METHODS

Practical Simulation Budget Allocation for Ranked Subset Partitioning

MOON GI SEOK [1], (Member, IEEE), AND SEON HAN CHOI [2,3], (Member, IEEE)
[1]Department of Computer Science and Engineering, Dongguk University, Seoul 04620, South Korea
[2]Department of Electronic and Electrical Engineering, Ewha Womans University, Seoul 03760, South Korea
[3]Graduate Program in Smart Factory, Ewha Womans University, Seoul 03760, South Korea

Corresponding author: Seon Han Choi (seonhan.choi@ewha.ac.kr)

ABSTRACT Stochastic simulation is a powerful tool for analyzing complex discrete-event dynamic systems; however, it does not exhibit sufficient efficiency because of the requirement of numerous replicated simulations for obtaining accurate analysis results. Ranking and selection (R&S) efficiently allocates a simulation budget using ordinal optimization to correctly select alternatives of interest. Existing R&S methods focus on selecting an optimal alternative or a subset of optimal alternatives. Based on a generalization of this methodology, we propose an R&S method for partitioning k alternatives into n $(2 \leq n \leq k)$ exclusive ranked subsets, which is effective for job distribution and web search applications. The proposed method evaluates if the observed simulation results for each alternative have sufficient precision to correctly distinguish between the ranked subsets. It sequentially allocates a small portion of the budget based on the evaluation results, gradually improving the precision to maximize the efficiency. The superior efficiency of the proposed method compared with that of the existing methods is demonstrated using various numerical experiments. Furthermore, a practical problem that involves relocation-zone distribution in bicycle-sharing systems demonstrates that the proposed method can be effectively applied in situations requiring high simulation efficiencies, such as digital twins in complex systems.

INDEX TERMS Discreteevent dynamic system, stochastic simulation, ranking and selection, ranked subset.

I. INTRODUCTION

Stochastic simulation is a powerful tool for analyzing complex discreteevent dynamic systems (DEDSs) such as transportation [1], telecommunication [2], military [3], mining [4], and cloud services [5], which rarely satisfy the strict assumptions required by analytic models [6]. However, in case of stochastic simulation numerous simulation replications are required to obtain accurate analysis results, which is a major concern associated with its application [7] There are a finite number (k) of alternatives, and each alternative is evaluated using stochastic simulation. If our analysis goal is to

determine the optimal alternatives without obtaining their exact simulation results, ranking and selection (R&S) is an effective solution [8]

Using ordinal optimization [9], R&S efficiently allocates a simulation budget (i.e., the number of simulation replications that can be conducted in a given computing environment) to k alternatives for correctly identifying their relative orders, thereby selecting the optimal alternatives Thus far various allocation methods in R&S have been developed, most of which are variants of three basic methods [10]: the indifferencezone [11], optimal computing budget allocation (OCBA) [12], and expected value information methods [13]. The differences among these methods lie in the way they define the statistical evidence for correct selection using relative

The associate editor coordinating the review of this manuscript and approving it for publication was Qi Zhou.

TABLE 1. Existing R&S studies classified according to the number of ranked subsets.

Number of ranked subsets	Analysis goal	Existing R&S studies[a]
2	Optimal alternative selection	[11]-[13], [16]
	Optimal subset (top-*m*) selection	[17], [20]-[22]
3	Best and worst alternative selection	[23], [24]
	Best and worst subset selection	[25], [26]
k	Complete ranking identification	[10], [27]

[a]The listed studies do not constitute an exhaustive list.

orders and the way simulation replications are allocated to strengthen this evidence. Wellorganized summaries of these methods and their variants can be found in [14] and [15]. Furthermore, an uncertainty evaluation (UE) method [16] based on statistical hypothesis testing has recently been proposed, and several practical variants [17], [18], [19] have been developed because of its high robustness to noise.

Most existing studies have focused on selecting one optimal alternative or the top-*m* alternatives (i.e., an optimal subset) [20], [21], [22]. From the perspective of partitioning, these can be regarded as dividing *k* alternatives into two exclusive ranked subsets; in this context, the subsets are ranked such that the exact simulation results for all alternatives in the higher-ranked subset are better than those for the lower-ranked subset. Some methods partition a set of alternatives into three ranked subsets by simultaneously selecting the best and worst alternatives [23], [24] or subsets [25], [26]. Notably, a few studies on complete ranking identification [10], [27] present extreme cases in which *k* alternatives are divided into *k* subsets. Table 1 summarizes the existing studies on R&S according to the number of partitioned subsets.

