Solving Large-Scale Fixed-Budget Ranking and Selection Problems

Preprint · September 2021

3 authors:

L. Jeff Hong
Fudan University

Guangxin Jiang
Harbin Institute of Technology

Ying Zhong
University of Electronic Science and Technology of China

Solving Large-Scale Fixed-Budget Ranking and Selection Problems

L. Jeff Hong
School of Management and School of Data Science, Fudan University, Shanghai, 200433 China, hong_liu@fudan.edu.cn

Guangxin Jiang
School of Management, Harbin Institute of Technology, Harbin, 150001 China, gxjiang@hit.edu.cn

Ying Zhong*
School of Management and Economics, University of Electronic Science and Technology of China, Chengdu, 611731 China, yzhong4@uestc.edu.cn

In recent years, with the rapid development of computing technology, developing parallel procedures to solve large-scale ranking and selection (R&S) problems has attracted a lot of research attention. In this paper, we take fixed-budget R&S procedure as an example to investigate potential issues of developing parallel procedures. We argue that to measure the performance of a fixed-budget R&S procedure in solving large-scale problems, it is important to quantify the minimal growth rate of the total sampling budget such that as the number of alternatives increases, the *probability of correct selection* (PCS) would not decrease to zero. We call such a growth rate of the total sampling budget the *rate for maintaining correct selection* (RMCS). We show that a tight lower bound for the RMCS of a broad class of existing fixed-budget procedures is in the order of $k \log k$. Then, we propose a new type of fixed-budget procedure, namely the fixed-budget knockout-tournament ($\mathcal{FBKT}$) procedure. We prove that, in terms of the RMCS, our procedure outperforms existing fixed-budget procedures and achieves the optimal order, i.e., the order of k. Moreover, we demonstrate that our procedure can be easily implemented in parallel computing environments with almost no non-parallelizable calculations. Lastly, a comprehensive numerical study shows that our procedure is indeed suitable for solving large-scale problems in parallel computing environments.

1. Introduction

Ranking and selection (R&S) is a fundamental problem in the area of stochastic simulation, which aims to select the alternative with the largest mean performance from a finite set of k alternatives, through conducting simulation experiments and observing the random outputs from every alternative. The problem was first considered by Bechhofer (1954) and has since then been studied extensively in the literature, see Bechhofer et al. (1995), Chick (2006), Kim and Nelson (2006), and Hong et al. (2021) for comprehensive reviews. The procedures designed to solve the problem have

* Corresponding Author

been used in production management (Luo et al. 2015), healthcare management (Fan et al. 2020, Shen et al. 2021), financial risk management (Lan et al. 2010, Waeber et al. 2010), transportation (Xiao et al. 2021), etc, and have been incorporated in commercial simulation software packages, such as Arena and Simio.

Due to the random outputs from every alternative, all R&S procedures need to repeatedly generate simulation observations (i.e., take samples) from all alternatives to estimate their mean performances, compare them, and identify the best. The quality of the selection is often measured by the probability that the best alternative is selected, i.e., the *probability of correct selection* (PCS). Typically, it is impossible for a procedure to deliver a PCS of one for any finite sampling budget. Therefore, while developing procedures, some compromises have to be made. Depending on the chosen compromises, existing procedures may be classified into two categories: fixed-budget procedures and fixed-precision procedures (Hunter and Nelson 2017). A fixed-budget procedure often attempts to allocate a fixed sampling budget N to different alternatives in a way that the PCS or a similar measure can be optimized (Chen et al. 2000, Glynn and Juneja 2004, Chick and Inoue 2001, Frazier et al. 2008). By contrast, for a fixed-precision procedure, there are no strict restrictions on the total sampling budget. The procedure is often developed to deliver a PCS higher than a pre-determined level upon stopping (Dudewicz and Dalal 1975, Rinott 1978, Paulson 1964, Kim and Nelson 2001, Fan et al. 2016).

Traditionally, R&S procedures were designed to solve small-scale problems, e.g., $k \leq 1,000$. It is because, in order to obtain a meaningful estimate of the best alternative, one needs to simulate every alternative many times. When the number of alternatives becomes large, the limited computational power of a single processor would restrict one from solving the problem in a reasonable amount of time. With the rapid development of computing technology, parallel computing environments are becoming prevalent and ready for ordinary users to access. Leveraging the computational power offered by parallel computing environments to solve large-scale problems has attracted a lot of research attention in recent years. However, developing parallel procedures to solve large-scale problems is non-trivial. As pointed out by Zhong and Hong (2021), when there are a large number of alternatives, it becomes theoretically important to understand the growth rate of the computational effort of a procedure with respect to the number of alternatives, because a procedure with a lower growth rate tends to outperform the procedure with a higher growth rate when the number of alternatives is sufficiently large. Practically, different from single-processor computing environments, in parallel computing environments, the non-parallelizable calculations of a procedure would significantly slow down the selection process and make the procedure inefficient. While developing parallel procedures, one should minimize the proportion of these calculations. Under the fixed-precision formulation, several parallel procedures have been proposed (Luo et al.

2015, Ni et al. 2017, Zhong and Hong 2021, Zhong et al. 2021). These studies show that if the aforementioned issues are addressed properly, the proposed procedures can solve problems with more than 10^4 and even up to 10^6 alternatives. However, developing parallel procedures to solve large-scale fixed-budget problems has seldom been discussed in the literature. In this paper, we aim to fill this gap.

So far, two types of fixed-budget procedures have been developed. One type of procedures, i.e., the optimal computing budget allocation (OCBA) procedure (Chen et al. 2000) and its variants (Chen et al. 2003, Glynn and Juneja 2004), determine the allocation of the sampling budget by solving a single static optimization problem. To solve the optimization problem, some approximations are needed. An important theoretical result for this type of procedures, developed by Chen et al. (2000) and Glynn and Juneja (2004), is that, if $N \to \infty$ and the Bonferroni inequality is used to approximate the PCS, the optimization problem admits a closed-form solution expressed as the sampling ratio between any two alternatives. Notice that the solution depends on the true values of the alternatives. As a result, in a practical implementation of this type of procedures, sample information of the alternatives is often plugged in to estimate the solution. Instead of trying to maximize the PCS directly, the other type of procedures, i.e., the expected improvement (EI) type of procedures (Chick and Inoue 2001, Chick et al. 2010, Frazier et al. 2008), consider minimizing the difference between the means of the selected alternative and the best alternative, i.e., the *expected opportunity cost* (EOC). This type of procedures are often developed under the Bayesian framework. The mean of each alternative is represented by a random variable. The random variable has a predictive distribution that characterizes one's belief about the unknown mean of each alternative and changes over time as more information is obtained from observations. These procedures often allocate observations one at a time in an adaptive manner. During the selection process, regarding the allocation of an observation, the procedures need to solve a one-step lookahead optimization problem.

Even though numerous fixed-budget procedures have been developed, these procedures may not be suitable for solving large-scale problems in parallel computing environments from both theoretical and practical aspects. Theoretically, we argue that to measure the performance of a fixed-budget procedure in solving large-scale problems, it is important to quantify the minimal growth rate of the total sampling budget N such that as the number of alternatives (i.e., k) increases, the PCS would not decrease to zero. The lower the minimal growth rate, the higher PCS a procedure tends to deliver while solving large-scale problems. It provides a theoretical framework to compare the performances of different fixed-budget procedures in the context of large-scale R&S. We call this minimal growth rate of the total sampling budget the *rate for maintaining correct selection* (RMCS). Intuitively, the RMCS of existing procedures should be in a higher order than k. It is because, as a common feature of existing procedures, during the selection process, no alternatives are eliminated, and when the selection stops, the alternative with the largest sample mean is selected. For these procedures, the PCS equals the probability that the best alternative has the largest sample mean among all k alternatives. In order to be selected, the best alternative essentially needs to compete with all other alternatives. As the number of alternatives k increases, it becomes harder to ensure that the best alternative beats all other alternatives. Therefore, to keep the PCS from decreasing to zero, more observations should be allocated to each alternative, and the total sampling budget N should grow in a higher order than k. In this paper, we take the analysis one-step further and establish a tight lower bound for the RMCS of a broad class of existing procedures. Specifically, we consider a sample allocation which provides the performance upper bound for a broad class of existing procedures. We show that even if such a sample allocation is adopted, as the number of alternatives increases, to keep the PCS from decreasing to zero, the total sampling budget N should grow at least in the order of $k \log k$. Therefore, it is of theoretical interest to investigate whether we can develop a procedure that outperforms existing procedures in terms of the RMCS.

Practically, the essence of parallel computing is to decompose a big computational task into small ones and assign them to different processors to speedup the execution time. The non-parallelizable part of a computer program quantifies the upper limit of the speedup. For example, suppose that 10% of a program can only be done on a single processor. In that case, regardless of how many processors are devoted to a parallelized execution of the program, the theoretical maximum speedup using parallel computing is limited to 10 times. The smaller proportion of the non-parallelizable part of a program, the better fit the program is for parallel computing. The selection structures of most existing fixed-budget procedures are serial in nature. During the selection process, the OCBA type of procedures need to frequently update the sample information of different alternatives to adjust their allocation schemes, and the EI type of procedures need to solve a sequence of one-step lookahead optimization problems. In some extreme cases, the procedures need to update or even optimize with every newly collected observation. When the number of alternatives gets large, the efforts spent on these calculations increase rapidly because significantly more observations need to be allocated. These calculations are non-parallelizable and can only be done on a single processor. It can be expected that, if these procedures are implemented in parallel computing environments to solve large-scale problems, such non-parallelizable calculations can significantly slow down the overall selection process and ruin the benefits of parallel computing. An ideal procedure for parallel computing should minimize the proportion of non-parallelizable calculations.

In this paper, inspired by the knockout-tournament ($\mathcal{KT}$) procedure proposed by Zhong and Hong (2021) to solve large-scale fixed-precision problems in parallel computing environments, we develop a new type of fixed-budget procedure. Compared to existing procedures, our procedure has a distinct selection structure. Specifically, the procedure conducts the selection round-by-round. At the beginning of each round, the procedure first pairs the alternatives that are still in contention. Then, for every pair of alternatives, the procedure makes a comparison between the alternatives by taking an equal number of observations for each alternative. The alternative with the larger sample mean advances to the next round of comparisons, and the other one is eliminated. By doing so, in each round, about half of the alternatives are eliminated. The procedure continues until there is only one alternative left. We show that by carefully determining the amount of sampling budget allocated to each round of selection, the procedure can achieve an RMCS in the order of k, which is optimal. To understand why we have such a result, we want to notice that, in our procedure, during the selection process, the best alternative does not need to compete with all other alternatives. Instead, it only needs to beat its opponent in each round, which is significantly smaller than k. Furthermore, because all comparisons are conducted locally between two alternatives, we show that our procedure can be easily implemented in parallel computing environments with almost no non-parallelizable calculations.

The remainder of the paper is organized as follows. We provide the problem formulation and conduct the lower bound analysis for the RMCS of existing fixed-budget procedures in Section 2. We propose a fixed-budget procedure and derive its RMCS in Section 3. The implementation details of the procedure in parallel computing environments are discussed in Section 4. Numerical results are presented in Section 5, followed by concluding remarks in Section 6.

2. Lower Bound Analysis for the RMCS of Existing Procedures

In this section, we first state the formal model for the fixed-budget ranking and selection (R&S) problem considered in this paper. Then, we review existing fixed-budget procedures. Finally, we establish a tight lower bound for the RMCS of a broad class of existing fixed-budget procedures.

2.1. Problem Formulation

Suppose that there are in total k alternatives in contention at the beginning of the selection process, and we use $\mathcal{K} = \{1, 2, \ldots, k\}$ to index all the alternatives. For each alternative $i \in \mathcal{K}$, it can generate independent and identically distributed (i.i.d.) observations $\{X_{i,\ell} : \ell = 1, 2, \ldots\}$ from a normal distribution with unknown mean μ_i and variance σ_i^2. Without loss of generality, we assume that the means of the alternatives are in ascending order, and alternative k is the best alternative, i.e., $\mu_k \geq \mu_{k-1} \geq \mu_{k-2} \cdots \geq \mu_1$. We further define $\bar{X}_i(n_i) = \sum_{\ell=1}^{n_i} X_{i,\ell}/n_i$ as the sample mean of the first n_i observations from alternative i. Throughout this paper, we consider the problem of finding the best alternative, i.e., alternative k, by taking observations from different alternatives. While developing statistical procedures to solve the problem, we adopt the fixed-budget formulation. The number of observations that a procedure can use to identify the best alternative is fixed and pre-specified by the user. We let N denote the pre-specified total sampling budget. Following the convention in the R&S literature, we call the probability that a procedure can correctly select alternative k when the total sampling budget is exhausted the *probability of correct selection* (PCS).

In this paper, to measure the performance of a procedure in solving large-scale problems, we quantify the minimal growth rate of N such that as k increases, the PCS does not decrease to zero. We call such a growth rate of N the *rate for maintaining correct selection* (RMCS). In what follows, we give a mathematical definition of the RMCS.

Definition 2.1 *Let $g(k)$ be a function of k, where there exists a positive constant $Q>0$ such that $g(k)>0$ for all $k>Q$. We say that the RMCS of a fixed-budget procedure is in the order of $g(k)$, if there exists a positive constant $M>0$, and the PCS of the procedure satisfies that,*

$$\lim_{k\to\infty} PCS=0 \text{ for } \lim_{k\to\infty} N/g(k)=0, \text{ and } \liminf_{k\to\infty} PCS>0 \text{ for } \liminf_{k\to\infty} N/g(k)\geq M.$$

In the R&S literature, fixed-precision procedures are closely related to fixed-budget procedures. Different from fixed-budget procedures, fixed-precision procedures are often designed to conduct reliable selections. Typically, for a fixed-precision procedure, there are no strict restrictions on the total sampling budget. However, it requires the user to specify a PCS $1-\alpha$ at the beginning of the selection process, where $\alpha>0$ is called the *probability of incorrect selection* (PICS), and the procedure needs to use as few observations as possible to deliver a PCS of at least $1-\alpha$. Recently, Zhong and Hong (2021) show that for a general fixed-precision procedure, as the number of alternatives k increases, to deliver a PCS of at least $1-\alpha$, a tight lower bound for the growth rate of expected total sample size is in the order of k. Even though this result is developed for a general fixed-precision procedure, with a few modifications to the analysis, one can show that such a lower bound, i.e., the order of k, applies to the RMCS of a general fixed-budget procedure as well. Therefore, it is theoretically important to ask whether the RMCS of existing fixed-budget procedures reaches the lower bound, and if not, how to design such a fixed-budget procedure.

2.2. Existing Fixed-Budget Procedures

Various fixed-budget procedures have been developed so far. As a common feature of the procedures, during the selection process, no alternatives are eliminated, and when the selection stops, the alternative with the largest sample mean is selected. In that case, the PCS equals the probability that alternative k has the largest sample mean among all k alternatives. Therefore, to determine the amount of sampling budget allocated to each alternative, the OCBA type of procedures try to solve a static optimization problem,

$$\begin{aligned} \max_{n_1,n_2,\ldots,n_k} \; & \mathbb{P}\left(\bar{X}_k\left(n_k\right) \geq \max_{i\in\mathcal{K}'} \bar{X}_i\left(n_i\right)\right), \text{ where } \mathcal{K}' = \mathcal{K}\setminus\{k\}, \\ \text{s.t.} \quad & n_1 + n_2 + \cdots + n_k = N; \\ & n_i \geq 1, \quad \forall i \in \mathcal{K}. \end{aligned} \tag{1}$$

Notice that a closed-form expression for the probability $\mathbb{P}\left(\bar{X}_k\left(n_k\right) \geq \max_{i\in\mathcal{K}'} \bar{X}_i\left(n_i\right)\right)$ is typically impossible to obtain. To find the solution, these procedures often maximize a lower bound of the probability, i.e., $1 - \sum_{i\in\mathcal{K}'} \mathbb{P}\left(\bar{X}_k\left(n_k\right) < \bar{X}_i\left(n_i\right)\right)$, which is calculated by using the Bonferroni inequality. Chen et al. (2000) and Glynn and Juneja (2004) show that by additionally assuming that $N \to \infty$, the solution to Problem (1) can be expressed as sampling ratio n_i/n_j between any two alternatives i and j. Notice that this solution depends on the true means and variances of the alternatives. Therefore, in a practical implementation, sample means and sample variances of the alternatives are often plugged in to estimate the solution. During the selection process, these procedures often need to frequently update the sample information of the alternatives so as to adjust their estimates of the solution to Problem (1). The procedures stop when the total sampling budget is exhausted. Clearly, the exact solution to Problem (1) provides the performance upper bound for this type of procedures.

Instead of trying to maximize the PCS and solve Problem (1) directly, there are some fixed-budget procedures that minimize the difference between the means of the selected alternative and the best alternative, i.e., the expected improvement (EI) type of procedures. This type of procedures are often developed under the Bayesian framework and treat the allocation of observations as a dynamic programming problem. These procedures often allocate observations one at a time. The mean of each alternative is represented by a random variable. The random variable has a predictive distribution that characterizes one's belief about the unknown mean of each alternative and changes over time as more information is obtained from observations. Upon allocating an observation, these procedures need to solve a one-step lookahead optimization problem. Specifically, these procedures use the predictive distribution to make a probabilistic forecast on the next outcome from each alternative and quantify the expected improvement on the mean of the selected alternative. Then, the observation can be allocated to the alternative with the largest expected improvement. Theoretically analyzing the performance of this type of procedures is often very difficult due to the nonlinearity and noconvexity of the functions used in the expected improvement calculations. However, Ryzhov (2016) recently shows that if $N \to \infty$, some EI type of procedures are equivalent to the classical OCBA procedure. It suggests that even though the key idea used to develop the EI type of procedures differs from that of the OCBA type of procedures, many EI type of procedures may still share the same optimal sample allocation as that of the OCBA type of procedures. Therefore, the exact solution to Problem (1) may still provide the performance upper bound for many EI type of procedures.

2.3. Lower Bound Analysis

Because the exact solution to Problem (1) provides the performance upper bound for a broad class of existing fixed-budget procedures, in this subsection, we show that if the exact solution to Problem (1) is used to allocate the sampling budget, the RMCS of such a sample allocation is lower bounded by the order of $k\log k$.

We first make an assumption on the R&S problem. It basically requires that the problem is not too easy to solve.

ASSUMPTION 1. *The observations generated by different alternatives are independent, the variances of all alternatives are lower bounded by a positive constant $\underline{\sigma}^2 > 0$, i.e., $\sigma_i^2 \geq \underline{\sigma}^2 > 0$ for all $i \in \mathcal{K}$, and the mean differences between the best alternative and any other alternatives are upper bounded by a positive constant $\delta_{upper} > 0$, i.e., $\mu_k - \mu_1 \leq \delta_{upper}$.*

We also need the following lemma, which can help us quantify the asymptotic behavior of the tail probability of a standard normal random variable.

LEMMA 1 **(Cramér (1938))**. *Let $\{Z_\ell : \ell = 1, 2, \ldots,\}$ be a sequence of i.i.d. standard normal random variables, and let $\mathcal{M}_\nu = \sum_{\ell=1}^{\nu} Z_\ell/\nu$. Then,*

$$(i) \quad \lim_{\nu\to\infty} \frac{1}{\nu} \log \mathbb{P}(\mathcal{M}_\nu \geq t) = -t^2/2 \quad (t > \mathbb{E}[Z_1]),$$
$$(ii) \quad \lim_{\nu\to\infty} \frac{1}{\nu} \log \mathbb{P}(\mathcal{M}_\nu \leq t) = -t^2/2 \quad (t < \mathbb{E}[Z_1]).$$

With the conditions stated in Assumption 1 and the results demonstrated in Lemma 1, we prove the following theorem.

