures?

Reason: Fixed-budget R&S procedures typically show that their sample-allocation scheme converges to the optimal scheme of Glynn and Juneja (2004), as shown in Section 4. However, the optimal scheme depends heavily on the asymptotic regime, i.e., the number of alternatives stays the same and the total sample size reaches infinity. If the number of alternatives also goes to infinity, as in Zhong and Hong (2020), it is no longer optimal. Indeed, the optimal rate of Zhong and Hong (2020) also applies to fixed-budget R&S problems. Therefore, fixed-budget large-scale R&S procedures that are both rate-optimal and efficient in solving practical large-scale problems must be designed.

Problem 3: How can we design effective dynamic-programming-based procedures to solve fixed-budget R&S problems?

Reason: As we have shown in Section 4, fixed-budget R&S problems are in essence finite-time stochastic dynamic programs. However, procedures in the literature are primarily static-allocation approximations or one-step-look-ahead approximations. There appears no serious attempt to directly solve the dynamic programs. However, under Bayesian formulation, the posterior distributions are Gaussian distributions that can be simulated very easily. Therefore, Monte-Carlo-simulation-based approximate dynamic programming (ADP) or reinforcement learning techniques that consider multiple steps seem applicable. Of course, one also has to demonstrate or quantify both theoretically and numerically that going beyond a single step may bring actual benefit.

Problem 4: How can we design fixed-budget R&S procedures that are suitable for parallel computing environments?

Reason: As of now, the research attention on parallel R&S seems primarily focused on fixed-precision procedures. Fixed-budget procedures are often based on dynamic-programming formulation, which requires a significant amount of communications among alternatives to determine a sample-allocation policy. Thoughtful

approaches need be proposed to avoid excessive synchronizations and communications in order to implement fixed-budget procedures in parallel computing environments.

Problem 5: Do many fixed-precision elimination procedures (e.g., Paulson's, \mathcal{KN} and \mathcal{KT}) that satisfy PCS guarantee also satisfy PGS guarantee?

Reason: As we reviewed in Section 2, the PCS guarantee requires only a single best, which must be at least δ larger than all other alternatives. Determining whether a practical problem satisfies this requirement is generally very difficult. Hence, a PGS guarantee that selects an alternative in the indifference zone is certainly more reasonable. However, several fixed-precision elimination procedures (e.g., Paulson's, \mathcal{KN} and \mathcal{KT}) that satisfy the PCS guarantee cannot be proven to satisfy the PGS guarantee. Some of them may be adjusted to satisfy such guarantee at the cost of significantly larger sample sizes (Eckman and Henderson, 2018a). To the best of our knowledge, empirical evidence has shown that Paulson's, \mathcal{KN} and \mathcal{KT} always satisfy the PGS guarantee. Hence, we wonder whether they actually satisfy the PGS guarantee or at least do under some conditions, e.g., when the number of alternatives is large.

Problem 6: How can we better integrate R&S into optimization-via-simulation (OvS) algorithms?

Reason: Many OvS algorithms require either keeping the current sample best solution or selecting from a group of neighboring solutions. These requirements are naturally R&S problems. Indeed, a number of R&S procedures have been proposed to work with OvS algorithms. For instance, Boesel et al. (2003) proposed using R&S at the end of the OvS process to select the best from all visited solutions, which they called "clean-up". Pichtlamken et al. (2006) propose to use R&S in neighborhood selection. Meanwhile, Hong and Nelson (2007a) considered methods to ensure that the current best discovered by the OvS algorithm is indeed the best at any time of the OvS process. In addition, He et al. (2010) and Zhang et al. (2017) incorporated the OBCA procedures into the cross-entropy and particle swarm OvS algorithms, respectively. However, besides the clean-up procedure, which requires extra work after the OvS process is done and provides no information for OvS algorithms to find a better solution, other ideas tend to slow down the optimization process significantly and output considerably worse solutions. Therefore, it is interesting to figure out how to integrate R&S into simulation optimization algorithms so that the optimization process may benefit from R&S. Recently, Eckman and Henderson (2018b) showed that the reuse of the simulation observations from the optimization process (also called the search process) by a clean-up procedure at the end of the optimization process may jeopardize the PCS/PGS guarantee. However, without reusing the simulation observations, R&S-based clean-up procedures may require a significant amount of simulation observa-

tions that may be too expensive to conduct in practice. Therefore, resolving this issue also imposes a theoretical challenge when integrating R&S into OvS algorithms.