In practice, partitioning *k* alternatives into *n* ranked subsets is often required where $3 < n < k$ In particular, this may occur when jobs are distributed to *n* workers. For example, consider a factory with five production lines. Each line can produce the same product; however, each line has a different defect rate owing to obsolescence. Ten production plans that specify the products to be produced and their quantities are provided; the required production times are similar. The effects of defects on each plan can be analyzed using stochastic simulation [28]. If these ten plans are partitioned into five ranked subsets where each subset comprises two plans according to the analysis results, and the five subsets are distributed to the five lines according to their defect rate (i.e., the plans most affected by the defects are placed on the line with the lowest defect rate), the production efficiency can be improved.

Another example is the problem of relocationzone distribution in bicycle-sharing systems Consider that there are eight relocation zones and that the amount of bicycle relocation work for each zone varies. Four relocation trucks having different bicycleloading capacities are employed. The amount of work required for each zone can be estimated using stochastic simulation [29]. If the eight zones are partitioned into four ranked subsets each comprising two zones according to the

estimation results, and these four subsets are distributed to the four trucks according to their capacity (i.e., the zones with the highest workload are assigned to the trucks with the largest bicycle-loading capacity), the relocation efficiency can be maximized. Although complete ranking identification methods can be applied in both examples, they would be inefficient because the number of subsets to be partitioned is less than *k* and the ranking within each subset is not required to be identified Ranked subset partitioning is a notable challenge in several web search applications [3], [31]. In addition, it is critical in feature extraction, construction, and selection tasks in machine learning and pattern recognition [32]

Despite this need, few studies on R&S have attempted to partition *k* alternatives into generalized *n* ranked subsets. Notably, Zhang et al. [32] proposed OCBA for subset ranking (OCBAsr) based on OCBA. However, OCBAsr needs to heuristically determine constants that can distinguish between the ranked subsets during allocation. Moreover, the performance of OCBAsr is highly sensitive to these constants. Although OCBAsr is more efficient than the equal allocation method, which allocates the same budget to all alternatives, its simulation efficiency can be further improved. When $n = 2$, OCBAsr reduces to OCBA for selecting the optimal subset (OCBAm) [20]. However, OCBAm exhibits lower efficiency than some recently proposed methods, such as OCBAm+ [21], EOC-m [22], and UEm [17]. Recently, with the increasing complexity of systems, the cost per simulation replication has increased; thus, efficiency has become more important in the application of simulations [10]. In particular, digital twins, which are simulation models synchronized with real systems, require high efficiencies for real-time system control [33]. Therefore, developing a method that is more efficient than OCBAsr is essential

This study proposes an R&S method for partitioning *k* alternatives into *n* ranked subsets. The number of alternatives in each subset may differ. Unlike OCBAsr, the proposed method employed UE owing to its suitability for complex practical problems because of its high robustness to noise [16] Notably, UEm—a method based on UE for selecting an optimal subset—exhibits superior efficiency over OCBAm, implying that UE-based approach is adequate for developing a more efficient method than OCBAsr. The proposed method aims to maximize the probability of correctly selecting *n* subsets ($P\{CS_n\}$) within a limited number of simulation replications. We define the uncertainty using UE to evaluate if the observed simulation results for each alternative are precise enough to accurately distinguish between the ranked subsets. The proposed method splits the given budget into a few simulation replications, sequentially allocates these replications based on the uncertainty, and gradually improves insufficient precision to maximize $P\{CS_n\}$ Our experimental results on both the numerical and practical problems demonstrate the high efficiency and utility of the proposed method.