THEOREM 1. *Suppose that Assumption 1 holds. Let $n_1^*, n_2^*, \ldots, n_k^*$ be the exact solution to Problem (1). If $\lim_{k\to\infty} N/(k\log k) = 0$, then,*

$$\lim_{k\to\infty} \mathbb{P}\left(\bar{X}_k(n_k^*) \geq \max_{i\in\mathcal{K}'} \bar{X}_i(n_i^*)\right) = 0.$$

Proof. Here and throughout, we let Z denote a standard normal random variable. For any positive constant $\epsilon > 0$, we can find a positive constant $U > 0$ such that $\mathbb{P}(Z \leq U) \geq \sqrt{1-\epsilon}$. Let

$U' = (\sigma_k U + \delta_{upper}) / \underline{\sigma}$ and $c = 1/(4U'^2)$. We further define $\mathcal{K}''$ as a subset of $\mathcal{K}'$ such that $n_i^* \leq 2c\log k$ for all $i \in \mathcal{K}''$. Then, we have,

$$\begin{aligned}
\mathbb{P}\left(\bar{X}_k(n_k^*) < \max_{i\in\mathcal{K}'} \bar{X}_i(n_i^*)\right) &\geq \mathbb{P}\left(\bar{X}_k(n_k^*) \leq \sigma_k U + \mu_k\right) \times \mathbb{P}\left(\max_{i\in\mathcal{K}'} \bar{X}_i(n_i^*) > \sigma_k U + \mu_k\right) \\
&\geq \mathbb{P}\left(\frac{\bar{X}_k(n_k^*) - \mu_k}{\frac{\sigma_k}{\sqrt{n_k^*}}} \leq \sqrt{n_k^*}U\right) \times \mathbb{P}\left(\max_{i\in\mathcal{K}''} \bar{X}_i(n_i^*) > \sigma_k U + \mu_k\right) \\
&\geq \mathbb{P}(Z \leq U) \times \left[1 - \mathbb{P}\left(\max_{i\in\mathcal{K}''} \bar{X}_i(n_i^*) \leq \sigma_k U + \mu_k\right)\right] \\
&\geq \sqrt{1-\epsilon} \times \left[1 - \mathbb{P}\left(\max_{i\in\mathcal{K}''} \bar{X}_i(n_i^*) \leq \sigma_k U + \mu_k\right)\right],
\end{aligned} \tag{2}$$

where the second inequality holds because $\max_{i\in\mathcal{K}'} \bar{X}_i(n_i^*) \geq \max_{i\in\mathcal{K}''} \bar{X}_i(n_i^*)$, and the third inequality holds because $n_k^* \geq 1$. Next, we focus on analyzing the second term in Equation (2). We have,

$$\begin{aligned}
1 - \mathbb{P}\left(\max_{i\in\mathcal{K}''} \bar{X}_i(n_i^*) \leq \sigma_k U + \mu_k\right) &= 1 - \prod_{i\in\mathcal{K}''} \mathbb{P}\left(\bar{X}_i(n_i^*) \leq \sigma_k U + \mu_k\right) \\
&= 1 - \prod_{i\in\mathcal{K}''} \left[1 - \mathbb{P}\left(\bar{X}_i(n_i^*) > \sigma_k U + \mu_k\right)\right] \\
&= 1 - \prod_{i\in\mathcal{K}''} \left[1 - \mathbb{P}\left(\frac{\bar{X}_i(n_i^*) - \mu_i}{\frac{\sigma_i}{\sqrt{n_i^*}}} > \sqrt{n_i^*}\frac{\sigma_k U + \mu_k - \mu_i}{\sigma_i}\right)\right] \\
&\geq 1 - \prod_{i\in\mathcal{K}''} \left[1 - \mathbb{P}\left(Z > \sqrt{2c\log k}\frac{\sigma_k U + \delta_{upper}}{\underline{\sigma}}\right)\right] \\
&= 1 - \prod_{i\in\mathcal{K}''} \left(1 - \mathbb{P}\left(\frac{Z}{\sqrt{\log k}} > \frac{1}{\sqrt{2}}\right)\right),
\end{aligned} \tag{3}$$

where the inequality holds because for $i \in \mathcal{K}''$, $n_i^* \leq 2c\log k$, $\sigma_i^2 \geq \underline{\sigma}^2$, and $\mu_k - \mu_i \leq \delta_{upper}$. Since $\lim_{k\to\infty} N/(k\log k) = 0$, there exists a positive constant $K_1 > 0$ such that $N < c(k+2)\log k$ for all $k \geq K_1$. It suggests that if $k \geq K_1$, there are at least $k/2+1$ alternatives in set $\mathcal{K}$ having fewer than $2c\log k$ observations. Therefore, if $k \geq K_1$, $|\mathcal{K}''| \geq k/2$, where $|\mathcal{K}''|$ is the cardinality of set $\mathcal{K}''$. Then, by letting $k \geq K_1$, we can rewrite Equation (3) as,

$$(3) = 1 - \left(1 - \mathbb{P}\left(\frac{Z}{\sqrt{\log k}} > \frac{1}{\sqrt{2}}\right)\right)^{|\mathcal{K}''|} \geq 1 - \left(1 - \mathbb{P}\left(\frac{Z}{\sqrt{\log k}} > \frac{1}{\sqrt{2}}\right)\right)^{\frac{k}{2}}. \tag{4}$$

By Lemma 1, it can be checked that,

$$\lim_{\log k\to\infty} \frac{1}{\log k}\log\mathbb{P}\left(\frac{Z}{\sqrt{\log k}} > \frac{1}{\sqrt{2}}\right) = -\frac{1}{4}.$$

It suggests that there exists another positive constant $K_2 > 0$, such that,

$$\frac{1}{\log k}\log\mathbb{P}\left(\frac{Z}{\sqrt{\log k}} > \frac{1}{\sqrt{2}}\right) \geq -\frac{1}{2},$$

for all $k \geq K_2$. Rearranging the terms in the equation listed above, we have $\mathbb{P}\left(Z/\sqrt{\log k} > 1/\sqrt{2}\right) \geq 1/\sqrt{k}$. Therefore, by letting $k \geq K_2$, we can bound Equation (4) as follows,

$$(4) \geq 1 - \left(1 - \frac{1}{\sqrt{k}}\right)^{\frac{k}{2}}. \tag{5}$$

For the second term in Equation (5), it can be easily checked that,

$$\lim_{k\to\infty}\left(1 - \frac{1}{\sqrt{k}}\right)^{\frac{k}{2}} = \lim_{k\to\infty}\exp\left(\frac{k}{2}\log\frac{\sqrt{k}-1}{\sqrt{k}}\right) = \exp\left(\lim_{k\to\infty}\frac{\log\frac{\sqrt{k}-1}{\sqrt{k}}}{\frac{2}{k}}\right) = \exp\left(\lim_{k\to\infty}\frac{\frac{1}{2k\left(\sqrt{k}-1\right)}}{-\frac{2}{k^2}}\right) = 0,$$

where the third equality holds due to L'Hôpital's Rule. It implies that there exists a positive constant $K_3 > 0$ such that, $\left(1 - 1/\sqrt{k}\right)^{\frac{k}{2}} < 1 - \sqrt{1-\epsilon}$, for all $k \geq K_3$. Then, by letting $k \geq K_3$, Equation (5) can be written as,

$$(5) \geq 1 - 1 + \sqrt{1-\epsilon} = \sqrt{1-\epsilon}. \tag{6}$$

Combining the results in Equations (2)-(6), we can deduce that, for any positive constant $\epsilon > 0$, by letting $k \geq \max\{K_1, K_2, K_3\}$, we have,

$$\begin{aligned}
\mathbb{P}\left(\bar{X}_k\left(n_k^*\right) < \max_{i\in\mathcal{K}'}\bar{X}_i\left(n_i^*\right)\right) &\geq \sqrt{1-\epsilon} \times \left[1 - \mathbb{P}\left(\max_{i\in\mathcal{K}''}\bar{X}_i\left(n_i^*\right) \leq \sigma_k U + \mu_k\right)\right] \\
&\geq \sqrt{1-\epsilon}\left[1 - \prod_{i\in\mathcal{K}''}\left(1 - \mathbb{P}\left(\frac{Z}{\log k} > \frac{1}{\sqrt{2}}\right)\right)\right] \\
&\geq \sqrt{1-\epsilon}\left[1 - \left(1 - \mathbb{P}\left(\frac{Z}{\log k} > \frac{1}{\sqrt{2}}\right)\right)^{\frac{k}{2}}\right] \\
&\geq \sqrt{1-\epsilon}\left[1 - \left(1 - \frac{1}{\sqrt{k}}\right)^{\frac{k}{2}}\right] \\
&\geq 1 - \epsilon.
\end{aligned}$$

Because $\mathbb{P}\left(\bar{X}_k\left(n_k^*\right) < \max_{i\in\mathcal{K}'}\bar{X}_i\left(n_i^*\right)\right) \leq 1$, by the definition of the limit, we have $\lim_{k\to\infty}\mathbb{P}\left(\bar{X}_k\left(n_k^*\right) < \max_{i\in\mathcal{K}'}\bar{X}_i\left(n_i^*\right)\right) = 1$. Therefore,

$$\lim_{k\to\infty}\mathbb{P}\left(\bar{X}_k\left(n_k^*\right) \geq \max_{i\in\mathcal{K}'}\bar{X}_i\left(n_i^*\right)\right) = 0.$$

It concludes the proof. □

Theorem 1 shows that if the exact solution to Problem (1) is used to allocate the sampling budget, as the number of alternatives increases, in order to keep the PCS from decreasing to zero, the total sampling budget should grow at least in the order of $k\log k$. It suggests that in terms of the RMCS, there is still room for improvement for existing fixed-budget procedures. Therefore, it is of theoretical interest to investigate whether we can develop a fixed-budget procedure that achieves an RMCS in the order of k.

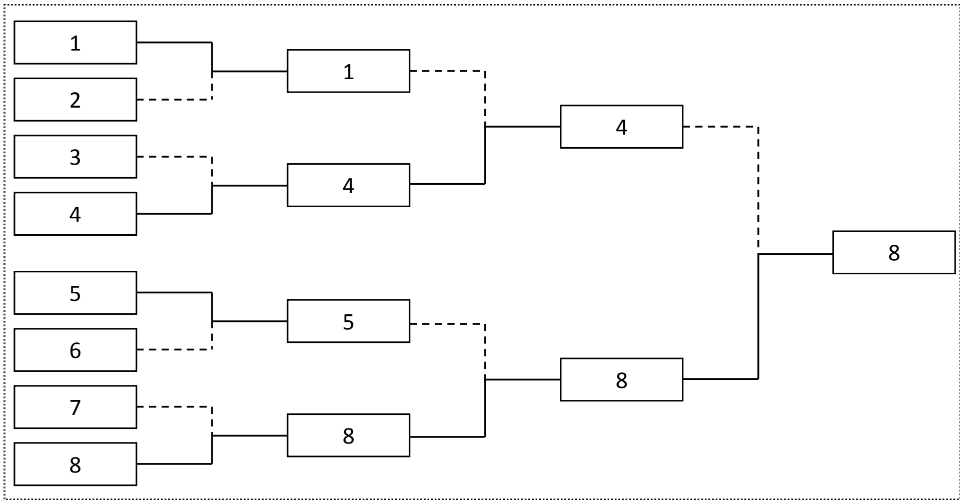

Figure 1 An illustration of using the $\mathcal{KT}$ procedure to solve a problem with 8 alternatives.

3. The Fixed-Budget Knockout-Tournament Procedure

In this section, we first introduce the knockout-tournament ($\mathcal{KT}$) procedure proposed by Zhong and Hong (2021) to solve large-scale fixed-precision R&S problems in parallel computing environments, because it is the building block of our procedure. We then develop a new fixed-budget procedure and show that the RMCS of the procedure is in the order of k. Lastly, some modifications have been made to the procedure to further improve its selection accuracy. Different from many existing fixed-budget procedures, while developing our procedure, we allow the use of common random numbers (CRNs) to improve the selection accuracy of the procedure.

3.1. The Fixed-Precision Knockout-Tournament Procedure

The knockout-tournament ($\mathcal{KT}$) procedure is a fixed-precision procedure developed based on the knockout-tournament arrangement of tennis Grand Slam tournaments. It conducts the selection round-by-round. At the beginning of each round, the procedure first pairs the alternatives that are still in contention. Then, for every pair of alternatives, the procedure conducts a "match" between the two by using an existing fixed-precision R&S procedure. The "winner" advances to the next round of comparisons, and the "loser" is eliminated. The procedure eliminates about half of the alternatives in each round and can select the best in $\lceil \log_2 k \rceil$ rounds where $\lceil \cdot \rceil$ is the ceiling function. The authors show that by properly designing the allocation of the PICS α to each round, and ensuring that falsely eliminating alternative k happens in each round with the probability less than the PICS as allocated, the procedure can deliver a PCS of at least $1-\alpha$. In Figure 1, we give an example of the selection of eight alternatives by the $\mathcal{KT}$ procedure.

One may observe that, as a distinct feature of the $\mathcal{KT}$ procedure, in order to be selected, alternative k does not need to beat all other $k-1$ alternatives. Instead, it only needs to win the "match" in each round and competes with at most $\lceil \log_2 k \rceil$ alternatives. Due to this feature, the authors prove that as the number of alternatives k increases, it is possible for the $\mathcal{KT}$ procedure to deliver a PCS of at least $1-\alpha$ with the expected total sample size growing linearly in k. This growth rate is lower than those of other fixed-precision R&S procedures in the literature, and it achieves the optimal. Therefore, while solving large-scale fixed-precision R&S problems, the $\mathcal{KT}$ procedure tends to use fewer observations than other fixed-precision R&S procedures do. The $\mathcal{KT}$ procedure provides a nice framework for designing procedures to solve large-scale fixed-precision R&S problems. It motivates us to consider whether we can alter the procedure to solve fixed-budget R&S problems and achieve an RMCS in the order of k.

3.2. The Procedure

To alter the $\mathcal{KT}$ procedure to solve fixed-budget R&S problems, one major difficulty is to ensure that the procedure can select the best alternative before the total sampling budget is exhausted. To overcome this issue, we consider two modifications to the procedure. First, when the selection starts, instead of allocating the PICS, we determine the sampling budget N_r that should be allocated to each round r and ensure that $\sum_{r=1}^{\lceil \log_2 k \rceil} N_r \le N$. Second, if there are $2p$ alternatives left at the beginning of round r, i.e., p "matches" are needed, we equally allocate the sampling budget N_r to these "matches". Then, for each "match", we can use a maximum of N_r/p observations to compare the alternatives. In this paper, we choose to directly take $N_r/2p$ observations for each of the two alternatives and advance the alternative with the larger sample mean to the next round. By doing so, we ensure that there are always fewer than N observations being used when the procedure stops. We call our procedure the $\mathcal{FBKT}$ procedure where $\mathcal{FBKT}$ stands for fixed-budget knockout-tournament and list the detailed description of the procedure as follows.

Procedure $\mathcal{FBKT}$

Step 1. (Initialization): *Set the total sampling budget N. Determine the sampling budget N_r that should be allocated to round r and ensure that $\sum_{r=1}^{\lceil \log_2 k \rceil} N_r \le N$. Let $\mathcal{I}_r$ denote the set of alternatives that are still in contention at the beginning of round r. Set $r=1$ and $\mathcal{I}_1 = \{1,2,\ldots,k\}$.*

Step 2. (Pairing and Screening): *Let $\mathcal{N}_r = \lfloor N_r/|\mathcal{I}_r| \rfloor$, $\mathcal{I}_{r+1} = \emptyset$, and $\mathcal{I}'_r = \mathcal{I}_r$, where $\lfloor \cdot \rfloor$ is the floor function.*

For $v=1$ *to* $\lfloor |\mathcal{I}_r|/2 \rfloor$

- *Randomly pick two alternatives i and j from $\mathcal{I}'_r$, take $\mathcal{N}_r$ observations for each of the two alternatives, and update $\mathcal{I}'_r = \mathcal{I}'_r \setminus \{i,j\}$.*
- *Let $\bar{X}_i^r$ and $\bar{X}_j^r$ denote the sample means of alternatives i and j calculated based on these $\mathcal{N}_r$ observations respectively. If $\bar{X}_i^r \geq \bar{X}_j^r$, update $\mathcal{I}_{r+1} = \mathcal{I}_{r+1} \bigcup \{i\}$. If $\bar{X}_i^r < \bar{X}_j^r$, update $\mathcal{I}_{r+1} = \mathcal{I}_{r+1} \bigcup \{j\}$.*

Endfor

If there is one alternative left in $\mathcal{I}'_r$, directly advance the leftover alternative to the next round of comparisons, i.e., include the index of the leftover alternative in $\mathcal{I}_{r+1}$.

Step 3. (Stopping Rule): *Update $r = r+1$. If $|\mathcal{I}_r| = 1$, let φ be the index of the alternative in $\mathcal{I}_r$ and select alternative φ as the best. Otherwise, go to **Step 2**.*

REMARK 1. In practice, if $\mathcal{N}_r = 0$, then one can stop the selection process and randomly choose one alternative in $\mathcal{I}_r$ as the best.

The $\mathcal{FBKT}$ procedure provides a simple way of developing fixed-budget procedures by using the selection structure of the $\mathcal{KT}$ procedure. In order to provide statistical guarantee upon stopping, while comparing two alternatives in round r, the $\mathcal{FBKT}$ procedure abandons all previous sample information. It can cause a waste of the sampling budget. However, the procedure conducts $k-1$ "matches" before it selects the best alternative. On average, each alternative only participates in two "matches". The waste is tolerable.

Compared to existing fixed-budget procedures, one advantage of the $\mathcal{FBKT}$ procedure is that it allows the practical use of CRNs. In the R&S literature, the use of CRNs is for the purpose of introducing positive correlations to the sample means of the alternatives that are in comparison. It requires one to use one identical (pseudo) random number stream to generate observations for different alternatives. Then, the observations generated by different alternatives are positively correlated, and it becomes easier to compare alternatives. Typically, CRNs cannot be directly applied to existing fixed-budget procedures. One major reason is that, for existing fixed-budget procedures, when alternatives are compared, the alternatives often have different numbers of observations. The correlations between the sample means of different alternatives decrease as the difference in sample size gets large. The use of CRNs can fail in this situation. Meanwhile, for the $\mathcal{FBKT}$ procedure, whenever a comparison is made, it only involves two alternatives, and the two alternatives always have the same number of observations. CRNs can be easily adopted in the procedure.

Notice that in the multi-armed bandit (MAB) field, there is a stream of research works known as the best arm identification. It is quite similar to the context of R&S problem, except that it often considers the situation where the observations are drawn from sub-gaussian distributions with known variance proxies, e.g., distributions with bounded supports. Similar to the $\mathcal{KT}$ procedure, the median elimination procedure, which is proposed by Even-Dar et al. (2002) to solve the best arm identification problem, can also eliminate about half of the alternatives in each round by eliminating the alternatives whose sample means are below the median. However, the procedure requires the observations generated by different alternatives be independent and does not support the use of CRNs. Therefore, in this paper, we choose to develop our procedure by modifying the $\mathcal{KT}$ procedure.