Naturally, the R&S field has numerous other interesting research problems and emerging research topics. As computing resources are becoming widely available, in general, increasingly complicated R&S problems need to be solved.

Acknowledgements The authors would like to thank Prof. Shane Henderson for his insightful and detailed comments that have significantly improved this paper.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made.

The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Andradóttir S, Kim S H (2010). Fully sequential procedures for comparing constrained systems via simulation. *Naval Research Logistics*, 57(5): 403–421
- Applegate E A, Feldman G, Hunter S R, Pasupathy R (2020). Multi-objective ranking and selection: Optimal sampling laws and tractable approximations via SCORE. *Journal of Simulation*, 14(1): 21–40
- Banerjee S (1961). On confidence interval for two-means problem based on separate estimates of variances and tabulated values of *t*-table. *Sankhya Series A*, 23(4): 359–378
- Batur D, Choobineh F (2012). Stochastic dominance based comparison for system selection. *European Journal of Operational Research*, 220(3): 661–672
- Batur D, Wang L, Choobineh F F (2018). Methods for systems selection based on sequential mean-variance analysis. *INFORMS Journal on Computing*, 30(4): 724–738
- Bechhofer R E (1954). A single-sample multiple decision procedure for ranking means of normal populations with known variances. *Annals of Mathematical Statistics*, 25(1): 16–39
- Bechhofer R E, Santner T J, Goldsman D M (1995). *Design and Analysis of Experiments for Statistical Selection, Screening, Multiple Comparisons*. Hoboken, NJ: Wiley
- Bellman R (1966). *Dynamic programming*. *Science*, 153(3731): 34–37
- Bertsekas D P (1995). *Dynamic Programming and Optimal Control*, vol. 1. 4th ed. Belmont, MA: Athena Scientific
- Boesel J, Nelson B L, Kim S H (2003). Using ranking and selection to "clean up" after simulation optimization. *Operations Research*, 51(5): 814–825
- Branke J, Chick S E, Schmidt C (2007). Selecting a selection procedure.