In the $\mathcal{FBKT}$ procedure, we equally allocate the sampling budget N_r to different "matches" in round r. This allocation scheme is not unique. In practice, one has the option to use other allocation schemes. For example, at the beginning of each round, one may take some initial observations for each alternative to estimate the sample variance of the difference between the two alternatives in every "match" and then allocate more sampling budget to the "matches" with larger sample variances. However, from a theoretical point of view, without knowing the configuration of the means and variances of the alternatives, it is unclear whether this allocation scheme would improve the performance of the procedure or not, because it increases the PCS of the procedure for the case where the "match" containing alternative k has a relatively large sample variance but lowers the PCS of the procedure when the sample variance of the "match" containing alternative k is relatively small. Therefore, in this paper, we simply choose to allocate the same amount of sampling budget to every "match" in each round.

3.3. Analysis on the RMCS

Because in the $\mathcal{FBKT}$ procedure, the best alternative only needs to win the "match" in each round, the PCS of the $\mathcal{FBKT}$ procedure is no longer in the same form as that in the existing fixed-budget procedures. Let $\mathcal{Q}_r$ denote the event that alternative k eliminates its opponent, namely, alternative k_r, in round r, i.e., $\mathcal{Q}_r = \{\bar{X}_k^r \geq \bar{X}_{k_r}^r\}$. Then, the PCS of the $\mathcal{FBKT}$ procedure is in the form of,

$$\mathrm{PCS} = \mathbb{P}\left(\bigcap_{r=1}^{\lceil \log_2 k \rceil} \mathcal{Q}_r\right). \tag{7}$$

Next, we show that due to such a difference, the $\mathcal{FBKT}$ procedure can attain an RMCS in the order of k. To proceed the analysis, we first define σ_{ij}^2 as the variance of the difference between any two alternatives i and j, i.e., $\sigma_{ij}^2 = \mathrm{Var}\left(X_{i,\ell} - X_{j,\ell}\right)$, where $i, j \in \mathcal{K}$ and $i \neq j$. Then, we make the following assumptions on the problem the procedure aims to solve. They require that the problem is not too difficult to solve.

Assumption 2. *There exists a constant $\delta > 0$ such that $\mu_k - \mu_i \geq \delta$ for all $i \in \mathcal{K}'$.*

Assumption 3. *The variance of the difference between any two alternatives is upper bounded by a constant $\sigma_{upper}^2 > 0$, i.e., $\sigma_{ij}^2 \leq \sigma_{upper}^2$ for all $i, j \in \mathcal{K}$ and $i \neq j$.*

In the R&S literature, parameter δ stated in Assumption 2 is called the indifference-zone (IZ) parameter. It ensures that the mean difference between the best and the second-best alternatives is lower bounded, and the best alternative is identifiable with a finite number of observations. In the $\mathcal{FBKT}$ procedure, we only need the existence of such a parameter and do not need to set it while running the procedure. We also prove the following lemmas.

LEMMA 2. *If* $F = \prod_{r=1}^{\infty}(1-\exp(-\varrho r))$ *for some constant* $\varrho > 0$*, then,*

$$F \geq \exp\left(-\frac{\pi^2}{6\varrho}\right).$$

LEMMA 3. *For a standard normal random variable* Z *and any positive constant* $d > 0$*, we have,*

$$\mathbb{P}(Z \geq d) \leq \exp\left(-\frac{d^2}{2}\right).$$

Here and throughout, we let ϕ be a positive integer. With the conditions stated in Assumptions 2 and 3 and the results demonstrated in Lemmas 2 and 3, we prove the following theorem.

THEOREM 2. *Suppose that Assumptions 2 and 3 hold, and* $\phi \geq 2$*. If* $N_r = \left\lfloor \frac{r}{\phi(\phi-1)} \left(\frac{\phi-1}{\phi}\right)^r N \right\rfloor$ *for* $r \geq 1$*, the* $\mathcal{FBKT}$ *procedure can stop before* N *observations are used. Furthermore, for any positive constant* $\eta_1 \geq 2\phi^2$*, if* $N/k \geq \eta_1$*, the PCS of the* $\mathcal{FBKT}$ *procedure satisfies that,*

$$PCS \geq \exp\left(-\frac{4\phi^2\pi^2\sigma_{upper}^2}{3\eta_1\delta^2}\right),$$

for all $k \geq 2$*.*

Proof. It can be verified that,

$$\begin{aligned}
\sum_{r=1}^{\lceil \log_2 k \rceil} \left\lfloor \frac{r}{\phi(\phi-1)} \left(\frac{\phi-1}{\phi}\right)^r N \right\rfloor &\leq \sum_{r=1}^{\infty} \frac{r}{\phi(\phi-1)} \left(\frac{\phi-1}{\phi}\right)^r N \\
&= \phi\left[\sum_{r=1}^{\infty} \frac{r}{\phi(\phi-1)} \left(\frac{\phi-1}{\phi}\right)^r N - \sum_{r=1}^{\infty} \frac{r}{\phi(\phi-1)} \left(\frac{\phi-1}{\phi}\right)^{r+1} N\right] \\
&= \phi\left[\frac{N}{\phi(\phi-1)} \sum_{r=1}^{\infty} \left(\frac{\phi-1}{\phi}\right)^r\right] \\
&= \frac{N}{\phi-1} \times (\phi-1) \\
&= N.
\end{aligned}$$

It suggests that by letting $N_r = \left\lfloor \frac{r}{\phi(\phi-1)} \left(\frac{\phi-1}{\phi}\right)^r N \right\rfloor$ for $r \geq 1$, the $\mathcal{FBKT}$ procedure can stop before N observations are used.

Next, we show that by using the sample allocation rule listed above, for any positive constant $\eta_1 \geq 2\phi^2$, if $N/k \geq \eta_1$, the PCS of the $\mathcal{FBKT}$ procedure is lower bounded for all $k \geq 2$. To prove this, we first let $\hat{k} = 2^{\lceil \log_2 k \rceil}$. By the definition of $\hat{k}$, we can conclude that $k \le \hat{k} < 2k$. Then, by letting $\mathcal{Q}_r^c$ be the complement of event $\mathcal{Q}_r$, based on Equation (7), we can write the PCS of the $\mathcal{FBKT}$ procedure as follows,

$$\text{PCS} = \mathbb{P}\left(\bigcap_{r=1}^{\log_2 \hat{k}} \{\mathcal{Q}_r\}\right) = \mathbb{P}(\mathcal{Q}_1) \prod_{r=2}^{\log_2 \hat{k}} \mathbb{P}(\mathcal{Q}_r | \mathcal{Q}_1, \dots, \mathcal{Q}_{r-1}) = (1 - \mathbb{P}(\mathcal{Q}_1^c)) \prod_{r=2}^{\log_2 \hat{k}} (1 - \mathbb{P}(\mathcal{Q}_r^c | \mathcal{Q}_1, \dots, \mathcal{Q}_{r-1})). \tag{8}$$

For the sampling budget $\mathcal{N}_r = \lfloor N_r / |\mathcal{I}_r| \rfloor$ that should be allocated to each alternative in round r, we have,

$$\mathcal{N}_r = \left\lfloor \frac{r}{\phi(\phi-1)} \left(\frac{\phi-1}{\phi}\right)^r \frac{N}{|\mathcal{I}_r|} \right\rfloor \ge \left\lfloor \frac{r}{\phi(\phi-1)} \left(\frac{\phi-1}{\phi}\right)^r \frac{\eta_1 k}{\frac{\hat{k}}{2^{r-1}}} \right\rfloor \ge r \left\lfloor \frac{\eta_1}{4\phi(\phi-1)} \left(\frac{2(\phi-1)}{\phi}\right)^r \right\rfloor, \tag{9}$$

where the first inequality holds because $N \ge \eta_1 k$ and for the $\mathcal{FBKT}$ procedure, at the beginning of round $r \ge 1$, there are fewer than $\hat{k}/2^{r-1}$ alternatives that are still in contention, i.e., $|\mathcal{I}_r| \le \hat{k}/2^{r-1}$. The second inequality holds because $\hat{k} < 2k$. Notice that $\eta_1 \ge 2\phi^2$ ensures that $\mathcal{N}_r$ is a positive integer no less than r. Then, based on the results listed in Equation (9), we can write Equation (8) as follows,

$$\begin{aligned}
(8) &= \left(1 - \mathbb{P}\left(\bar{X}_{k_1}^1 > \bar{X}_k^1\right)\right) \prod_{r=2}^{\log_2 \hat{k}} \left(1 - \mathbb{P}\left(\bar{X}_{k_r}^r > \bar{X}_k^r \middle| \mathcal{Q}_1, \dots, \mathcal{Q}_{r-1}\right)\right) \\
&= \left(1 - \mathbb{P}\left(\frac{\bar{X}_{k_1}^1 - \bar{X}_k^1 - \mu_{k_1} + \mu_k}{\frac{\sigma_{k_1 k}}{\sqrt{\mathcal{N}_1}}} > \sqrt{\mathcal{N}_1}\frac{\mu_k - \mu_{k_1}}{\sigma_{k_1 k}}\right)\right) \times \\
&\quad \prod_{r=2}^{\log_2 \hat{k}} \left(1 - \mathbb{P}\left(\frac{\bar{X}_{k_r}^r - \bar{X}_k^r - \mu_{k_r} + \mu_k}{\frac{\sigma_{k_r k}}{\sqrt{\mathcal{N}_r}}} > \sqrt{\mathcal{N}_r}\frac{\mu_k - \mu_{k_r}}{\sigma_{k_r k}} \middle| \mathcal{Q}_1, \dots, \mathcal{Q}_{r-1}\right)\right) \\
&\ge \left(1 - \mathbb{P}\left(Z > \sqrt{\left\lfloor \frac{\eta_1}{2\phi^2} \right\rfloor} \frac{\delta}{\sigma_{upper}}\right)\right) \times \\
&\quad \prod_{r=2}^{\log_2 \hat{k}} \left(1 - \mathbb{P}\left(Z > \sqrt{r \left\lfloor \frac{\eta_1}{4\phi(\phi-1)} \left(\frac{2(\phi-1)}{\phi}\right)^r \right\rfloor} \frac{\delta}{\sigma_{upper}} \middle| \mathcal{Q}_1, \dots, \mathcal{Q}_{r-1}\right)\right) \\
&\ge \left(1 - \mathbb{P}\left(Z > \sqrt{\left\lfloor \frac{\eta_1}{2\phi^2} \right\rfloor} \frac{\delta}{\sigma_{upper}}\right)\right) \prod_{r=2}^{\log_2 \hat{k}} \left(1 - \mathbb{P}\left(Z > \sqrt{r \left\lfloor \frac{\eta_1}{2\phi^2} \right\rfloor} \frac{\delta}{\sigma_{upper}} \middle| \mathcal{Q}_1, \dots, \mathcal{Q}_{r-1}\right)\right),
\end{aligned} \tag{10}$$

where the first inequality holds due to Equation (9) and Assumptions 2 and 3. The second inequality holds due to the fact that $[2(\phi-1)/\phi]^r \geq 2(\phi-1)/\phi$ when $r \geq 2$ and $\phi \geq 2$. Because event $\left\{Z > \sqrt{r\lfloor \eta_1/(2\phi^2)\rfloor}\delta/\sigma_{upper}\right\}$ is independent of events $\{\mathcal{Q}_1,\ldots,\mathcal{Q}_{r-1}\}$, Equation (10) yields,

$$
\begin{aligned}
(10) &= \prod_{r=1}^{\log_2 \hat{k}} \left(1-\mathbb{P}\left(Z > \sqrt{r\left\lfloor \frac{\eta_1}{2\phi^2}\right\rfloor}\frac{\delta}{\sigma_{upper}}\right)\right) \\
&\geq \prod_{r=1}^{\infty}\left(1-\exp\left(-\frac{\lfloor \eta_1/(2\phi^2)\rfloor \delta^2}{2\sigma_{upper}^2} r\right)\right) \\
&\geq \exp\left(-\frac{\pi^2\sigma_{upper}^2}{3\lfloor \eta_1/(2\phi^2)\rfloor \delta^2}\right) \\
&\geq \exp\left(-\frac{4\phi^2\pi^2\sigma_{upper}^2}{3\eta_1\delta^2}\right), \qquad (11)
\end{aligned}
$$

where the first inequality holds due to Lemma 3, and the second inequality holds due to Lemma 2. Therefore, combining the results listed in Equations (8), (10), and (11), we can conclude that if $N/k \geq \eta_1$, the PCS of the $\mathcal{FBKT}$ procedure is lower bounded by $\exp\left(-4\phi^2\pi^2\sigma_{upper}^2/(3\eta_1\delta^2)\right)$. It concludes the proof. $\square$

Theorem 2 shows that by carefully determining the amount of sampling budget allocated to each round of selection, the $\mathcal{FBKT}$ procedure can achieve an RMCS in the order of k. It is lower than those of existing procedures. This result suggests that to solve large-scale problems, the $\mathcal{FBKT}$ procedure tends to deliver a higher PCS than existing procedures do. Notice that in Theorem 2, positive integer ϕ quantifies the decay rate of the amount of sampling budget allocated each round of selection as r increases as well as the lower bound that the procedure can provide for the PCS in the worst-case scenario. The smaller value for ϕ, the larger the decay rate, and the larger the lower bound the procedure can provide for the PCS. By setting $\phi = 2$, the procedure can provide the largest lower bound for the PCS.

We are aware that there is a potential issue with Theorem 2. As the number of alternatives k increases, the mean difference between the best and the second-best alternatives may shrink to 0, i.e., $\delta \to 0$. Then, the results stated in Theorem 2 may no longer hold. To overcome this issue, instead of letting the $\mathcal{FBKT}$ procedure select the best alternative, one may consider letting the procedure conduct the good selection. Specifically, for any IZ parameter $\delta > 0$, one may require the procedure to select an alternative within δ to the best. We call the probability that the procedure can select an alternative within δ to the best the *probability of good selection* (PGS), i.e., $PGS = \mathbb{P}(\mu_k - \mu_\varphi \leq \delta)$, where φ is the index of the alternative selected by the procedure. In the proposition below, we prove that by setting $\phi \geq 3$, the $\mathcal{FBKT}$ procedure can keep the PGS from decreasing to 0 as the total sampling budget N grows linearly in k. Its proof is included in the appendix.

Proposition 1. *Suppose that Assumption 3 holds and $\phi \geq 3$. Then, for any positive constant $\eta_1' \geq 2\phi^2$ and IZ parameter $\delta > 0$, if $N_r = \left\lfloor \frac{r}{\phi(\phi-1)} \left(\frac{\phi-1}{\phi}\right)^r N \right\rfloor$ for $r \geq 1$ and $N/k \geq \eta_1'$, the PGS of the $\mathcal{FBKT}$ procedure satisfies that,*

$$PGS \geq \exp\left(-\frac{8\pi^2\sigma_{upper}^2\phi^3(\phi-1)}{3\eta_1'\delta^2\left(\sqrt{2\phi(\phi-1)}-\phi\right)^2}\right),$$

for all $k \geq 2$.

Notice that by combining the results stated in Theorem 2 and Proposition 1, we can conclude that setting $\phi = 3$ can keep both PCS and PGS from decreasing to zero as N grows linearly in k. At the same time, it provides relatively large lower bounds for the PCS and the PGS. Therefore, in practice, for the $\mathcal{FBKT}$ procedure, while allocating the sampling budget to each round of selection, we recommend setting $\phi = 3$.

3.4. Seeding to Improve the Selection Accuracy

The $\mathcal{FBKT}$ procedure randomly pairs the alternatives in each round. Under this pairing scheme, the opponent of an alternative cannot be controlled. As a result, during the selection process, the best alternative may frequently participate in competitive "matches" and face alternatives whose means are very close to it. It can significantly increase the probability that the best alternative is falsely eliminated and reduce the selection accuracy of the procedure. To resolve this issue, while pairing the alternatives in each round, we can borrow the idea of seeding commonly used in many sports tournaments. The purpose of seeding is to separate the top players in a draw so that they will not meet in the early rounds of a tournament. To achieve that, players will be seeded (ranked) at the beginning of each round based on their previous records. Then, the strongest player is made to play the weakest player, the second strongest player plays the second weakest participant, etc. By doing so, top players have larger chances of surviving to the end of the tournament. In what follows, we list the detailed description of the procedure where such a seeding method is used to pair the alternatives in each round and call it the $\mathcal{FBKT}^{\mathcal{S}}$ procedure.

Procedure $\mathcal{FBKT}^{\mathcal{S}}$

Step 1. (Initialization): *Set the total sampling budget $N_0 + N$ where $N_0 = n_0 k$. For each alternative $i \in \mathcal{K}$, take n_0 observations, let $\bar{\mathfrak{X}}_i$ be the sample mean of these n_0 observations, and set $\mathfrak{N}_i = n_0$. Determine the sampling budget N_r that should be allocated to round r and ensure that $\sum_{r=1}^{\lceil \log_2 k \rceil} N_r \leq N$. Let $\mathcal{I}_r$ denote the set of alternatives that are still in contention at the beginning of round r. Set $r = 1$ and $\mathcal{I}_1 = \{1, 2, \ldots, k\}$.*

Step 2. (Pairing and Screening): *Let* $\mathcal{N}_r = \lfloor N_r/|\mathcal{I}_r|\rfloor$, $\mathcal{I}_{r+1} = \emptyset$, *and* $\mathcal{I}'_r = \mathcal{I}_r$. *Sort the alternatives in* $\mathcal{I}'_r$ *in ascending order based on their values of* $\bar{\mathfrak{X}}_i$. *If there is an odd number of alternatives in* $\mathcal{I}'_r$, *let alternative* j' *be the last alternative in* $\mathcal{I}'_r$, *and update* $\mathcal{I}_{r+1} = \mathcal{I}_{r+1} \cup \{j'\}$ *and* $\mathcal{I}'_r = \mathcal{I}'_r \backslash \{j'\}$.

For $v = 1$ *to* $\lfloor|\mathcal{I}_r|/2\rfloor$

- *Let alternatives* i *and* j *be the first and the last alternatives in* $\mathcal{I}'_r$ *respectively, take* $\mathcal{N}_r$ *observations for each of the two alternatives, and update* $\mathcal{I}'_r = \mathcal{I}'_r \backslash \{i,j\}$.
- *Let* $\bar{X}_i^r$ *and* $\bar{X}_j^r$ *denote the sample means of alternatives* i *and* j *calculated based on these* $\mathcal{N}_r$ *observations respectively. If* $\bar{X}_i^r \geq \bar{X}_j^r$, *then update* $\mathcal{I}_{r+1} = \mathcal{I}_{r+1} \bigcup \{i\}$, $\bar{\mathfrak{X}}_i = \left(\bar{\mathfrak{X}}_i \cdot \mathfrak{N}_i + \bar{X}_i^r \cdot \mathcal{N}_r\right)/(\mathfrak{N}_i + \mathcal{N}_r)$, *and* $\mathfrak{N}_i = \mathfrak{N}_i + \mathcal{N}_r$. *If* $\bar{X}_i^r < \bar{X}_j^r$, *then update* $\mathcal{I}_{r+1} = \mathcal{I}_{r+1} \bigcup \{j\}$, $\bar{\mathfrak{X}}_j = \left(\bar{\mathfrak{X}}_j \cdot \mathfrak{N}_j + \bar{X}_j^r \cdot \mathcal{N}_r\right)/(\mathfrak{N}_j + \mathcal{N}_r)$, *and* $\mathfrak{N}_j = \mathfrak{N}_j + \mathcal{N}_r$.

Endfor

Step 3. (Stopping Rule): *Update* $r = r+1$. *If* $|\mathcal{I}_r| = 1$, *let* φ *be the index of the alternative in* $\mathcal{I}_r$ *and select alternative* φ *as the best. Otherwise, go to* ***Step 2***.

In the $\mathcal{FBKT}^{\mathcal{S}}$ procedure, we pair the alternatives in almost the same way as a typical sports tournament does. We let $\bar{\mathfrak{X}}_i$ record the sample mean of alternative i calculated based on all its observations previously collected and use it to rank and seed alternatives in each round. In order to apply the seeding method in the first round, we add an initial stage to the $\mathcal{FBKT}^{\mathcal{S}}$ procedure and generate n_0 observations for each alternative to estimate its sample mean. It is worth noting that observations generated prior to round r can only be used to pair the alternatives in round r. While conducting "matches" in round r, we still require the procedure to use observations generated in that round to compare the alternatives. Therefore, all the results stated in Theorem 2 and Proposition 1 remain intact for the $\mathcal{FBKT}^{\mathcal{S}}$ procedure except that after the modifications, the procedure requires $N_0 + N$ observations to complete the selection. In practice, we recommend allocating 1/11 of the total sampling budget, i.e., $N_0 : N = 1:10$, to the initial stage.

4. Implementing the Procedure in Parallel Computing Environments

While solving a R&S problem, in order to get a meaningful result, one often needs to simulate every alternative many times. When the number of alternatives is very large, the limited computational power of a single processor restricts one from solving the problem in a reasonable amount of time. Therefore, to solve large-scale R&S problems, leveraging the computational power offered by parallel computing environments is crucial. As the practical performance of the $\mathcal{FBKT}^{\mathcal{S}}$ procedure tends to be better than that of the $\mathcal{FBKT}$ procedure, in this section, we consider implementing the $\mathcal{FBKT}^{\mathcal{S}}$ procedure in parallel computing environments to solve large-scale problems.

Because the $\mathcal{FBKT}^{\mathcal{S}}$ procedure has a relatively simple selection structure, the procedure can be implemented in parallel computing environments without much difficulty. However, there is a potential issue that needs to be addressed. To run the procedure in parallel computing environments, as the selection proceeds, it is possible that there are more processors than the alternatives that are still in contention. If one keeps using the $\mathcal{FBKT}^{\mathcal{S}}$ procedure to finish the remaining selection, it becomes impossible to ensure that every processor always can have a "match" to execute. It may cause some processors to stay idle for a relatively long time. In this situation, the selection is not fully parallelizable. When the number of processors is large, it can significantly slow down the selection process and ruin the benefits of parallel computing. To resolve this issue, we consider the following way of implementing the procedure in parallel computing environments.

Suppose that there are m processors in a parallel computing environment. At the very beginning of the selection process, we equally divide the alternatives into m groups and assign each processor one group of alternatives as well as approximately $(N_0+N)/m$ amount of sampling budget. Without any prior knowledge on the configuration of the means of the alternatives, the division of alternatives can be random. Then, the procedure conducts a two-phase selection. In the first phase, following the selection structure of the $\mathcal{FBKT}^{\mathcal{S}}$ procedure, if every processor only pairs the alternatives within a group, every processor can independently identify the local best alternative in the group. In the second phase, after finding the local best alternative, every processor uses the remaining sampling budget to generate some additional observations for the local best alternative. Then, based on the observations generated in the second phase, the procedure directly selects the alternative with the largest sample mean from the set of local best alternatives as the final best alternative. By doing so, we avoid using the $\mathcal{FBKT}^{\mathcal{S}}$ procedure to conduct the remaining selection of the m local best alternatives. We list the detailed description of the procedure as follows and call it the $\mathcal{FBKT}^{\mathcal{S}+}$ procedure.

Procedure $\mathcal{FBKT}^{\mathcal{S}+}$ [Parallel Implementation]

Step 1. (Initialization): *Set the total sampling budget N_0+N where $N_0=n_0k$ and the number of processors m. Let $R=\lceil\log_2 k/m\rceil$. Determine the sampling budget N_r' that should be allocated to round r in a processor and ensure that $\sum_{r=1}^{R+1}N_r'\le N/m$. For $s=1,2,\ldots,m$, let $\mathcal{I}_r^s$ denote the set of alternatives that are still in contention at the beginning of round r in processor s. For $i=1,2,\ldots,k$, set $\mathcal{I}_1^{(i \bmod m)+1}=\mathcal{I}_1^{(i \bmod m)+1}\bigcup\{i\}$.*

Step 2. (Parallel Selection): *Execute the selection in processor $s=1,2,\ldots,m$:*

- ***a.*** *For each alternative $i\in\mathcal{I}_1^s$, take n_0 observations, let $\bar{\mathfrak{X}}_i$ be the sample mean of these n_0 observations, and set $\mathfrak{N}_i=n_0$. Let $r=1$.*

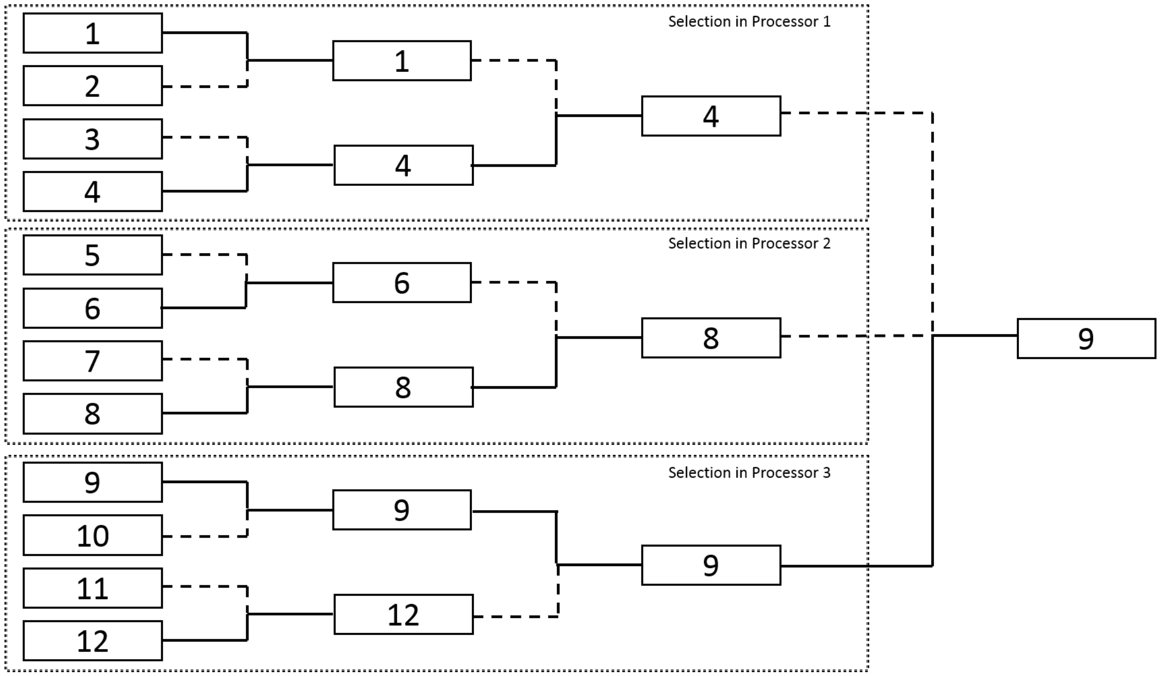

Figure 2 An illustration of implementing the $\mathcal{FBKT}^{\mathcal{S}+}$ procedure in a parallel computing environment with 3 processors.

b. *Let* $\mathcal{N}_r^s = \lfloor N_r' / |\mathcal{I}_r^s| \rfloor$, $\mathcal{I}_{r+1}^s = \emptyset$, *and* $\mathcal{I}_r^{s'} = \mathcal{I}_r^s$. *Sort the alternatives in* $\mathcal{I}_r^{s'}$ *in ascending order based on their values of* $\bar{\mathfrak{X}}_i$. *If there is an odd number of alternatives in* $\mathcal{I}_r^{s'}$, *let alternative* j' *be the last alternative in* $\mathcal{I}_r^{s'}$, *and update* $\mathcal{I}_{r+1}^s = \mathcal{I}_{r+1}^s \cup \{j'\}$ *and* $\mathcal{I}_r^{s'} = \mathcal{I}_r^{s'} \setminus \{j'\}$.

For $v = 1$ *to* $\lfloor |\mathcal{I}_r^s| / 2 \rfloor$

- *Let alternatives* i *and* j *be the first and the last alternatives in* $\mathcal{I}_r^{s'}$ *respectively, take* $\mathcal{N}_r^s$ *observations for each of the two alternatives, and update* $\mathcal{I}_r^{s'} = \mathcal{I}_r^{s'} \setminus \{i, j\}$.
- *Let* $\bar{X}_i^r$ *and* $\bar{X}_j^r$ *denote the sample means of alternatives* i *and* j *calculated based on these* $\mathcal{N}_r^s$ *observations respectively. If* $\bar{X}_i^r \geq \bar{X}_j^r$, *then update* $\mathcal{I}_{r+1}^s = \mathcal{I}_{r+1}^s \bigcup \{i\}$, $\bar{\mathfrak{X}}_i = \left(\bar{\mathfrak{X}}_i \cdot \mathfrak{N}_i + \bar{X}_i^r \cdot \mathcal{N}_r^s\right) / (\mathfrak{N}_i + \mathcal{N}_r^s)$, *and* $\mathfrak{N}_i = \mathfrak{N}_i + \mathcal{N}_r^s$. *If* $\bar{X}_i^r < \bar{X}_j^r$, *then update* $\mathcal{I}_{r+1}^s = \mathcal{I}_{r+1}^s \bigcup \{j\}$, $\bar{\mathfrak{X}}_j = \left(\bar{\mathfrak{X}}_j \cdot \mathfrak{N}_j + \bar{X}_j^r \cdot \mathcal{N}_r^s\right) / (\mathfrak{N}_j + \mathcal{N}_r^s)$, *and* $\mathfrak{N}_j = \mathfrak{N}_j + \mathcal{N}_r^s$.

Endfor

c. *Update* $r = r + 1$. *If* $|\mathcal{I}_r^s| = 1$, *let* φ_s *denote the index of the alternative in* $\mathcal{I}_r^s$ *and select alternative* φ_s *as the local best alternative in processor* s. *Otherwise go to* ***b***.

Step 3. (Selecting the Best): *Let* $\mathcal{I}_{final} = \{\varphi_1, \varphi_2, \ldots, \varphi_m\}$. *Set a (pseudo) random number stream* ϑ *and broadcast it to all processors. For processor* $s = 1, 2, \ldots, m$, *set* $r = R + 1$, *use the (pseudo) random number stream* ϑ *to generate additional* N_r' *observations for alternative* φ_s, *and let* $\bar{X}_{\varphi_s}^r$ *denote its sample mean. Select alternative* $\varphi = \arg\max_{i \in \mathcal{I}_{final}} \bar{X}_i^r$ *as the final output.*

In Figure 2, we give an example of the selection of twelve alternatives by the $\mathcal{FBKT}^{\mathcal{S}+}$ procedure in a parallel computing environment with three processors. Typically, the $\mathcal{FBKT}^{\mathcal{S}+}$ procedure can stop in $\lceil \log_2 k/m \rceil + 1$ rounds. Because at most $\lceil k/m \rceil$ alternatives are assigned to each processor, following the selection structure of the $\mathcal{FBKT}^{\mathcal{S}}$ procedure, every processor can identify the local best alternative in $\lceil \log_2 k/m \rceil$ rounds. Therefore, the observations generated in round $\lceil \log_2 k/m \rceil + 1$ are used to compare m local best alternatives in the second phase. Notice that there are almost no non-parallelizable calculations in the $\mathcal{FBKT}^{\mathcal{S}+}$ procedure, except that to apply CRNs in the second phase, the procedure needs to broadcast a (pseudo) random number stream to every processor and use it to generate observations for different local best alternatives. It suggests that the procedure is well-suited for parallel computing environments. In the theorem listed below, we prove that after the modifications, it is still possible for the $\mathcal{FBKT}^{\mathcal{S}+}$ procedure to keep the PCS and PGS from decreasing to 0 as the total sampling budget grows linearly in k. However, the lower bounds for the PCS and PGS additionally depend on the number of processors m available in a parallel computing environment. Its proof is included in the appendix.

Theorem 3. *Let m be a constant. If $\phi \geq 2$ and $N'_r = \left\lfloor \frac{r}{\phi(\phi-1)} \left(\frac{\phi-1}{\phi}\right)^r \frac{N}{m} \right\rfloor$ for $r = 1, 2, \ldots, \lceil \log_2 k/m \rceil$ and $N'_r = \left\lfloor \frac{r}{\phi-1} \left(\frac{\phi-1}{\phi}\right)^r \frac{N}{m} \right\rfloor$ for $r = \lceil \log_2 k/m \rceil + 1$, the $\mathcal{FBKT}^{\mathcal{S}+}$ procedure can stop before $N_0 + N$ observations are used. Furthermore,*

(1) *suppose that Assumptions 2 and 3 hold. For any positive constant $\eta_2 \geq 2\phi^2$, if $N/k \geq \eta_2$, the PCS of the $\mathcal{FBKT}^{\mathcal{S}+}$ procedure satisfies that,*

$$PCS \geq \exp\left(-\frac{4\phi^2\pi^2\sigma^2_{upper}}{3\eta_2\delta^2}\right)\left(1-(m-1)\exp\left(-\frac{\left(\log_2 \frac{k}{m}+1\right)\eta_2\delta^2}{8\sigma^2_{upper}\phi}\left(\frac{2(\phi-1)}{\phi}\right)^{\log_2 k/m+1}\right)\right),$$

for all $k > m$;

(2) *suppose that Assumption 3 holds. For any positive constant $\eta'_2 \geq 2\phi^2$, $\phi \geq 3$, and IZ parameter $\delta > 0$, if $N/k \geq \eta'_2$, the PGS of the $\mathcal{FBKT}^{\mathcal{S}+}$ procedure satisfies that,*

$$PGS \geq \exp\left(-\frac{8\pi^2\sigma^2_{upper}\phi^3(\phi-1)}{3\eta'_2\delta^2\left(\sqrt{2\phi(\phi-1)}-\phi\right)^2}\right)\left(1-(m-1)\exp\left(-\frac{\left(\log_2\frac{k}{m}+1\right)\eta'_2\phi'\delta^2}{16\sigma^2_{upper}\phi(\phi-1)}\right)\right),$$

where $\phi' = \left(3\phi - 2 + 2\sqrt{2\phi(\phi-1)}\right)$.

Theorem 3 suggests that as we choose to compare m local best alternatives all at once in the second phase, there is some loss on the selection accuracy of the $\mathcal{FBKT}^{\mathcal{S}+}$ procedure. However, when there are a large number of alternatives, the loss is not large. It can be observed that as k increases, the second terms in the lower bounds for the PCS and the PGS increase to one.

5. Numerical Experiments

In this section, we examine the performances of our procedure under different settings. The goals of the numerical studies are threefold: 1) to compare the performances of our procedure and some existing procedures, and verify that the RMCS of our procedure is indeed in the order of k; 2) to show that the seeding method proposed in Section 3.4 improves the selection accuracy of our procedure significantly; 3) to illustrate that we can easily implement our procedure in parallel computing environments to solve large-scale problems. For the experiments in the first two subsections, we consider a simple setting where the observations are generated from normal distributions and conduct the experiments on a personal computer (PC). For the experiments in the third subsection, the observations are obtained by conducting real simulations. The programming language is Java. All the codes used in this section can be retrieved from `https://github.com/biazhong/FBKT`.

5.1. Comparing the $\mathcal{FBKT}$ Procedure and Existing Procedures

In this subsection, we use a simple example to test the performances of the $\mathcal{FBKT}$ procedure, the OCBA procedure, and the equal allocation (EA) procedure. The OCBA procedure is a well-known fixed-budget procedure. The procedure yields the near-optimal solution to Problem (1) by assuming that an infinite amount of sampling budget is allocated to each alternative. The EA procedure adopts a naive sample allocation. It equally allocates the sampling budget to different alternatives. For both procedures, when the total sampling budget is exhausted, the alternative with the largest sample mean is selected as the best. In this experiment, we consider the problem where the mean of the best alternative is 0.1 and all other alternatives share a common mean 0, i.e., $\mu_1 = \mu_2 = \cdots = \mu_{k-1} = \mu_k - 0.1 = 0$. Furthermore, the variances of all alternatives are equal and set to be 1, i.e., $\sigma_1^2 = \sigma_2^2 = \cdots = \sigma_k^2 = 1$. We conduct the experiment on a PC with Intel Core i9-11900K central processing unit (CPU), 64 Gigabytes (GB) memory, and Windows 10 operating system. For the $\mathcal{FBKT}$ procedure, while allocating the sampling budget to each round, we use the sample allocation rule listed in Theorem 2 and set positive integer $\phi = 3$. For the OCBA procedure, true values of all alternatives are used to determine the amount of sampling budget allocated to each alternative when we implement the procedure. We compare the performances of the three procedures by varying the number of alternatives $k = 10^\ell$, where $\ell = 1, 2, \ldots, 6$, and set the total sampling budget $N = 1.5 \times 10^3 k$. We plot the estimated PCS of each procedure against the number of alternatives k based on 1,000 independent macro replications in Figure 3.

Based on the results reported in Figure 3, we have two findings. First, as the number of alternatives k increases and the total sampling budget grows linearly in k, the $\mathcal{FBKT}$ procedure can

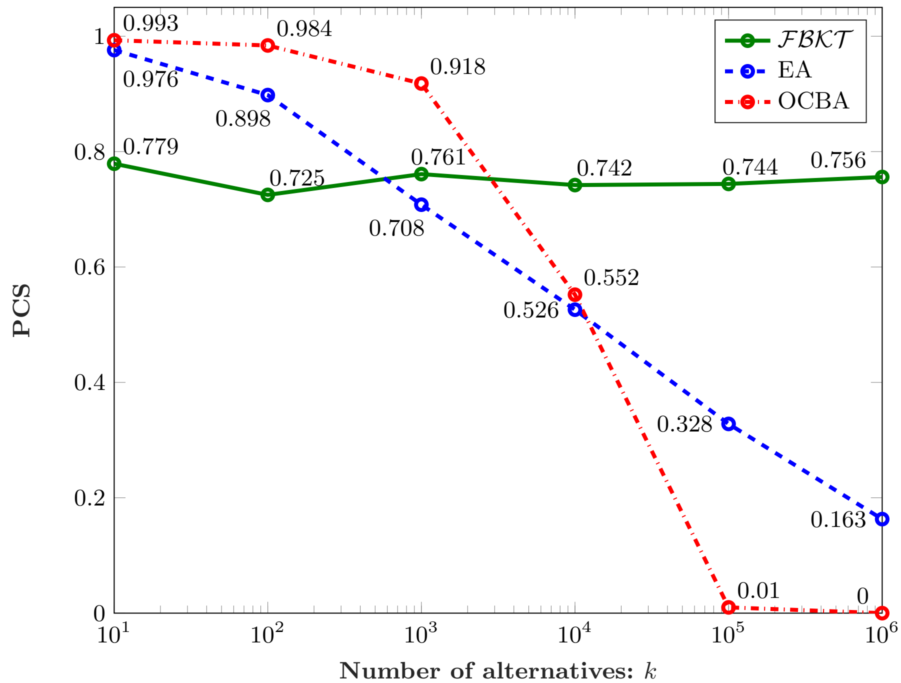

Figure 3 A comparison of the $\mathcal{FBKT}$ procedure, the EA procedure, and the OCBA procedure.

always deliver a PCS higher than 0.7. It is consistent with our analysis that the RMCS of the $\mathcal{FBKT}$ procedure is in the order of k. Meanwhile, as k increases, the PCS of the EA procedure and that of the OCBA procedure keep decreasing and eventually fall below that of the $\mathcal{FBKT}$ procedure when $k \geq 10^4$. It suggests that while solving large-scale problems, the $\mathcal{FBKT}$ procedure tends to deliver a higher PCS than existing procedures do. Second, while comparing the EA procedure and the OCBA procedure, we have an interesting finding. In this experiment, when $k \geq 10^5$, the OCBA procedure performs worse than the EA procedure (the most naive way of allocating sampling budget) does. It is probably because the OCBA procedure estimates the solution to Problem (1) by assuming that an infinite amount of sampling budget is allocated to each alternative. In this experiment, when $k \geq 10^5$, allocating roughly 1,500 observations to each alternative is not large enough to guarantee a good estimate.