- Management Science, 53(12): 1916–1932
- Branke J, Zhang W (2015). A new myopic sequential sampling algorithm for multi-objective problems. In: Proceedings of the Winter Simulation Conference. IEEE, 3589–3598
- Branke J, Zhang W, Tao Y (2016). Multiobjective ranking and selection based on hypervolume. In: Proceedings of the Winter Simulation Conference. IEEE, 859–870
- Bubeck S, Cesa-Bianchi N (2012). Regret analysis of stochastic and nonstochastic multi-armed bandit problems. arXiv Preprint. arXiv:1204.5721
- Chen C H (1996). A lower bound for the correct subset-selection probability and its application to discrete-event system simulations. IEEE Transactions on Automatic Control, 41(8): 1227–1231
- Chen C H, Lin J, Yücesan E, Chick S E (2000). Simulation budget allocation for further enhancing the efficiency of ordinal optimization. Discrete Event Dynamic Systems, 10(3): 251–270
- Chen E J (2005). Using parallel and distributed computing to increase the capability of selection procedures. In: Proceedings of the Winter Simulation Conference. IEEE, 723–731
- Chick S E (2006). Subjective probability and Bayesian methodology. In: Henderson S G, Nelson B L, eds. Handbooks in Operations Research and Management Science: Simulation, Chapter 9. Amsterdam, Elsevier, 13: 225–257
- Chick S E, Branke J, Schmidt C (2010). Sequential sampling to myopically maximize the expected value of information. INFORMS Journal on Computing, 22(1): 71–80
- Chick S E, Inoue K (2001). New two-stage and sequential procedures for selecting the best simulated system. Operations Research, 49(5): 732–743
- Clark G M, Yang W (1986). A bonferroni selection procedure when using common random numbers with unknown variances. In: Proceedings of the Winter Simulation Conference. IEEE, 313–315
- Dean J, Ghemawat S (2008). MapReduce: Simplified data processing on large clusters. Communications of the ACM, 51(1): 107–113
- Eckman D J, Henderson S G (2018a). Guarantees on the probability of good selection. In: Proceedings of the Winter Simulation Conference. IEEE, 351–365
- Eckman D J, Henderson S G (2018b). Reusing search data in ranking and selection: What could possibly go wrong? ACM Transactions on Modeling and Computer Simulation, 28(3): 1–15
- Eckman D J, Henderson S G (2020). Fixed-confidence, fixed-tolerance guarantees for selection-of-the-best procedures. ACM Transactions on Modeling and Computer Simulation, 31(2): 7
- Even-Dar E, Mannor S, Mansour Y (2002). PAC bounds for multi-armed bandit and Markov decision processes. In: COLT '02: Proceedings of the 15th Annual Conference on Computational Learning Theory. Springer-Verlag, 255–270
- Even-Dar E, Mannor S, Mansour Y (2006). Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems. Journal of Machine Learning Research, 7(39): 1079–1105
- Fabian V (1974). Note on Anderson's sequential procedures with triangular boundary. Annals of Statistics, 2(1): 170–176
- Fan W W, Hong L J, Nelson B L (2016). Indifference-zone-free selection of the best. Operations Research, 64(6): 1499–1514
- Fan W W, Hong L J, Zhang X W (2013). Robust selection of the best. In: Proceedings of the Winter Simulation Conference. IEEE, 868–876
- Fan W W, Hong L J, Zhang X W (2020). Distributionally robust selection of the best. Management Science, 66(1): 190–208
- Feldman G, Hunter S R (2018). SCORE allocations for bi-objective ranking and selection. ACM Transactions on Modeling and Computer Simulation, 28(1): 1–28
- Frazier P I (2014). 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
- Frazier P I, Powell W B, Dayanik S (2008). A knowledge-gradient policy for sequential information collection. SIAM Journal on Control and Optimization, 47(5): 2410–2439
- Frazier P I, Powell W B, Dayanik S (2009). The knowledge-gradient policy for correlated normal beliefs. INFORMS Journal on Computing, 21(4): 599–613
- Fu M C, Henderson S G (2017). History of seeking better solutions, aka simulation optimization. In: Proceedings of the Winter Simulation Conference. IEEE, 131–157
- Fu M C, Hu J Q, Chen C H, Xiong X (2007). Simulation allocation for determining the best design in the presence of correlated sampling. INFORMS Journal on Computing, 19(1): 101–111
- Gabillon V, Ghavamzadeh M, Lazaric A (2012). Best arm identification: A unified approach to fixed budget and fixed confidence. In: Pereira F, Burges C J C, Bottou L, Weinberger K Q, eds. Advances in Neural Information Processing Systems. Minneapolis, MN: Curran Associates, Inc., 25: 3212–3220
- Gao F, Gao S, Xiao H, Shi Z (2019a). Advancing constrained ranking and selection with regression in partitioned domains. IEEE Transactions on Automation Science and Engineering, 16(1): 382–391
- Gao S, Chen W, Shi L (2017a). A new budget allocation framework for the expected opportunity cost. Operations Research, 65(3): 787–803
- Gao S, Du J, Chen C H (2019b). Selecting the optimal system design under covariates. In: 15th International Conference on Automation Science and Engineering (CASE). IEEE, 547–552
- Gao S, Xiao H, Zhou E, Chen W (2017b). Robust ranking and selection with optimal computing budget allocation. Automatica, 81: 30–36
- Glynn P, Juneja S (2004). A large deviations perspective on ordinal optimization. In: Proceedings of the Winter Simulation Conference. IEEE, 577–585
- Glynn P, Juneja S (2018). Selecting the best system and multi-armed bandits. arXiv Preprint. arXiv:1507.04564
- Goldsman D, Nelson B L, Banks J (1998). Comparing systems via simulation. In: Banks J, ed. Handbook of Simulation: Principles, Methodology, Advances, Applications, Practice. Hoboken, NJ: Wiley, 273–306
- Gropp W, Lusk E, Skjellum A (1999). Using MPI: Portable Parallel Programming with the Message-Passing Interface, vol. 1. Cambridge, MA: MIT press
- Gupta S (1956). On a Decision Rule for a Problem in Ranking Means. Dissertation for the Doctoral Degree. Chapel Hill, NC: University of North Carolina
- He D, Chick S E, Chen C H (2007). Opportunity cost and OCBA selection procedures in ordinal optimization for a fixed number of alternative systems. IEEE Transactions on Systems, Man and Cybernetics: Part C, Applications and Reviews, 37(5): 951–961