5.2. The Use of the Seeding Method

In this subsection, we focus on investigating how our procedure may benefit from using the seeding method proposed in Section 3.4 to pair alternatives in each round. We conduct the experiment on the same PC as the one used in the previous experiment. Since in practice the means and variances of the alternatives often vary significantly, in this experiment, for each macro replication, we consider a mean configuration that, $\mu_i = -1.5(k-i)/k$ for $i = 1, 2, \ldots, k-1$ and $\mu_k = 0.1$, and let $\sigma_i^2 = 0.1 + \text{Unif}(0, 1.5)$ for $i = 1, 2, \ldots, k$, where $\text{Unif}(a', b')$ is the uniform distribution with

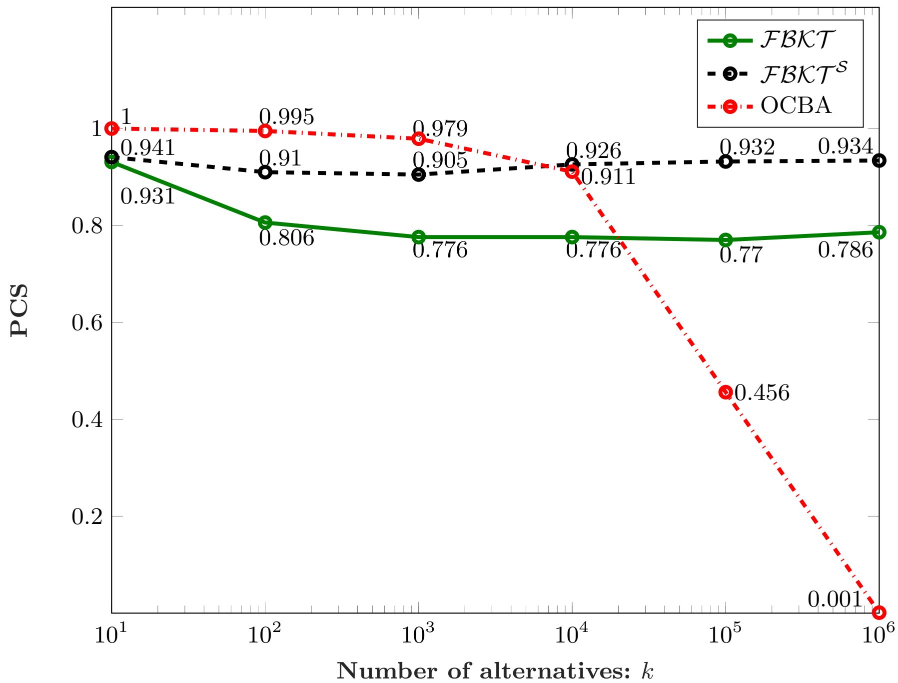

Figure 4 The use of the seeding method to pair the alternatives in each round of selection.

the support $[a', b']$. Similar to the previous experiment, we test the performances of the $\mathcal{FBKT}$ procedure and the $\mathcal{FBKT}^{\mathcal{S}}$ procedure with different values for k. As a reference, we also include the performance of the OCBA procedure in this experiment. To make a fair comparison, for the $\mathcal{FBKT}$ procedure and the OCBA procedure, we set $N = 100k$, and for the $\mathcal{FBKT}^{\mathcal{S}}$ procedure, we set $N_0 = 9k$ and $N = 91k$. For the $\mathcal{FBKT}$ procedure and the $\mathcal{FBKT}^{\mathcal{S}}$ procedure, the sample allocation rule listed in Theorem 2 is used and we still set positive integer $\phi = 3$. We plot the estimated PCS of each procedure against the number of alternatives k based on $1,000$ independent macro replications in Figure 4.

We highlight the main findings from this experiment as follows. First, both the $\mathcal{FBKT}$ procedure and the $\mathcal{FBKT}^{\mathcal{S}}$ procedure can keep the PCS from decreasing to 0 as the total sampling budget grows linearly in k. It suggests that the seeding method does not affect the order of the RMCS of our procedure. Second, because the means and variances of the alternatives are spread out in this particular experiment, the seeding method significantly improves the selection accuracy of our procedure. It can be observed that, in terms of the PCS, the $\mathcal{FBKT}^{\mathcal{S}}$ procedure performs uniformly better than the $\mathcal{FBKT}$ procedure does. The $\mathcal{FBKT}^{\mathcal{S}}$ procedure always delivers a PCS higher than 0.9. Third, the performance of the OCBA is similar to its performance in the previous experiment. When the number of alternatives k exceeds 10^4, the PCS of the procedure plunges.

5.3. A Throughput-Maximization Problem

In this subsection, we conduct the experiments in parallel computing environments. We test the performances of our procedure with a throughput-maximization problem taken from `SimOpt.org` (Henderson and Pasupathy 2021). The throughput-maximization problem is a practical simulation testing problem and is often used to examine the practical performance of a procedure in parallel computing environments (Luo et al. 2015, Ni et al. 2017, Zhong and Hong 2021, Zhong et al. 2021). The problem considers a flow line with three stations, i.e., 1, 2, and 3. An infinite number of jobs wait in front of station 1. Every job is sequentially processed by the three stations. The processing times of a job at stations 1, 2, and 3 are independently exponentially distributed with service rates $\mathtt{s}_1$, $\mathtt{s}_2$, and $\mathtt{s}_3$ respectively. At stations 2 and 3, there is finite buffer storage, denoted as $\mathtt{b}_2$ and $\mathtt{b}_3$ respectively. If the buffer at stations i, for $i = 2,3$, is fully occupied, station $i-1$ must hold the finished job until the job at station i is finished and released. Our objective is to find an allocation of the service rate and the buffer to maximize the expected steady-state throughput of the flow line subject to the constraints $\mathtt{s}_1 + \mathtt{s}_2 + \mathtt{s}_3 = \mathcal{L}_1$, $\mathtt{b}_2 + \mathtt{b}_3 = \mathcal{L}_2$, and $(\mathtt{s}_1, \mathtt{s}_2, \mathtt{s}_3, \mathtt{b}_2, \mathtt{b}_3) \in \mathbb{Z}_+^5$, where $\mathcal{L}_1$ and $\mathcal{L}_2$ are the problem parameters that specify the number of feasible solutions. For every feasible solution, we obtain observations by running simulation experiments. Specifically, for each simulation experiment, we warm-up the system with 2,000 jobs and observe the throughput of the subsequent 50 jobs. While conducting the experiments, we choose three different combinations of the values for $\mathcal{L}_1$ and $\mathcal{L}_2$. It corresponds to three problems with different numbers of alternatives. We summarize the basic information of the problems in Table 1.

Table 1 Summarized information of the throughput-maximization problems.

$(\mathcal{L}_1, \mathcal{L}_2)$	Number of alternatives: k	Largest mean: μ_k
$(20, 20)$	$3,249$	5.7761
$(50, 50)$	$57,624$	15.6976
$(128, 128)$	$1,016,127$	41.6582

In the rest of this section, we implement the procedures on Apache Spark and conduct the experiments on a local server with 48 processors, 128 GB of memory, and Red Hat Enterprise Linux (RHEL) 7.4 operating system. Apache Spark is a popular parallel computing platform widely supported by commercial clouds, including Amazon EC2/S3, Microsoft Azure, Google Cloud Platform, etc. Because the selection structures of most existing procedures are not suitable for parallel computing environments, while examining the performances of our procedure under different settings, we only add the performances of the EA procedure as a reference. To implement the EA procedure in parallel computing environments, we assign an equal number of alternatives to different processors at the very beginning of the selection process and let each processor independently simulate observations for the alternatives as assigned. After every processor finishes simulating observations, we select the alternative with the largest sample mean as the best. We use *utilization* to measure the average idle time of a processor while running a procedure in parallel computing environments. Because for a processor, most calculations are used to generate observations, we calculate the utilization of a procedure as follows,

$$\text{Utilization} = \frac{\text{total simulation time}}{\text{number of processors} \times \text{wall-clock time}},$$

where the total simulation time sums the time used to generate observations across all processors. The closer the utilization of a procedure is to one, the smaller proportion of the non-parallelizable calculations in the procedure.

5.3.1. The Problem with 3,249 Alternatives We first use the problem with 3,249 alternatives to test the performances of the $\mathcal{FBKT}^{\mathcal{S}+}$ procedure and the EA procedure. For each procedure, we consider two different versions with or without CRNs. For the $\mathcal{FBKT}^{\mathcal{S}+}$ procedure, the sample allocation rule listed in Theorem 3 is used and we let $\phi = 3$. Furthermore, for the procedure, we set $N_0 = 15 \times 3,249$ and $N = 150 \times 3,249$. For the EA procedure, we set $N = 165 \times 3,249$. Based on 100 independent macro replications, we estimate the PCS, the PGS, the wall-clock time, the total simulation time, and the utilization of each procedure. We also include 95% confidence intervals for the latter three estimates. For the PGS, we estimate the probability that the procedure selects an alternative whose mean lies within 0.01 to that of the best alternative, i.e., $\delta = 0.01$. We report the results in Table 2.

Table 2 A comparison of the $\mathcal{FBKT}^{\mathcal{S}+}$ procedure and the EA procedure using the configuration: $k = 3,249$ alternatives on 48 processors.

Procedure	PCS	PGS ($\delta = 0.01$)	Wall-clock time (sec)	Total simulation time ($\times 10^2$ sec)	Utilization
EA (without CRNs)	0.19	0.89	7.71 ± 0.03	3.61 ± 0.01	$97.66\% \pm 0.16\%$
$\mathcal{FBKT}^{\mathcal{S}+}$ (without CRNs)	0.38	0.98	7.49 ± 0.03	3.52 ± 0.01	$97.84\% \pm 0.16\%$
EA (with CRNs)	0.67	1.00	7.72 ± 0.03	3.62 ± 0.01	$97.61\% \pm 0.16\%$
$\mathcal{FBKT}^{\mathcal{S}+}$ (with CRNs)	0.95	1.00	7.50 ± 0.03	3.52 ± 0.01	$97.89\% \pm 0.17\%$

According to Table 2, we have four findings. First, in terms of the PCS, no matter whether the CRNs are used or not, the $\mathcal{FBKT}^{\mathcal{S}+}$ procedure outperforms the EA procedure, and for both procedures, the use of CRNs significantly increases the selection accuracy. Second, in this experiment, the PGS of each procedure is relatively high. It suggests that the procedures can select an alternative whose mean is very close to that of the best alternative when they cannot correctly select the best alternative. Third, observing the fourth column of Table 2, we may conclude that the $\mathcal{FBKT}^{\mathcal{S}+}$ procedure tends to use slightly less wall-clock time to identify the best alternative than the EA procedure does. It is because, while solving practical problems, there is always some sampling budget left unused in the $\mathcal{FBKT}^{\mathcal{S}+}$ procedure. However, all the sampling budget is used in the EA procedure. Fourth, the EA procedure is considered to have the most parallel-friendly selection structure. In this experiment, the utilization of the $\mathcal{FBKT}^{\mathcal{S}+}$ procedure is almost the same as that of the EA procedure and is very close to one. It suggests that the $\mathcal{FBKT}^{\mathcal{S}+}$ procedure is indeed very suitable for parallel computing environments.

To further investigate how the number of processors in the parallel computing environments may affect the performance of the $\mathcal{FBKT}^{\mathcal{S}+}$ procedure. Under the same settings as the ones used in the previous experiment, we separately let the $\mathcal{FBKT}^{\mathcal{S}+}$ procedure (without CRNs) solve the problem with different numbers of processors. We summarize the results in Table 3 based on 100 independent macro replications. To better illustrate the results, we also plot the estimated wall-clock time of the procedure against the number of processors in Figure 5.

Table 3 Examining the performance of the $\mathcal{FBKT}^{\mathcal{S}+}$ procedure (without CRNs) using the configuration: $k = 3,249$ alternatives on different numbers of processors.

Number of processors (m)	PCS	PCS ($\delta = 0.01$)	Wall-clock time (sec)	Total simulation time ($\times 10^2$ sec)	Utilization
8	0.61	1.00	28.96 ± 0.08	2.30 ± 0.01	$99.31\% \pm 0.18\%$
16	0.51	1.00	15.37 ± 0.03	2.44 ± 0.01	$99.30\% \pm 0.09\%$
24	0.44	1.00	11.03 ± 0.08	2.58 ± 0.01	$97.45\% \pm 0.35\%$
32	0.43	1.00	9.34 ± 0.03	2.84 ± 0.01	$95.05\% \pm 0.26\%$
40	0.43	1.00	8.18 ± 0.03	3.15 ± 0.01	$96.23\% \pm 0.21\%$
48	0.38	0.98	7.49 ± 0.03	3.52 ± 0.01	$97.84\% \pm 0.16\%$

From Table 3, we observe that as the number of processors m increases, the PCS of the $\mathcal{FBKT}^{\mathcal{S}+}$ procedure decreases. This is consistent with our analysis in Theorem 3 that as the procedure chooses to compare m local best alternatives all at once in the second phase, there is some loss on the selection accuracy. Furthermore, from Figure 5, we find that for the purpose of reducing the wall-clock time, it is always beneficial to add more processors. However, the marginal effect of reducing the wall-clock time by adding one more processor decreases as m increases. It is because in this experiment, the total computing resource, e.g., the memory of the local server, is limited. As more processors are running, the average amount of computing resource assigned to each processor decreases. Therefore, the average time needed to generate an observation increases so does the total simulation time (see the fifth column of Table 3). It offsets some potential speedup.

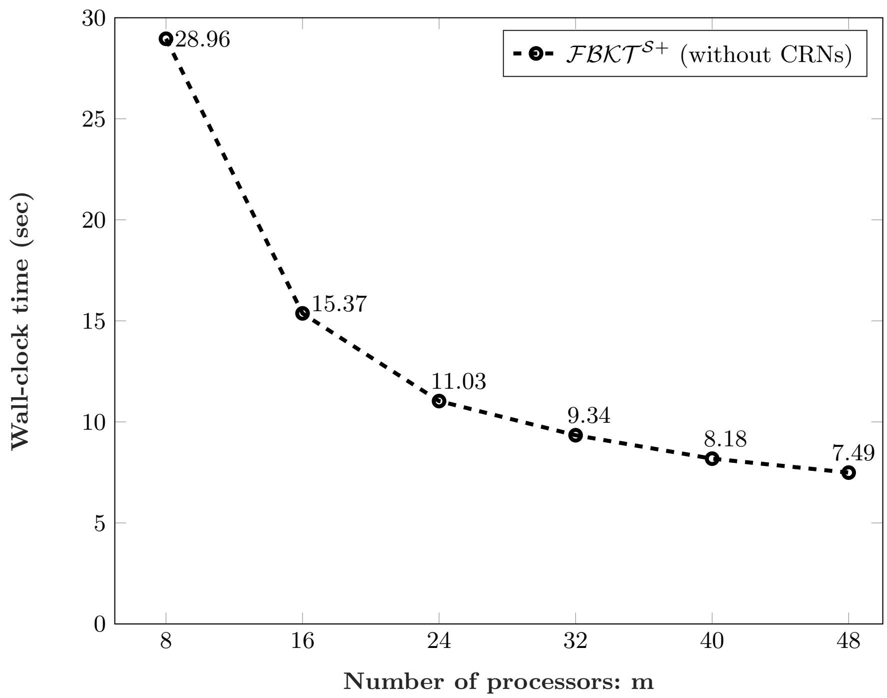

Figure 5 Reducing the wall-clock time by adding more processors.

5.4. The Problem with 57,624 Alternatives

Next, we use the problem with $57,624$ alternatives to test how the total sampling budget may affect the performances of the $\mathcal{FBKT}^{\mathcal{S}+}$ procedure. For the $\mathcal{FBKT}^{\mathcal{S}+}$ procedure, CRNs are used in this experiment. All other settings are the same as the ones used in the previous two experiments, i.e, the sample allocation rule listed in Theorem 3 is used and $\phi = 3$. We test the performances of the procedure by letting $N_0 = 2q \times 57,624$ and $N = 20q \times 57,624$, where $q = 1,2,4,8,16$. For each problem instance, we run 100 macro replications. Because in this experiment, even for the case where $q = 16$, the total sampling budget is not large enough to deliver a PCS close to one, we use the PGS ($\delta = 0.001$) to quantify the selection accuracy of the procedure. We report the results in Table 4.

Table 4 Examining the performance of the $\mathcal{FBKT}^{\mathcal{S}+}$ procedure (with CRNs) using the configuration: $k = 57,624$ with different total sampling budgets.

N_0 ($\times 57,624$)	N ($\times 57,624$)	PGS ($\delta = 0.001$)	Wall-clock time (sec)	Total simulation time ($\times 10^3$ sec)	Utilization
2	20	0.67	17.73 ± 0.04	0.840 ± 0.001	$98.66\% \pm 0.08\%$
4	40	0.70	35.52 ± 0.04	1.689 ± 0.001	$99.10\% \pm 0.05\%$
8	80	0.77	69.93 ± 0.07	3.334 ± 0.003	$99.33\% \pm 0.03\%$
16	160	0.83	140.15 ± 0.09	6.698 ± 0.004	$99.56\% \pm 0.02\%$
32	320	0.95	279.92 ± 0.13	13.393 ± 0.006	$99.68\% \pm 0.01\%$

From Table 4, we may conclude that, as the total sampling budget increases, the selection accuracy of the procedure increases. When $N_0 = 32 \times 57,624$ and $N = 320 \times 57,624$, the $\mathcal{FBKT}^{\mathcal{S}+}$ procedure can select an alternative whose mean lies within 0.001 to that of the best alternative with probability 0.95.

5.4.1. The Problem with 1,016,127 Alternatives In practice, the time needed to simulate an observation may vary from one observation to another. Lastly, we use the problem with $1,016,127$ alternatives to test how the variations in simulation times may affect the performances of the $\mathcal{FBKT}^{\mathcal{S}+}$ procedure. For the $\mathcal{FBKT}^{\mathcal{S}+}$ procedure, CRNs are used in this experiment and we set $N_0 = 4 \times 1,016,127$ and $N = 40 \times 1,016,127$. All other settings are the same as the ones used in the previous experiments. To demonstrate the variability in simulation times, during the selection process, after a processor generates an observation, we pause the processor for some random time ω. In previous experiments, we find that it generally takes a processor approximately 0.75 millisecond (ms) to simulate an observation. Therefore, we consider following three different settings for ω: (1) $\omega = 1\text{ms}$; (2) $\omega = \text{Unif}(0.5\text{ms}, 1.5\text{ms})$; (3) $\omega = \text{Unif}(0\text{ms}, 2\text{ms})$. These three settings represent three different levels of variability in simulation times. For each problem instance, we run 10 macro replications. Because fewer than 100 macro replications are conducted, we no longer report the estimated PCS and PGS of the procedure. Instead, we measure the mean of the selected alternative to quantify the selection accuracy of the procedure. We report the results in Table 5.

Table 5 Examining the performance of the $\mathcal{FBKT}^{\mathcal{S}+}$ procedure (with CRNs) with random simulation times using the configuration: $k = 1,016,127$ alternatives on 48 processors.

Pause time (ω)	Mean of the selected alternative	Wall-clock time ($\times 10^3$ sec)	Total simulation time ($\times 10^4$ sec)	Utilization
Unif (0ms, 2ms)	41.6539 ± 0.0009	1.496 ± 0.002	7.160 ± 0.004	$99.72\% \pm 0.14\%$
Unif (0.5ms, 1.5ms)	41.6541 ± 0.0013	1.497 ± 0.001	7.165 ± 0.003	$99.78\% \pm 0.13\%$
1ms	41.6537 ± 0.0019	1.497 ± 0.002	7.164 ± 0.003	$99.75\% \pm 0.13\%$

For the problem with 1,016,127 alternatives, the mean of the best alternative is 41.6582. From Table 5, we find that in this experiment, on average, the $\mathcal{FBKT}^{\mathcal{S}+}$ procedure can select an alternative whose mean lies within 0.005 to that of the best alternative. Furthermore, the wall-clock times, total simulations times, and utilizations of the procedure are almost the same in the three settings. It suggests that the practical performance of the procedure is relatively robust to the variations in simulations times.

6. Concluding Remarks

Solving large-scale fixed-budget R&S problems is fundamental but challenging. In this paper, we propose to use the RMCS to measure the performance of a fixed-budget procedure in solving large-scale problems, and establish a tight lower bound for the RMCS of a broad class of existing procedures. Then, we propose a new fixed-budget procedure, namely the $\mathcal{FBKT}$ procedure. We prove that its RMCS is in the order of k, which is superior to the existing fixed-budget procedures and achieves the optimal order. To further improve selection accuracy, we develop a seeding method to pair the alternatives in each round. Moreover, we demonstrate that our procedure can be easily implemented in parallel computing environments with almost no non-parallelizable calculations. Lastly, a comprehensive numerical study shows that our procedure is indeed very suitable for solving large-scale problems in parallel computing environments.

References

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

Bechhofer RE, Santner TJ, Goldsman DM (1995) *Design and Analysis of Experiments for Statistical Selection, Screening, and Multiple Comparisons* (New York: John Wiley & Sons).

Chen CH, Donohue K, Yücesan E, Lin J (2003) Optimal computing budget allocation for monte carlo simulation with application to product design. *Simulation Modelling Practice and Theory* 11(1):57–74.

Chen CH, Lin J, Yücesan E, Chick SE (2000) Simulation budget allocation for further enhancing the efficiency of ordinal optimization. *Discrete Event Dynamic Systems* 10(3):251–270.

Chick SE (2006) Subjective probability and bayesian methodology. Henderson SG, Nelson BL, eds., *Elsevier Handbooks in Operations Research and Management Science: Simulation*, 225–257 (Elsevier).

Chick SE, Branke J, Schmidt C (2010) Sequential sampling to myopically maximize the expected value of information. *INFORMS Journal on Computing* 22(1):71–80.

Chick SE, Inoue K (2001) New two-stage and sequential procedures for selecting the best simulated system. *Operations Research* 49(5):732–743.

Cormen TH, Leiserson CE, Rivest RL, Stein C (2001) *Introduction to Algorithms* (Cambridge, Massachusetts: The MIT press), 2nd edition.

Cramér H (1938) Sur un nouveau théoreme-limite de la théorie des probabilités. *Actualités Scienti ques et Industrielles* 736:5–23.

Dudewicz EJ, Dalal SR (1975) Allocation of observations in ranking and selection with unequal variances. *Sankhyā: The Indian Journal of Statistics* B37:28–78.

Even-Dar E, Mannor S, Mansour Y (2002) PAC bounds for multi-armed bandit and markov decision processes. Kivinen J, Sloan R, eds., *Proceedings of the 15th International Conference on Computational Learning Theory*, 255–270 (Berlin, Heidelberg: Springer).

Fan W, Hong LJ, Nelson BL (2016) Indifference-zone-free selection of the best. *Operations Research* 64(6):1499–1514.

Fan W, Hong LJ, Zhang X (2020) Distributionally robust selection of the best. *Management Science* 66(1):190–208.

Frazier PI, Powell W, Dayanik S (2008) A knowledge-gradient policy for sequential information collection. *SIAM Journal on Control and Optimization* 47(5):2410–2439.

Glynn P, Juneja S (2004) A large deviations perspective on ordinal optimization. Ingalls RG, Rossetti MD, Smith JS, Peters BA, eds., *Proceedings of the 2004 Winter Simulation Conference*, 577–585 (Piscataway, New Jersey: IEEE).

Henderson SG, Pasupathy R (2021) Simulation optimization library. `https://github.com/simopt-admin/simopt/wiki`, [Online; accessed 26-August-2021].

Hong LJ, Fan W, Luo J (2021) Review on ranking and selection: A new perspective. *Frontiers of Engineering Management* 8(3):321–343.

Hunter SR, Nelson BL (2017) Parallel ranking and selection. *Advances in Modeling and Simulation*, 249–275 (Springer).

Kim SH, Nelson BL (2001) A fully sequential procedure for indifference-zone selection in simulation. *ACM Transactions on Modeling and Computer Simulation* 11(3):251–273.

Kim SH, Nelson BL (2006) Selecting the best system. Henderson SG, Nelson BL, eds., *Elsevier Handbooks in Operations Research and Management Science: Simulation*, 501–534 (Elsevier).

Lan H, Nelson BL, Staum J (2010) A confidence interval procedure for expected shortfall risk measurement via two-level simulation. *Operations Research* 58(5):1481–1490.

Luo J, Hong LJ, Nelson BL, Wu Y (2015) Fully sequential procedures for large-scale ranking-and-selection problems in parallel computing environments. *Operations Research* 63(5):1177–1194.

Ni EC, Ciocan DF, Henderson SG, Hunter SR (2017) Efficient ranking and selection in parallel computing environments. *Operations Research* 65(3):821–836.

Paulson E (1964) A sequential procedure for selecting the population with the largest mean from k normal populations. *The Annals of Mathematical Statistics* 35(1):174–180.

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

Ryzhov IO (2016) On the convergence rates of expected improvement methods. *Operations Research* 64(6):1515–1528.

Shen H, Hong LJ, Zhang X (2021) Ranking and selection with covariates for personalized decision making. *INFORMS Journal on Computing* Forthcoming.

Waeber R, Frazier PI, Henderson SG (2010) Performance measures for ranking and selection procedures. Johansson B, Jain S, Montoya-Torres J, Hugan J, Yücesan E, eds., *Proceedings of the 2010 Winter Simulation Conference*, 1235–1245 (Piscataway, New Jersey: IEEE).

Xiao H, Lee LH, Morrice D, Chen CH, Hu X (2021) Ranking and selection for terminating simulation under sequential sampling. *IISE Transactions* 53(7):735–750.

Zhong Y, Hong LJ (2021) Knockout-tournament procedures for large-scale ranking and selection in parallel computing environments. *Operations Research* forthcoming.

Zhong Y, Liu S, Luo J, Hong LJ (2021) Speeding up paulson's procedure for large-scale problems using parallel computing. *INFORMS Journal on Computing* forthcoming.

Appendix A: Proof of Lemmas

A.1. Proof of Lemma 2

If $F=\prod_{r=1}^{\infty}\left(1-\exp\left(-\varrho r\right)\right)$ for some constant $\varrho>0$, then we have

$$\log F=\sum_{r=1}^{\infty}\log\left(1-\exp\left(-\varrho r\right)\right). \tag{12}$$

Because the Taylor series for function $\log(1-x)$ is $-x-x^2/2-x^3/3-\cdots$, we can rewrite Equation (12) as,

$$\begin{aligned}(12)&=-\sum_{r=1}^{\infty}\sum_{\gamma=1}^{\infty}\frac{\exp\left(-\varrho\gamma r\right)}{\gamma}\\&=-\sum_{\gamma=1}^{\infty}\sum_{r=1}^{\infty}\frac{\exp\left(-\varrho\gamma r\right)}{\gamma}\\&=-\sum_{\gamma=1}^{\infty}\frac{1}{\gamma}\frac{\exp\left(-\varrho\gamma\right)}{1-\exp\left(-\varrho\gamma\right)}\\&\geq-\sum_{\gamma=1}^{\infty}\frac{1}{\gamma}\frac{1}{\varrho\gamma}=-\frac{1}{\varrho}\sum_{\gamma=1}^{\infty}\frac{1}{\gamma^2}=-\frac{\pi^2}{6\varrho},\end{aligned}$$

where the inequality holds because $\exp\left(-\varrho\gamma\right)/\left(1-\exp\left(-\varrho\gamma\right)\right)\leq 1/\left(\varrho\gamma\right)$ for any $\varrho\gamma>0$. It concludes the proof. $\square$

A.2. Proof of Lemma 3

By the Chernoff bound, for any positive constant $\kappa>0$, we have

$$\mathbb{P}\left(Z\geq d\right)\leq\frac{\mathbb{E}\left[\exp\left(\kappa Z\right)\right]}{\exp\left(\kappa d\right)}=\exp\left(-\frac{1}{2}\kappa^2-\kappa d\right). \tag{13}$$

Letting $\kappa=d$, Equation (13) yields,

$$\mathbb{P}\left(Z\geq d\right)\leq\exp\left(-\frac{d^2}{2}\right). \quad \square$$

Appendix B: Proof of Propositions

B.1. Proof of Proposition 1

We prove Proposition 1 in a similar way to that of Theorem 2. We show that if the sample allocation rule $N_r = \left\lfloor \frac{r}{\phi(\phi-1)} \left(\frac{\phi-1}{\phi}\right)^r N \right\rfloor$ is used and $\phi \geq 3$, for any positive constant $\eta_1' \geq 2\phi^2$ and IZ parameter $\delta > 0$, the PGS of the $\mathcal{FBKT}$ procedure is lower bounded for all $k \geq 2$. We first let $\hat{k} = 2^{\lceil \log_2 k \rceil}$. Notice that the procedure can select the best alternative in $\log_2 \hat{k}$ rounds. For $1 \leq r \leq \log_2 \hat{k}$, we let ζ_r denote the index of the alternative with the largest mean among the alternatives that are still in contention at the beginning of round r, i.e., $\mu_{\zeta_r} = \max_{i \in \mathcal{I}_r} \mu_i$, and ζ_r' denote the index of the alternative that competes with alternative ζ_r in round r. To ease the notation, we set $\zeta_r = \varphi$ for $r = \log_2 \hat{k} + 1$, where φ is the index of the alternative which is finally selected by the procedure. We also let,

$$\delta_r = \frac{\sqrt{2(\phi-1)} - \sqrt{\phi}}{\sqrt{\phi}} \left(\frac{\phi}{2(\phi-1)}\right)^{\frac{r}{2}} \delta,$$

for $r \geq 1$. It can be verified that if $\phi \geq 3$, $\delta_r > 0$ for $r \geq 1$ and $\sum_{r=1}^{\infty} \delta_r = \delta$. For $1 \leq r \leq \log_2 \hat{k}$, we let $\mathcal{Q}_r' = \{\mu_{\zeta_r} - \mu_{\zeta_r'} \leq \delta_r\} \cup \{\mu_{\zeta_r} - \mu_{\zeta_r'} > \delta_r \text{ and } \bar{X}_{\zeta_r}^r \geq \bar{X}_{\zeta_r'}^r\}$, and $\mathcal{Q}_r'^c$ denote the complement of event $\mathcal{Q}_r'$, i.e., $\mathcal{Q}_r'^c = \{\mu_{\zeta_r} - \mu_{\zeta_r'} > \delta_r \text{ and } \bar{X}_{\zeta_r}^r < \bar{X}_{\zeta_r'}^r\}$. Then, we can bound the PGS of the $\mathcal{FBKT}$ procedure as follows,

$$\begin{aligned}
\text{PGS} &\geq \mathbb{P}\left(\bigcap_{r=1}^{\log_2 \hat{k}} \left\{\mu_{\zeta_r} - \mu_{\zeta_{r+1}} \leq \delta_r\right\}\right) \\
&\geq \mathbb{P}\left(\bigcap_{r=1}^{\log_2 \hat{k}} \{\mathcal{Q}_r'\}\right) \\
&= \mathbb{P}(\mathcal{Q}_1') \prod_{r=2}^{\log_2 \hat{k}} \mathbb{P}\left(\mathcal{Q}_r' \middle| \mathcal{Q}_1', \ldots, \mathcal{Q}_{r-1}'\right) \\
&= \left(1 - \mathbb{P}\left(\mathcal{Q}_1'^c\right)\right) \prod_{r=2}^{\log_2 \hat{k}} \left(1 - \mathbb{P}\left(\mathcal{Q}_r'^c \middle| \mathcal{Q}_1', \ldots, \mathcal{Q}_{r-1}'\right)\right),
\end{aligned} \tag{14}$$

where the first inequality holds because if the mean difference between alternatives ζ_r and ζ_{r+1} is less than δ_r for $1 \leq r \leq \log_2 \hat{k}$, the mean difference between alternative φ and alternative k is less than $\sum_{r=1}^{\infty} \delta_r = \delta$, and the second inequality holds because event $\mathcal{Q}_r'$ implies event $\{\mu_{\zeta_r} - \mu_{\zeta_{r+1}} \leq \delta_r\}$. Notice that based on the analysis used in Equation (9) in the proof of Theorem 2, we can conclude that,

$$\mathcal{N}_r \geq r \left\lfloor \frac{\eta_1'}{4\phi(\phi-1)} \left(\frac{2(\phi-1)}{\phi}\right)^r \right\rfloor.$$

Then, we can rewrite Equation (14) as follows,

$$(14) = \left(1 - \mathbb{P}\left(\mu_{\zeta_1} - \mu_{\zeta_1'} > \delta_1 \text{ and } \bar{X}_{\zeta_1}^1 < \bar{X}_{\zeta_1'}^1\right)\right) \times$$

$$
\begin{aligned}
&\prod_{r=2}^{\log_2 \hat{k}} \left(1-\mathbb{P}\left(\mu_{\zeta_r}-\mu_{\zeta_r'}>\delta_r \text{ and } \bar{X}_{\zeta_r}^r<\bar{X}_{\zeta_r'}^r \middle| \mathcal{Q}_1',\ldots,\mathcal{Q}_{r-1}'\right)\right)\\
&=\left(1-\mathbb{P}\left(\mu_{\zeta_1}-\mu_{\zeta_1'}>\delta_1 \text{ and } \frac{-\bar{X}_{\zeta_1}^1+\bar{X}_{\zeta_1'}^1+\mu_{\zeta_1}-\mu_{\zeta_1'}}{\sigma_{\zeta_1\zeta_1'}/\sqrt{\mathcal{N}_1}}>\sqrt{\mathcal{N}_1}\frac{\mu_{\zeta_1}-\mu_{\zeta_1'}}{\sigma_{\zeta_1\zeta_1'}}\right)\right)\times\\
&\prod_{r=2}^{\log_2 \hat{k}} \left(1-\mathbb{P}\left(\mu_{\zeta_r}-\mu_{\zeta_r'}>\delta_r \text{ and } \frac{-\bar{X}_{\zeta_r}^r+\bar{X}_{\zeta_r'}^r+\mu_{\zeta_r}-\mu_{\zeta_r'}}{\sigma_{\zeta_r\zeta_r'}/\sqrt{\mathcal{N}_r}}>\sqrt{\mathcal{N}_r}\frac{\mu_{\zeta_r}-\mu_{\zeta_r'}}{\sigma_{\zeta_r\zeta_r'}}\middle|\mathcal{Q}_1',\ldots,\mathcal{Q}_{r-1}'\right)\right)\\
&\geq\left(1-\mathbb{P}\left(\mu_{\zeta_1}-\mu_{\zeta_1'}>\delta_1 \text{ and } Z\geq\sqrt{\left\lfloor\frac{\eta_1'}{2\phi^2}\right\rfloor}\frac{\delta_1}{\sigma_{upper}}\right)\right)\times\\
&\prod_{r=2}^{\log_2 \hat{k}} \left(1-\mathbb{P}\left(\mu_{\zeta_r}-\mu_{\zeta_r'}>\delta_r \text{ and } Z>\sqrt{r\left\lfloor\frac{\eta_1'}{4\phi(\phi-1)}\left(\frac{2(\phi-1)}{\phi}\right)^r\right\rfloor}\frac{\delta_r}{\sigma_{upper}}\middle|\mathcal{Q}_1',\ldots,\mathcal{Q}_{r-1}'\right)\right)\\
&\geq\left(1-\mathbb{P}\left(Z\geq\sqrt{\left\lfloor\frac{\eta_1'}{2\phi^2}\right\rfloor}\frac{\delta_1}{\sigma_{upper}}\right)\right)\times\\
&\prod_{r=2}^{\log_2 \hat{k}} \left(1-\mathbb{P}\left(Z>\sqrt{r\left\lfloor\frac{\eta_1'}{4\phi(\phi-1)}\left(\frac{2(\phi-1)}{\phi}\right)^r\right\rfloor}\frac{\delta_r}{\sigma_{upper}}\middle|\mathcal{Q}_1',\ldots,\mathcal{Q}_{r-1}'\right)\right)\\
&=\prod_{r=1}^{\log_2 \hat{k}} \left(1-\mathbb{P}\left(Z>\sqrt{r\left\lfloor\frac{\eta_1'}{4\phi(\phi-1)}\left(\frac{2(\phi-1)}{\phi}\right)^r\right\rfloor}\frac{\delta_r}{\sigma_{upper}}\right)\right),
\end{aligned}
\tag{15}
$$

where the last equality holds because event $\{Z>\sqrt{r\lfloor\eta_1'(2(\phi-1)/\phi)^r/(4\phi(\phi-1))\rfloor}\delta_r/\sigma_{upper}\}$ is independent of events $\{\mathcal{Q}_1',\ldots,\mathcal{Q}_{r-1}'\}$. Then, applying Lemma 3 to Equation (15), we have,

$$
\begin{aligned}
(15)\geq&\prod_{r=1}^{\log_2 \hat{k}}\left(1-\exp\left(-r\left\lfloor\frac{\eta_1'}{4\phi(\phi-1)}\left(\frac{2(\phi-1)}{\phi}\right)^r\right\rfloor\frac{\delta_r^2}{2\sigma_{upper}^2}\right)\right)\\
=&\prod_{r=1}^{\log_2 \hat{k}}\left(1-\exp\left(-r\left\lfloor\frac{\eta_1'}{4\phi(\phi-1)}\left(\frac{2(\phi-1)}{\phi}\right)^r\right\rfloor\frac{\left(\frac{\sqrt{2(\phi-1)}-\sqrt{\phi}}{\sqrt{\phi}}\right)^2\left(\frac{\phi}{2(\phi-1)}\right)^r\delta^2}{2\sigma_{upper}^2}\right)\right)\\
\geq&\prod_{r=1}^{\log_2 \hat{k}}\left(1-\exp\left(-r\frac{\eta_1'}{8\phi(\phi-1)}\left(\frac{2(\phi-1)}{\phi}\right)^r\frac{\left(\frac{\sqrt{2(\phi-1)}-\sqrt{\phi}}{\sqrt{\phi}}\right)^2\left(\frac{\phi}{2(\phi-1)}\right)^r\delta^2}{2\sigma_{upper}^2}\right)\right)\\
=&\prod_{r=1}^{\log_2 \hat{k}}\left(1-\exp\left(-r\frac{\eta_1'\delta^2}{8\sigma_{upper}^2}\left(\frac{1}{\phi}-\frac{1}{\sqrt{2\phi(\phi-1)}}\right)^2\right)\right),
\end{aligned}
\tag{16}
$$

where the second inequality holds because if $a\geq1$, then $\lfloor a\rfloor\geq a/2$. Applying Lemma 2 to Equation (16), it yields,

$$
(16)\geq\exp\left(-\frac{8\pi^2\sigma_{upper}^2\phi^3(\phi-1)}{3\eta_1'\delta^2\left(\sqrt{2\phi(\phi-1)}-\phi\right)^2}\right).
$$