- He D, Lee L H, Chen C H, Fu M C, Wasserkrug S (2010). Simulation optimization using the cross-entropy method with optimal computing budget allocation. *ACM Transactions on Modeling and Computer Simulation*, 20(1): 1–22
- Healey C M, Andradóttir S, Kim S H (2014). Selection procedures for simulations with multiple constraints under independent and correlated sampling. *ACM Transactions on Modeling and Computer Simulation*, 24(3): 14
- Healey C M, Andradóttir S, Kim S H (2015). A minimal switching procedure for constrained ranking and selection under independent or common random numbers. *IIE Transactions*, 47(11): 1170–1184
- Hong L J (2006). Fully sequential indifference-zone selection procedures with variance-dependent sampling. *Naval Research Logistics*, 53(5): 464–476
- Hong L J, Luo J, Nelson B L (2015a). Chance constrained selection of the best. *INFORMS Journal on Computing*, 27(2): 317–334
- Hong L J, Nelson B L (2005). The tradeoff between sampling and switching: New sequential procedures for indifference-zone selection. *IIE Transactions*, 37(7): 623–634
- Hong L J, Nelson B L (2007a). A framework for locally convergent random-search algorithms for discrete optimization via simulation. *ACM Transactions on Modeling and Computer Simulation*, 17(4): 19
- Hong L J, Nelson B L (2007b). Selecting the best system when systems are revealed sequentially. *IIE Transactions*, 39(7): 723–734
- Hong L J, Nelson B L (2009). A brief introduction to optimization via simulation. In: *Proceedings of the Winter Simulation Conference*. IEEE, 75–85
- Hong L J, Nelson B L, Xu J (2015b). Discrete optimization via simulation. In: Fu M C, ed. *Handbook of Simulation Optimization*. Berlin: Springer
- Hu R, Ludkovski M (2017). Sequential design for ranking response surfaces. *SIAM/ASA Journal on Uncertainty Quantification*, 5(1): 212–239
- Hunter S R, Applegate E A, Arora V, Chong B, Cooper K, Rincón-Guevara O, Vivas-Valencia C (2019). An introduction to multi-objective simulation optimization. *ACM Transactions on Modeling and Computer Simulation*, 29(1): 1–36
- Hunter S R, Nelson B L (2017). Parallel ranking and selection. In: Tolk A, Fowler J, Shao G, Yücesan E, eds. *Advances in Modeling and Simulation*. Berlin: Springer, 249–275
- Hunter S R, Pasupathy R (2013). Optimal sampling laws for stochastically constrained simulation optimization on finite sets. *INFORMS Journal on Computing*, 25(3): 527–542
- Jamieson K, Malloy M, Nowak R, Bubeck S (2014). *lil'UCB*: An optimal exploration algorithm for multi-armed bandits. In: *Conference on Learning Theory*, 423–439
- Kamiński B, Szufel P (2018). On parallel policies for ranking and selection problems. *Journal of Applied Statistics*, 45(9): 1690–1713
- Kao S C, Lai T L (1980). Sequential selection procedures based on confidence sequences for normal populations. *Communications in Statistics: Theory and Methods*, 9(16): 1657–1676
- Kaufmann E, Kalyanakrishnan S (2013). Information complexity in bandit subset selection. *Journal of Machine Learning Research*, 30: 228–251
- Kim S H, Nelson B L (2001). A fully sequential procedure for indifference-zone selection in simulation. *ACM Transactions on Modeling and Computer Simulation*, 11(3): 251–273