Therefore, if $N/k \geq \eta_1'$, the PGS of the $\mathcal{FBKT}$ procedure is lower bounded by $\exp(-4\pi^2\sigma_{upper}^2/(3\eta_1'\delta^2(1/\phi - 1/\sqrt{2\phi(\phi-1)})^2))$. It concludes the proof. $\square$

Appendix C: Proof of Theorems

C.1. Proof of Theorem 3

Let $R = \lceil \log_2 k/m \rceil$. We first show that if $\phi \geq 2$ and $N_r' = \left\lfloor \frac{r}{\phi(\phi-1)} \left(\frac{\phi-1}{\phi}\right)^r \frac{N}{m} \right\rfloor$ for $r = 1, 2, \ldots, R$ and $N_r' = \left\lfloor \frac{r}{\phi-1} \left(\frac{\phi-1}{\phi}\right)^r \frac{N}{m} \right\rfloor$ for $r = R+1$, the $\mathcal{FBKT}^{\mathcal{S}+}$ procedure can stop before $N_0 + N$ observations are used. It can be checked that

$$\begin{aligned}
\sum_{r=1}^{R+1} N_r' &\leq \sum_{r=1}^{R} \frac{r}{\phi(\phi-1)} \left(\frac{\phi-1}{\phi}\right)^r \frac{N}{m} + \frac{R+1}{\phi-1}\left(\frac{\phi-1}{\phi}\right)^{R+1} \frac{N}{m} \\
&= \sum_{r=1}^{\infty} \frac{r}{\phi(\phi-1)} \left(\frac{\phi-1}{\phi}\right)^r \frac{N}{m} + \frac{R+1}{\phi-1}\left(\frac{\phi-1}{\phi}\right)^{R+1} \frac{N}{m} - \sum_{r=R+1}^{\infty} \frac{r}{\phi(\phi-1)} \left(\frac{\phi-1}{\phi}\right)^r \frac{N}{m}.
\end{aligned} \tag{17}$$

For the first term in Equation (17), we have,

$$\begin{aligned}
\sum_{r=1}^{\infty} \frac{r}{\phi(\phi-1)} \left(\frac{\phi-1}{\phi}\right)^r \frac{N}{m} &= \phi \left[\sum_{r=1}^{\infty} \frac{r}{\phi(\phi-1)} \left(\frac{\phi-1}{\phi}\right)^r \frac{N}{m} - \sum_{r=1}^{\infty} \frac{r}{\phi(\phi-1)} \left(\frac{\phi-1}{\phi}\right)^{r+1} \frac{N}{m}\right] \\
&= \phi \left[\frac{N}{\phi(\phi-1)m} \sum_{r=1}^{\infty} \left(\frac{\phi-1}{\phi}\right)^r\right] \\
&= \frac{N}{m}.
\end{aligned} \tag{18}$$

For the third term in Equation (17), we have,

$$\begin{aligned}
\sum_{r=R+1}^{\infty} \frac{r}{\phi(\phi-1)} \left(\frac{\phi-1}{\phi}\right)^r \frac{N}{m} &= \sum_{r=1}^{\infty} \frac{r+R}{\phi(\phi-1)} \left(\frac{\phi-1}{\phi}\right)^{r+R} \frac{N}{m} \\
&= \frac{N}{\phi(\phi-1)m} \left(\frac{\phi-1}{\phi}\right)^R \left[\sum_{r=1}^{\infty} r \left(\frac{\phi-1}{\phi}\right)^r + R \sum_{r=1}^{\infty} \left(\frac{\phi-1}{\phi}\right)^r\right] \\
&= \frac{N}{\phi(\phi-1)m} \left(\frac{\phi-1}{\phi}\right)^R [\phi(\phi-1) + R(\phi-1)] \\
&= \frac{N}{m}\left(\frac{\phi-1}{\phi}\right)^R + \frac{NR}{\phi m}\left(\frac{\phi-1}{\phi}\right)^R.
\end{aligned} \tag{19}$$

Plugging the results in Equations (18) and (19) into Equation (17) yields,

$$\begin{aligned}
\sum_{r=1}^{R+1} N_r' &\leq \sum_{r=1}^{\infty} \frac{r}{\phi(\phi-1)} \left(\frac{\phi-1}{\phi}\right)^r \frac{N}{m} + \frac{R+1}{\phi-1}\left(\frac{\phi-1}{\phi}\right)^{R+1} \frac{N}{m} - \sum_{r=R+1}^{\infty} \frac{r}{\phi(\phi-1)} \left(\frac{\phi-1}{\phi}\right)^r \frac{N}{m} \\
&= \frac{N}{m} + \frac{(R+1)N}{\phi m}\left(\frac{\phi-1}{\phi}\right)^R - \frac{N}{m}\left(\frac{\phi-1}{\phi}\right)^R - \frac{NR}{\phi m}\left(\frac{\phi-1}{\phi}\right)^R \\
&= \frac{N}{m} - \frac{N}{m}\left(\frac{\phi-1}{\phi}\right)^{R+1} \\
&\leq \frac{N}{m}.
\end{aligned} \tag{20}$$

The results stated in Equation (20) suggests that if the sample allocation rule listed in Theorem 3 is used, processor s for $s=1,2,\ldots,m$ simulates fewer than $|\mathcal{I}_1^s|\,n_0+N/m$ observations during the entire selection process. Notice that sets $\mathcal{I}_1^1,\mathcal{I}_1^2,\ldots,\mathcal{I}_1^m$ are mutually exclusive and $\cup_{s=1}^m\mathcal{I}_1^s=\mathcal{K}$. It implies that the total number of observations simulated by all processors is upper bounded by $\sum_{s=1}^m|\mathcal{I}_1^s|\,n_0+N/m\le N_0+N$. Therefore, we can conclude that the $\mathcal{FBKT}^{\mathcal{S}+}$ procedure stops before all N_0+N observations are used. Then, we prove the first part of Theorem 3.

Proof of Part (1) Given that Assumptions 2 and 3 hold, we aim to show that for any positive constant $\eta_2\ge 2\phi^2$, if $\phi\ge 2$ and $N/k\ge\eta_2$, the PCS of the $\mathcal{FBKT}^{\mathcal{S}+}$ procedure can be lower bounded. To prove this, we first let $\hat{k}'=2^R$. Notice that the $\mathcal{FBKT}^{\mathcal{S}+}$ procedure equally allocates k alternatives to m processors. Each processor handles the selection of at most $\lceil k/m\rceil$ alternatives, and can identify the local best alternative in R rounds. For $1\le r\le R$ and $s=1,2,\ldots,m$, at the beginning of round r in processor s, there are at most $\hat{k}'/2^{r-1}$ alternatives that are still in contention, i.e., $|\mathcal{I}_r^s|\le\hat{k}'/2^{r-1}$. Let s_k denote the processor where alternative k is assigned and k_r denote the alternative that competes with alternative k in round r in processor s_k. For $1\le r\le R$, we define $\mathcal{E}_r$ as the event that alternative k eliminates alternative k_r in round r in processor s_k, i.e., $\mathcal{E}_r=\left\{\bar{X}_k^r\ge\bar{X}_{k_r}^r\right\}$. For $r=R+1$, we define $\mathcal{E}_r$ as the event that alternative k eliminates all other alternatives in $\mathcal{I}_{final}$, i.e., $\{\bar{X}_k^{R+1}\ge\max_{i\in\mathcal{I}_{final}\setminus\{k\}}\bar{X}_i^{R+1}\}$. For $1\le r\le R+1$, we let $\mathcal{E}_r^c$ be the complement of event $\mathcal{E}_r$. Then, we can write the PCS of the $\mathcal{FBKT}^{\mathcal{S}+}$ procedure as follows,

$$\mathrm{PCS}=\mathbb{P}\left(\bigcap_{r=1}^{R+1}\mathcal{E}_r\right)=\mathbb{P}\left(\mathcal{E}_1\right)\prod_{r=2}^{R}\mathbb{P}\left(\mathcal{E}_r|\mathcal{E}_1,\ldots,\mathcal{E}_{r-1}\right)\mathbb{P}\left(\mathcal{E}_{R+1}|\mathcal{E}_1,\ldots,\mathcal{E}_R\right).\tag{21}$$

For $1\le r\le R$, we establish a lower bound for the sampling budget that should be allocated each alternative in round r in processor s_k as follows,

$$\begin{aligned}\mathcal{N}_r^{s_k}&=\left\lfloor\frac{N_r'}{|\mathcal{I}_r^{s_k}|}\right\rfloor\\&=\left\lfloor\frac{r}{\phi\left(\phi-1\right)}\left(\frac{\phi-1}{\phi}\right)^r\frac{N}{m\,|\mathcal{I}_r^{s_k}|}\right\rfloor\\&\ge\left\lfloor\frac{r}{\phi\left(\phi-1\right)}\left(\frac{\phi-1}{\phi}\right)^r\frac{\eta_2 k}{m\frac{\hat{k}'}{2^{r-1}}}\right\rfloor\\&=\left\lfloor\frac{r}{\phi\left(\phi-1\right)}\left(\frac{2\left(\phi-1\right)}{\phi}\right)^r\frac{\eta_2 k}{2m\hat{k}'}\right\rfloor\\&\ge r\left\lfloor\frac{\eta_2}{4\phi\left(\phi-1\right)}\left(\frac{2\left(\phi-1\right)}{\phi}\right)^r\right\rfloor,\end{aligned}\tag{22}$$

where the first inequality holds because $N\ge\eta_2 k$ and $|\mathcal{I}_r^{s_k}|\le\hat{k}'/2^{r-1}$ for $1\le r\le R$. The second inequality holds because by the definition of $\hat{k}'$, $\hat{k}'<2k/m$. Notice that $\eta_2\ge 2\phi^2$ ensures that $\mathcal{N}_r^{s_k}$

is a positive integer no less than r. Then, based on the results listed in Equation (22), we can write the first two terms in Equation (21) as follows,

$$\begin{aligned}
&\mathbb{P}(\mathcal{E}_1)\prod_{r=2}^{R}\mathbb{P}(\mathcal{E}_r|\mathcal{E}_1,\ldots,\mathcal{E}_{r-1})\\
&=(1-\mathbb{P}(\mathcal{E}_1^c))\prod_{r=2}^{R}(1-\mathbb{P}(\mathcal{E}_r^c|\mathcal{E}_1,\ldots,\mathcal{E}_{r-1}))\\
&=\left(1-\mathbb{P}\left(\bar{X}_{k_1}^1>\bar{X}_k^1\right)\right)\prod_{r=2}^{R}\left(1-\mathbb{P}\left(\bar{X}_{k_r}^r>\bar{X}_k^r\middle|\mathcal{E}_1,\ldots,\mathcal{E}_{r-1}\right)\right)\\
&=\left(1-\mathbb{P}\left(\frac{\bar{X}_{k_1}^1-\bar{X}_k^1-\mu_{k_1}+\mu_k}{\frac{\sigma_{kk_1}}{\sqrt{\mathcal{N}_1^{s_k}}}}>\sqrt{\mathcal{N}_1^{s_k}}\frac{\mu_k-\mu_{k_1}}{\sigma_{kk_1}}\right)\right)\times\\
&\prod_{r=2}^{R}\left(1-\mathbb{P}\left(\frac{\bar{X}_{k_r}^r-\bar{X}_k^r-\mu_{k_r}+\mu_k}{\frac{\sigma_{kk_r}}{\sqrt{\mathcal{N}_r^{s_k}}}}>\sqrt{\mathcal{N}_r^{s_k}}\frac{\mu_k-\mu_{k_r}}{\sigma_{kk_r}}\middle|\mathcal{E}_1,\ldots,\mathcal{E}_{r-1}\right)\right)\\
&\geq\left(1-\mathbb{P}\left(Z>\sqrt{\left\lfloor\frac{\eta_2}{2\phi^2}\right\rfloor}\frac{\delta}{\sigma_{upper}}\right)\right)\prod_{r=2}^{R}\left(1-\mathbb{P}\left(Z>\sqrt{r\left\lfloor\frac{\eta_2}{4\phi(\phi-1)}\left(\frac{2(\phi-1)}{\phi}\right)^r\right\rfloor}\frac{\delta}{\sigma_{upper}}\middle|\mathcal{E}_1,\ldots\mathcal{E}_{r-1}\right)\right)\\
&\geq\left(1-\mathbb{P}\left(Z>\sqrt{\left\lfloor\frac{\eta_2}{2\phi^2}\right\rfloor}\frac{\delta}{\sigma_{upper}}\right)\right)\prod_{r=2}^{R}\left(1-\mathbb{P}\left(Z>\sqrt{r\left\lfloor\frac{\eta_2}{2\phi^2}\right\rfloor}\frac{\delta}{\sigma_{upper}}\middle|\mathcal{E}_1,\ldots\mathcal{E}_{r-1}\right)\right),
\end{aligned}\tag{23}$$

where the first inequality holds due to Equation (22) and Assumptions 2 and 3. The second inequality holds due to the fact that $[2(\phi-1)/\phi]^r\geq 2(\phi-1)/\phi$ when $r\geq 2$ and $\phi\geq 2$. Because event $\left\{Z>\sqrt{r\lfloor\eta_2/(2\phi^2)\rfloor}\delta/\sigma_{upper}\right\}$ is independent of events $\{\mathcal{E}_1,\ldots,\mathcal{E}_{r-1}\}$, Equation (23) yields

$$\begin{aligned}
(23)&=\prod_{r=1}^{R}\left(1-\mathbb{P}\left(Z>\sqrt{r\left\lfloor\frac{\eta_2}{2\phi^2}\right\rfloor}\frac{\delta}{\sigma_{upper}}\right)\right)\\
&\geq\prod_{r=1}^{\infty}\left(1-\exp\left(-\frac{\lfloor\eta_2/(2\phi^2)\rfloor\delta^2}{2\sigma_{upper}^2}r\right)\right)\\
&\geq\exp\left(-\frac{\pi^2\sigma_{upper}^2}{3\lfloor\eta_2/(2\phi^2)\rfloor\delta^2}\right)\\
&\geq\exp\left(-\frac{4\phi^2\pi^2\sigma_{upper}^2}{3\eta_2\delta^2}\right),
\end{aligned}\tag{24}$$

where the first inequality holds due to Lemma 1, and the second last inequality holds due to Lemma 2.

For processor $s=1,2,\ldots,m$, $N'_{R+1}=\left\lfloor\frac{R+1}{\phi-1}\left(\frac{\phi-1}{\phi}\right)^{R+1}\frac{N}{m}\right\rfloor$ observations are generated for the local best alternative in round $R+1$. Let $\bar{X}_i^{R+1}$ be the sample mean of alternative $i\in\mathcal{I}_{final}$, calculated based on these N'_{R+1} observations. Then, we can write the third term in Equation (21) as follow,

$$\mathbb{P}(\mathcal{E}_{R+1}|\mathcal{E}_1,\ldots,\mathcal{E}_R)=\mathbb{P}\left(\bar{X}_k^{R+1}>\max_{i\in\mathcal{I}_{final}\setminus\{k\}}\bar{X}_i^{R+1}\middle|\mathcal{E}_1,\ldots,\mathcal{E}_R\right)$$

$$
\begin{aligned}
&\geq 1-\mathbb{P}\left(\bigcup_{i\in\mathcal{I}_{final}\setminus\{k\}}\left\{\bar{X}_k^{R+1}<\bar{X}_i^{R+1}\right\}\middle|\mathcal{E}_1,\ldots,\mathcal{E}_R\right)\\
&=1-\mathbb{P}\left(\bigcup_{i\in\mathcal{I}_{final}\setminus\{k\}}\left\{\frac{\bar{X}_k^{R+1}-\bar{X}_i^{R+1}-\mu_k+\mu_i}{\sigma_{ki}/\sqrt{N'_{R+1}}}<\sqrt{N'_{R+1}}\frac{-\mu_k+\mu_i}{\sigma_{ki}}\right\}\middle|\mathcal{E}_1,\ldots,\mathcal{E}_R\right)\\
&\geq 1-\mathbb{P}\left(\bigcup_{i\in\mathcal{I}_{final}\setminus\{k\}}\left\{\frac{\bar{X}_k^{R+1}-\bar{X}_i^{R+1}-\mu_k+\mu_i}{\sigma_{ki}/\sqrt{N'_{R+1}}}<\sqrt{N'_{R+1}}\frac{-\mu_k+\mu_i}{\sigma_{ki}}\right\}\middle|\mathcal{E}_1,\ldots,\mathcal{E}_R\right)\\
&\geq 1-\mathbb{P}\left(\bigcup_{i\in\mathcal{I}_{final}\setminus\{k\}}\left\{\frac{\bar{X}_k^{R+1}-\bar{X}_i^{R+1}-\mu_k+\mu_i}{\sigma_{ki}/\sqrt{N'_{R+1}}}<\sqrt{N'_{R+1}}\frac{-\delta}{\sigma_{upper}}\right\}\middle|\mathcal{E}_1,\ldots,\mathcal{E}_R\right)\\
&\geq 1-(m-1)\,\mathbb{P}\left(Z<\sqrt{N'_{R+1}}\frac{-\delta}{\sigma_{upper}}\middle|\mathcal{E}_1,\ldots,\mathcal{E}_R\right)\\
&=1-(m-1)\,\mathbb{P}\left(Z<\sqrt{N'_{R+1}}\frac{-\delta}{\sigma_{upper}}\right)\\
&\geq 1-(m-1)\exp\left(-\frac{N'_{R+1}\delta^2}{2\sigma_{upper}^2}\right),
\end{aligned}
\tag{25}
$$

where the last equality holds because event $\{Z<-\delta\sqrt{N'_{R+1}}/\sigma_{upper}\}$ is independent of events $\{\mathcal{E}_1,\ldots,\mathcal{E}_R\}$ and the last inequality holds due to Lemma 3. For N'_{R+1}, we have,

$$
\begin{aligned}
N'_{R+1}&=\left\lfloor\frac{R+1}{\phi-1}\left(\frac{\phi-1}{\phi}\right)^{R+1}\frac{N}{m}\right\rfloor\\
&\geq\left\lfloor\frac{\left(\log_2\frac{k}{m}+1\right)\eta_2 k}{m\,(\phi-1)}\left(\frac{\phi-1}{\phi}\right)^{\log_2 k/m+2}\right\rfloor\\
&=\left\lfloor\frac{\left(\log_2\frac{k}{m}+1\right)\eta_2 k}{m\,(\phi-1)}\left(\frac{2\,(\phi-1)}{\phi}\right)^{\log_2 k/m+2}\frac{m}{4k}\right\rfloor\\
&\geq\frac{\left(\log_2\frac{k}{m}+1\right)\eta_2}{8\,(\phi-1)}\left(\frac{2\,(\phi-1)}{\phi}\right)^{\log_2 k/m+2}.
\end{aligned}
\tag{26}
$$

Plugging the results in Equation (26) into Equation (25) yields,

$$
\begin{aligned}
\mathbb{P}\left(\mathcal{E}_{R+1}|\mathcal{E}_1,\ldots,\mathcal{E}_R\right)&\geq 1-(m-1)\exp\left(-\frac{N'_{R+1}\delta^2}{2\sigma_{upper}^2}\right)\\
&\geq 1-(m-1)\exp\left(-\frac{\left(\log_2\frac{k}{m}+1\right)\eta_2\delta^2}{8\sigma_{upper}^2\phi}\left(\frac{2\,(\phi-1)}{\phi}\right)^{\log_2 k/m+1}\right).
\end{aligned}
\tag{27}
$$

Therefore, with the results listed in Equations (24) and (27), we can rewrite Equation (21) as,

$$
\begin{aligned}
\mathrm{PCS}&=\mathbb{P}\left(\mathcal{E}_1\right)\prod_{r=2}^{R}\mathbb{P}\left(\mathcal{E}_r|\mathcal{E}_1,\ldots,\mathcal{E}_{r-1}\right)\mathbb{P}\left(\mathcal{E}_{R+1}|\mathcal{E}_1,\ldots,\mathcal{E}_R\right)\\
&\geq\exp\left(-\frac{4\phi^2\pi^2\sigma_{upper}^2}{3\eta_2\delta^2}\right)\left(1-(m-1)\exp\left(-\frac{\left(\log_2\frac{k}{m}+1\right)\eta_2\delta^2}{8\sigma_{upper}^2\phi}\left(\frac{2\,(\phi-1)}{\phi}\right)^{\log_2 k/m+1}\right)\right).
\end{aligned}
$$

It concludes the proof of part (1).