- Kim S H, Nelson B L (2006a). On the asymptotic validity of fully sequential selection procedures for steady-state simulation. *Operations Research*, 54(3): 475–488
- Kim S H, Nelson B L (2006b). Selecting the best system. In: Henderson S G, Nelson B L, eds. *Handbooks in Operations Research and Management Science: Simulation*. Amsterdam: Elsevier, 501–534
- Lee L H, Chew E P, Teng S, Goldsman D (2010). Finding the non-dominated Pareto set for multi-objective simulation models. *IIE Transactions*, 42(9): 656–674
- Lee L H, Pujowidianto N A, Li L W, Chen C H, Yap C M (2012). Approximate simulation budget allocation for selecting the best design in the presence of stochastic constraints. *IEEE Transactions on Automatic Control*, 57(11): 2940–2945
- Li X, Zhang X, Zheng Z (2018). Data-driven ranking and selection: High-dimensional covariates and general dependence. In: *Proceedings of the Winter Simulation Conference*. IEEE, 1933–1944
- Luo J, Hong L J (2011). Large-scale ranking and selection using cloud computing. In: *Proceedings of the Winter Simulation Conference*. IEEE, 4051–4061
- Luo J, Hong L J, Nelson B L, Wu Y (2015). Fully sequential procedures for large-scale ranking-and-selection problems in parallel computing environments. *Operations Research*, 63(5): 1177–1194
- Ma S (2018). *Sequential Ranking and Selection Procedures and Sample Complexity*. Dissertation for the Doctoral Degree. New York: Cornell University
- Ma S, Henderson S G (2017). An efficient fully sequential selection procedure guaranteeing probably approximately correct selection. In: *Proceedings of the Winter Simulation Conference*. IEEE, 2225–2236
- Nelson B L, Matejcek F J (1995). Using common random numbers for indifference-zone selection and multiple comparisons in simulation. *Management Science*, 41(12): 1935–1945
- Nelson B L, Swann J, Goldsman D, Song W (2001). Simple procedures for selecting the best simulated system when the number of alternatives is large. *Operations Research*, 49(6): 950–963
- Ni E C, Ciocan D F, Henderson S G, Hunter S R (2015). Comparing message passing interface and MapReduce for large-scale parallel ranking and selection. In: *Proceedings of the Winter Simulation Conference*. IEEE, 3858–3867
- Ni E C, Ciocan D F, Henderson S G, Hunter S R (2017). Efficient ranking and selection in parallel computing environments. *Operations Research*, 65(3): 821–836
- Ni E C, Hunter S R, Henderson S G (2013). Ranking and selection in a high performance computing environment. In: *Proceedings of the Winter Simulation Conference*. IEEE, 833–845
- Ni E C, Hunter S R, Henderson S G (2014). A comparison of two parallel ranking and selection procedures. In: *Proceedings of the Winter Simulation Conference*. IEEE, 3761–3772
- Pasupathy R, Hunter S R, Pujowidianto N A, Lee L H, Chen C H (2015). Stochastically constrained ranking and selection via SCORE. *ACM Transactions on Modeling and Computer Simulation*, 25(1): 1–26
- Paulson E (1949). A multiple decision procedure for certain problems in the analysis of variance. *Annals of Mathematical Statistics*, 20(1): 95–98
- Paulson E (1964). A sequential procedure for selecting the population with the largest mean from k . *Annals of Mathematical Statistics*, 35