Proof of Part (2) Given that Assumptions 3 holds, we aim to show that for any positive constant $\eta_2' \geq 2\phi^2$ and IZ parameter δ, if $\phi \geq 3$ and $N/k \geq \eta_2'$, the PGS of the $\mathcal{FBKT}^{\mathcal{S}+}$ procedure can be lower bounded. To prove this, we first let $\hat{k}' = 2^R$, and

$$\delta_r = \frac{\sqrt{2(\phi-1)} - \sqrt{\phi}}{\sqrt{\phi}} \left(\frac{\phi}{2(\phi-1)}\right)^{\frac{r}{2}} \delta,$$

for $r \geq 1$. Because $\phi \geq 3$, $\delta_r > 0$ for $r \geq 1$ and $\sum_{r=1}^{\infty} \delta_r = \delta$. Notice that, each processor can identify the local best alternative in R rounds and the procedure selects the final output in round $R+1$. For $1 \leq r \leq R$, we let $\rho_r = \arg\max_{i \in \cup_{s=1}^m \mathcal{I}_s^r} \mu_i$, and define $\mathfrak{s}_r$ as the processor that contains alternative ρ_r. For $r = R+1$, we let $\rho_r = \arg\max_{i \in \mathcal{I}_{final}} \mu_i$. To ease the notation, we let $\rho_r = \varphi$ for $r = R+2$, where φ is the index of the alternative which is finally selected by the procedure. For $1 \leq r \leq R$, we further define event $\mathcal{E}_r' = \{\mu_{\rho_r} - \mu_{\rho_r'} \leq \delta_r\} \cup \{\mu_{\rho_r} - \mu_{\rho_r'} > \delta_r \text{ and } \bar{X}_{\rho_r}^r \geq \bar{X}_{\rho_r'}^r\}$, where ρ_r' is the index of the alternative that competes with alternative ρ_r in processor $\mathfrak{s}_r$. For $r = R+1$, we define event $\mathcal{E}_r' = \cap_{i \in \mathcal{I}_{final} \setminus \{\rho_r\}} \left\{\{\mu_{\rho_r} - \mu_i \leq \delta_r\} \cup \{\mu_{\rho_r} - \mu_i > \delta_r \text{ and } \bar{X}_{\rho_r}^r \geq \bar{X}_i^r\}\right\}$. By letting $\mathcal{E}_r'^c$ be the complement of event $\mathcal{E}_r'$ for $1 \leq r \leq R+1$, we can bound the PGS of the $\mathcal{FBKT}^{\mathcal{S}+}$ procedure as follows,

$$\text{PGS} \geq \mathbb{P}\left(\bigcap_{r=1}^{R+1} \{\mu_{\rho_r} - \mu_{\rho_{r+1}} \leq \delta_r\}\right) \geq \mathbb{P}\left(\bigcap_{r=1}^{R+1} \{\mathcal{E}_r'\}\right) = \mathbb{P}(\mathcal{E}_1') \prod_{r=2}^{R} \mathbb{P}\left(\mathcal{E}_r' \middle| \mathcal{E}_1', \ldots, \mathcal{E}_{r-1}'\right) \mathbb{P}\left(\mathcal{E}_{R+1}' \middle| \mathcal{E}_1', \ldots, \mathcal{E}_R'\right), \tag{28}$$

where the first inequality holds because as long as the mean difference between alternatives ρ_r and ρ_{r+1} is less than δ_r for $1 \leq r \leq R+1$ the mean difference between alternatives k and φ is less than $\sum_{r=1}^{\infty} \delta_r = \delta$, and the second inequality holds because event $\mathcal{E}_r'$ implies event $\{\mu_{\rho_r} - \mu_{\rho_{r+1}} \leq \delta_r\}$ for $1 \leq r \leq R+1$. Notice that following the same arguments used in the analysis in Equation (22), we can conclude that

$$\mathcal{N}_r^{\mathfrak{s}_r} \geq r \left\lfloor \frac{\eta_2'}{4\phi(\phi-1)} \left(\frac{2(\phi-1)}{\phi}\right)^r \right\rfloor. \tag{29}$$

Because $\eta_2' \geq 2\phi^2$, $\mathcal{N}_r^{\mathfrak{s}_r}$ is a positive integer no less than r. Then, based on the results stated in Equation (29), we can rewrite the first two terms in Equation (28) as follows,

$$\begin{aligned}
&\mathbb{P}(\mathcal{E}_1') \prod_{r=2}^{R} \mathbb{P}\left(\mathcal{E}_r' \middle| \mathcal{E}_1', \ldots, \mathcal{E}_{r-1}'\right) \\
&= \left(1 - \mathbb{P}\left(\mathcal{E}_1'^c\right)\right) \prod_{r=2}^{R} \left(1 - \mathbb{P}\left(\mathcal{E}_r'^c \middle| \mathcal{E}_1', \ldots, \mathcal{E}_{r-1}'\right)\right) \\
&= \left(1 - \mathbb{P}\left(\mu_{\rho_1} - \mu_{\rho_1'} > \delta_1 \text{ and } \bar{X}_{\rho_1}^1 < \bar{X}_{\rho_1'}^1\right)\right) \times \\
&\prod_{r=2}^{R} \left(1 - \mathbb{P}\left(\mu_{\rho_r} - \mu_{\rho_r'} > \delta_r \text{ and } \bar{X}_{\rho_r}^r < \bar{X}_{\rho_r'}^r \middle| \mathcal{E}_1', \ldots, \mathcal{E}_{r-1}'\right)\right)
\end{aligned}$$

$$= \left(1-\mathbb{P}\left(\mu_{\rho_1}-\mu_{\rho_1'}>\delta_1 \text{ and } \frac{-\bar{X}^1_{\rho_1}+\bar{X}^1_{\rho_1'}+\mu_{\rho_1}-\mu_{\rho_1'}}{\sigma_{\rho_1\rho_1'}/\sqrt{\mathcal{N}_1^{\mathfrak{s}_1}}} > \sqrt{\mathcal{N}_1^{\mathfrak{s}_1}}\frac{\mu_{\rho_1}-\mu_{\rho_1'}}{\sigma_{\rho_1\rho_1'}}\right)\right)\times$$
$$\prod_{r=2}^{R}\left(1-\mathbb{P}\left(\mu_{\rho_r}-\mu_{\rho_r'}>\delta_r \text{ and } \frac{-\bar{X}^r_{\rho_r}+\bar{X}^r_{\rho_r'}+\mu_{\rho_r}-\mu_{\rho_r'}}{\sigma_{\rho_r\rho_r'}/\sqrt{\mathcal{N}_r^{\mathfrak{s}_r}}} > \sqrt{\mathcal{N}_r^{\mathfrak{s}_r}}\frac{\mu_{\rho_r}-\mu_{\rho_r'}}{\sigma_{\rho_r\rho_r'}}\middle|\mathcal{E}_1',\ldots,\mathcal{E}_{r-1}'\right)\right)$$
$$\geq \left(1-\mathbb{P}\left(\mu_{\rho_1}-\mu_{\rho_1'}>\delta_1 \text{ and } Z>\sqrt{\left\lfloor\frac{\eta_2'}{2\phi^2}\right\rfloor}\frac{\delta_1}{\sigma_{upper}}\right)\right)\times$$
$$\prod_{r=2}^{R}\left(1-\mathbb{P}\left(\mu_{\rho_r}-\mu_{\rho_r'}>\delta_r \text{ and } Z>\sqrt{r\left\lfloor\frac{\eta_2'}{4\phi(\phi-1)}\left(\frac{2(\phi-1)}{\phi}\right)^r\right\rfloor}\frac{\delta_r}{\sigma_{upper}}\middle|\mathcal{E}_1',\ldots,\mathcal{E}_{r-1}'\right)\right)$$
$$\geq \left(1-\mathbb{P}\left(Z>\sqrt{\left\lfloor\frac{\eta_2'}{2\phi^2}\right\rfloor}\frac{\delta_1}{\sigma_{upper}}\right)\right)\prod_{r=2}^{R}\left(1-\mathbb{P}\left(Z>\sqrt{r\left\lfloor\frac{\eta_2'}{4\phi(\phi-1)}\left(\frac{2(\phi-1)}{\phi}\right)^r\right\rfloor}\frac{\delta_r}{\sigma_{upper}}\middle|\mathcal{E}_1',\ldots,\mathcal{E}_{r-1}'\right)\right)$$
$$=\prod_{r=1}^{R}\left(1-\mathbb{P}\left(Z>\sqrt{r\left\lfloor\frac{\eta_2'}{4\phi(\phi-1)}\left(\frac{2(\phi-1)}{\phi}\right)^r\right\rfloor}\frac{\delta_r}{\sigma_{upper}}\middle|\mathcal{E}_1',\ldots,\mathcal{E}_{r-1}'\right)\right), \tag{30}$$

where the last inequality holds because event $\{Z>\sqrt{r\lfloor\eta_2'/(4\phi(\phi-1))(2(\phi-1)/\phi)^r\rfloor}\delta_r/\sigma_{upper}\}$ is independent of events $\{\mathcal{E}_1',\ldots,\mathcal{E}_{r-1}'\}$. Then, applying Lemmas 2 and 3 to Equation (30), we have,

$$\begin{aligned}
(30) &\geq \prod_{r=1}^{R}\left(1-\exp\left(-r\left\lfloor\frac{\eta_2'}{4\phi(\phi-1)}\left(\frac{2(\phi-1)}{\phi}\right)^r\right\rfloor\frac{\delta_r^2}{2\sigma_{upper}^2}\right)\right)\\
&=\prod_{r=1}^{\infty}\left(1-\exp\left(-r\left\lfloor\frac{\eta_2'}{4\phi(\phi-1)}\left(\frac{2(\phi-1)}{\phi}\right)^r\right\rfloor\frac{\left(\frac{\sqrt{2(\phi-1)}-\sqrt{\phi}}{\sqrt{\phi}}\right)^2\left(\frac{\phi}{2(\phi-1)}\right)^r\delta^2}{2\sigma_{upper}^2}\right)\right)\\
&\geq\prod_{r=1}^{\log_2\hat{k}}\left(1-\exp\left(-r\frac{\eta_2'}{8\phi(\phi-1)}\left(\frac{2(\phi-1)}{\phi}\right)^r\frac{\left(\frac{\sqrt{2(\phi-1)}-\sqrt{\phi}}{\sqrt{\phi}}\right)^2\left(\frac{\phi}{2(\phi-1)}\right)^r\delta^2}{2\sigma_{upper}^2}\right)\right)\\
&=\prod_{r=1}^{\log_2\hat{k}}\left(1-\exp\left(-r\frac{\eta_2'\delta^2}{8\sigma_{upper}^2}\left(\frac{1}{\phi}-\frac{1}{\sqrt{2\phi(\phi-1)}}\right)^2\right)\right)\\
&\geq\exp\left(-\frac{8\pi^2\sigma_{upper}^2\phi^3(\phi-1)}{3\eta_2'\delta^2\left(\sqrt{2\phi(\phi-1)}-\phi\right)^2}\right),
\end{aligned} \tag{31}$$

where the first inequality holds due to Lemma 3, the second inequality holds because if $a\geq 1$, then $\lfloor a\rfloor\geq a/2$, and the third inequality holds due to Lemma 2.

For processor $s=1,2,\ldots,m$, $N_{R+1}'=\left\lfloor\frac{R+1}{\phi-1}\left(\frac{\phi-1}{\phi}\right)^{R+1}\frac{N}{m}\right\rfloor$ observations are generated for the local best alternative in round $R+1$. Let $\bar{X}_i^{R+1}$ be the sample mean of alternative $i\in\mathcal{I}_{final}$ calculated based on these N_{R+1}' observations. Then, by letting $\mathcal{I}_{final}'=\mathcal{I}_{final}\backslash\{\rho_{R+1}\}$, we can write the third term in Equation (28) as,

$$\mathbb{P}\left(\mathcal{E}_{R+1}'\middle|\mathcal{E}_1',\ldots,\mathcal{E}_R'\right)$$

$$
\begin{aligned}
&= \mathbb{P}\left(\bigcap_{i\in\mathcal{I}'_{final}} \left\{\{\mu_{\rho_{R+1}}-\mu_i \le \delta_{R+1}\} \cup \{\mu_{\rho_{R+1}}-\mu_i > \delta_{R+1} \text{ and } \bar{X}^{R+1}_{\rho_{R+1}} \ge \bar{X}^{R+1}_i\}\right\} \middle| \mathcal{E}'_1,\ldots,\mathcal{E}'_R\right)\\
&= 1-\mathbb{P}\left(\bigcup_{i\in\mathcal{I}'_{final}} \left\{\mu_{\rho_{R+1}}-\mu_i > \delta_{R+1} \text{ and } \bar{X}^{R+1}_{\rho_{R+1}} < \bar{X}^{R+1}_i\right\} \middle| \mathcal{E}'_1,\ldots,\mathcal{E}'_R\right)\\
&= 1-\mathbb{P}\left(\bigcup_{i\in\mathcal{I}'_{final}} \left\{\mu_{\rho_{R+1}}-\mu_i > \delta_{R+1} \text{ and } \frac{-\bar{X}^{R+1}_{\rho_{R+1}}+\bar{X}^{R+1}_i+\mu_{\rho_{R+1}}-\mu_i}{\sigma_{\rho_{R+1}i}/\sqrt{N'_{R+1}}} > \sqrt{N'_{R+1}}\frac{\mu_{\rho_{R+1}}-\mu_i}{\sigma_{\rho_{R+1}i}}\right\} \middle| \mathcal{E}'_1,\ldots,\mathcal{E}'_R\right)\\
&\ge 1-\mathbb{P}\left(\bigcup_{i\in\mathcal{I}'_{final}} \left\{\mu_{\rho_{R+1}}-\mu_i > \delta_{R+1} \text{ and } \frac{-\bar{X}^{R+1}_{\rho_{R+1}}+\bar{X}^{R+1}_i+\mu_{\rho_{R+1}}-\mu_i}{\sigma_{\rho_{R+1}i}/\sqrt{N'_{R+1}}} > \sqrt{N'_{R+1}}\frac{\delta_{R+1}}{\sigma_{upper}}\right\} \middle| \mathcal{E}'_1,\ldots,\mathcal{E}'_R\right)\\
&\ge 1-\mathbb{P}\left(\bigcup_{i\in\mathcal{I}'_{final}} \left\{\frac{-\bar{X}^{R+1}_{\rho_{R+1}}+\bar{X}^{R+1}_i+\mu_{\rho_{R+1}}-\mu_i}{\sigma_{\rho_{R+1}i}/\sqrt{N'_{R+1}}} > \sqrt{N'_{R+1}}\frac{\delta_{R+1}}{\sigma_{upper}}\right\} \middle| \mathcal{E}'_1,\ldots,\mathcal{E}'_R\right)\\
&\ge 1-(m-1)\,\mathbb{P}\left(Z > \sqrt{N'_{R+1}}\frac{\delta_{R+1}}{\sigma_{upper}} \middle| \mathcal{E}'_1,\ldots,\mathcal{E}'_R\right)\\
&= 1-(m-1)\,\mathbb{P}\left(Z > \sqrt{N'_{R+1}}\frac{\delta_{R+1}}{\sigma_{upper}}\right)\\
&\ge 1-(m-1)\exp\left(-\frac{N'_{R+1}\delta^2_{R+1}}{2\sigma^2_{upper}}\right),
\end{aligned}
\tag{32}
$$

where the second equality holds due to De Morgan's Law, and the last inequality holds due to Lemma 3. According the results listed in Equation (26), for N'_{R+1}, we have,

$$
N'_{R+1} \ge \frac{\left(\log_2\frac{k}{m}+1\right)\eta'_2}{8(\phi-1)}\left(\frac{2(\phi-1)}{\phi}\right)^{\log_2 k/m+2}. \tag{33}
$$

Plugging the results in Equation (33) into Equation (32), we have

$$
\begin{aligned}
&\mathbb{P}\left(\mathcal{E}'_{R+1}\middle|\mathcal{E}'_1,\ldots,\mathcal{E}'_R\right)\\
&\ge 1-(m-1)\exp\left(-\frac{N'_{R+1}\delta^2_{R+1}}{2\sigma^2_{upper}}\right)\\
&\ge 1-(m-1)\exp\left(-\frac{\frac{\left(\log_2\frac{k}{m}+1\right)\eta'_2}{8(\phi-1)}\left(\frac{2(\phi-1)}{\phi}\right)^{\log_2 k/m+2}\left(\frac{\sqrt{2(\phi-1)}-\sqrt{\phi}}{\sqrt{\phi}}\right)^2\left(\frac{\phi}{2(\phi-1)}\right)^{\log_2 k/m+2}\delta^2}{2\sigma^2_{upper}}\right)\\
&= 1-(m-1)\exp\left(-\frac{\left(\log_2\frac{k}{m}+1\right)\eta'_2\left(3\phi-2+2\sqrt{2\phi(\phi-1)}\right)}{16\sigma^2_{upper}\phi(\phi-1)}\delta^2\right).
\end{aligned}
\tag{34}
$$

Therefore, with the results listed in Equations (31) and (34), we can rewrite Equation (28) as

$$
\mathrm{PGS} \ge \mathbb{P}\left(\mathcal{E}'_1\right)\prod_{r=2}^{R}\mathbb{P}\left(\mathcal{E}'_r\middle|\mathcal{E}'_1,\ldots,\mathcal{E}'_{r-1}\right)\mathbb{P}\left(\mathcal{E}'_{R+1}\middle|\mathcal{E}'_1,\ldots,\mathcal{E}'_R\right)
$$

$$\geq \exp\left(-\frac{8\pi^2\sigma_{upper}^2\phi^3\left(\phi-1\right)}{3\eta_2'\delta^2\left(\sqrt{2\phi\left(\phi-1\right)}-\phi\right)^2}\right)\left(1-\left(m-1\right)\exp\left(-\frac{\left(\log_2\frac{k}{m}+1\right)\eta_2'\left(3\phi-2+2\sqrt{2\phi\left(\phi-1\right)}\right)}{16\sigma_{upper}^2\phi\left(\phi-1\right)}\delta^2\right)\right).$$

It concludes the proof of part (2). $\square$