- (1): 174–180
- Pearce M, Branke J (2017). Efficient expected improvement estimation for continuous multiple ranking and selection. In: Proceedings of the Winter Simulation Conference. IEEE, 2161–2172
- Pei L, Nelson B L, Hunter S (2018). A new framework for parallel ranking & selection using an adaptive standard. In: Proceedings of the Winter Simulation Conference. IEEE, 2201–2212
- Peng Y, Chong E K, Chen C H, Fu M C (2018). Ranking and selection as stochastic control. *IEEE Transactions on Automatic Control*, 63(8): 2359–2373
- Peng Y, Fu M C (2017). Myopic allocation policy with asymptotically optimal sampling rate. *IEEE Transactions on Automatic Control*, 62(4): 2041–2047
- Pichitlamken J, Nelson B L, Hong L J (2006). A sequential procedure for neighborhood selection-of-the-best in optimization via simulation. *European Journal of Operational Research*, 173(1): 283–298
- Rinott Y (1978). On two-stage selection procedures and related probability-inequalities. *Communications in Statistics: Theory and Methods*, 7(8): 799–811
- Ryzhov I O (2016). On the convergence rates of expected improvement methods. *Operations Research*, 64(6): 1515–1528
- Shen H, Hong L J, Zhang X (2017). Ranking and selection with covariates. In: Proceedings of the Winter Simulation Conference. IEEE, 169
- Shen H, Hong L J, Zhang X (2019). Ranking and selection with covariates for personalized decision making. *arXiv Preprint. arXiv:1710.02642*
- Song E, Nelson B L, Hong L J (2015). Input uncertainty and indifference-zone ranking & selection. In: Proceedings of the Winter Simulation Conference. IEEE, 414–424
- Tsai S C, Luo J, Sung C C (2017). Combined variance reduction techniques in fully sequential selection procedures. *Naval Research Logistics*, 64(6): 502–527
- Tsai S C, Nelson B L (2009). Fully sequential selection procedures with control variates. *IIE Transactions*, 42(1): 71–82
- Wald A (1945). Sequential tests of statistical hypotheses. *Annals of Mathematical Statistics*, 16(2): 117–186
- Wald A (1947). *Sequential Analysis*. New York: John Wiley & Sons
- Wang W, Wan H (2017). Sequential probability ratio test for multiple-objective ranking and selection. In: Proceedings of the Winter Simulation Conference. IEEE, 1998–2009
- Wilcox R R (1984). A table for Rinott’s selection procedure. *Journal of Quality Technology*, 16(2): 97–100
- Wu D, Zhou E (2019). Fixed confidence ranking and selection under input uncertainty. In: Proceedings of the Winter Simulation Conference. IEEE, 3717–3727
- Xie J, Frazier P, Chick S E (2016). Bayesian optimization via simulation with pairwise sampling and correlated prior beliefs. *Operations Research*, 64(2): 542–559
- Yücesan E, Luo Y C, Chen C H, Lee I (2001). Distributed web-based simulation experiments for optimization. *Simulation Practice and Theory*, 9(1–2): 73–90
- Zaharia M, Chowdhury M, Franklin M J, Shenker S, Stoica I (2010). Spark: Cluster computing with working sets. In: *HotCloud’10: Proceedings of the 2nd USENIX Conference on Hot Topics in Cloud Computing*. Berkeley, CA, 10
- Zhang S, Xu J, Lee L H, Chew E P, Wong W P, Chen C H (2017). Optimal computing budget allocation for particle swarm optimization in stochastic optimization. *IEEE Transactions on Evolutionary Computation*, 21(2): 206–219
- Zhang X, Shen H, Hong L J, Ding L (2020). Knowledge gradient for selection with covariates: Consistency and computation. *arXiv Preprint. arXiv:1906.05098*
- Zhong Y, Hong L J (2020). Knockout-tournament procedures for large-scale ranking and selection in parallel computing environments. *Operations Research*, to appear
- Zhong Y, Liu S, Luo J, Hong L J (2020). Speeding up Paulson’s procedure for large-scale problems using parallel computing. *INFORMS Journal on Computing*, in press, doi: 10.1287/ijoc.2020.1054