The remainder of this paper is organized as follows. In Section II, the problem is formulated. Our ranked subset partitioning method is proposed in Section III. The

Notation	Description
B	Number of simulation replications (i.e., simulation budget).
k	Number of alternatives.
x_i	Alternative (i.e., simulation input), where $i \in \{1, \dots, k\}$ identifies each alternative.
Θ	Given set of alternatives: $\Theta = \{x_1, \dots, x_k\}$.
n	Number of ranked subsets to be partitioned; $2 \le n \le k$.
m_u	Number of alternatives in a subset with ranking u, where $u \in \{1, \dots, n\}$ and $\sum_{u=1}^{n} m_u = k$.
Y_{ij}	Stochastic simulation output of x_i in j-th replication, where Y_{ij} follows a normal distribution: $Y_{ij} \sim \mathcal{N}(\mu_i, \sigma_i^2)$.
b_i	Number of simulation replications allocated on x_i.
$\hat{\mu}_i$	Sample mean of $Y_{i1}, \dots, Y_{ib_i}$: $\hat{\mu}_i = 1/b_i \cdot \sum_{j=1}^{b_i} Y_{ij} \sim \mathcal{N}(\mu_i, \sigma_i^2/b_i)$.
μ_i	Mean of Y_{ij} (i.e., exact simulation result of x_i): $\mu_i = E(Y_{ij}) = \lim_{b_i \to \infty} \hat{\mu}_i$.
s_i^2	Sample variance of $Y_{i1}, \dots, Y_{ib_i}$: $s_i^2 = \sum_{j=1}^{b_i} (Y_{ij} - \hat{\mu}_i)^2/(b_i - 1)$.
r_i	Actual relative ranking of x_i within Θ based on μ_i (e.g., when $\Theta = \{x_1, x_2, x_3\}$ and $\mu_3 < \mu_1 < \mu_2$, then $r_1 = 2$, $r_2 = 3$, and $r_3 = 1$).
$\hat{r}_i$	Observed relative ranking of x_i within Θ based on $\hat{\mu}_i$.

experimental results are presented in Section IV, and the conclusions are presented in Section V

II. PROBLEM FORMULATION

The basic notations used in this study are summarized in Table 2.

As is common in the literature on R&S, it is assumed that the stochastic simulation output follows a normal distribution and is independent across alternatives and simulation replications. This is reasonable in practice because the output of a DEDS simulation is typically determined as an average value or a batch mean; therefore, the central limit theorem holds [8]. To develop a practical method, we assume no prior information about the output distribution before performing the simulation. Additionally, to simplify the problem, we assume that no two alternatives exhibit identical exact simulation results ($\mu_i \neq \mu_j, ij \in \Theta$, and $i \neq j$). The case of two alternatives exhibiting identical results is unusual in practice, and without considering tolerance, an infinite number of simulation replications would be needed to confirm the equality, which is impossible.

Our goal is to partition k alternatives into n ranked subsets: $S_1, S_2 \dots, S_n$ When low values are desired for the simulation results, the uth ranked subset S_u which comprises m_u alternatives is defined as follows:

$$S_u = \left\{ x_i \in \Theta \left| \sum_{j=1}^{u-1} m_j < r_i \le \sum_{j=1}^{u} m_j \right. \right\}. \quad (1)$$

Here, S_u comprises alternatives for which the actual relative rankings within Θ based on their exact simulation results are between $\sum_{j=1}^{u-1} m_j + 1$ and $\sum_{j=1}^{u-1} m_j + m_u$. According to the definition presented in (1) the n ranked subsets are mutually exclusive ($S_1 \cap \dots \cap S_n = \phi$), and their union ($\bigcup_{u=1}^{n} S_u$) is equal to Θ ($\sum_{u=1}^{n} m_u = k$). The exact simulation results of the alternatives in a ranked subset are better than those of all

the alternatives in the lower-ranked subsets ($x_i \in S_u : \mu_i < \forall \mu_j, x_j \in S_{u+1} \cup \dots \cup S_n$).

The actual relative rankings of alternatives are essential to distinguish between the n ranked subsets; however, no prior information about them is provided. In practice, after simulation replications are performed k alternatives are partitioned into estimated n ranked subsets $\hat{S}_1 \dots, \hat{S}_n$ based on their observed relative rankings. Similar to $S_u \hat{S}_u$, which comprises m_u alternatives is defined as follows:

$$\hat{S}_u = \left\{ x_i \in \Theta \left| \sum_{j=1}^{u-1} m_j < \hat{r}_i \le \sum_{j=1}^{u} m_j \right. \right\}. \quad (2)$$

Because of the error in estimating the sample mean, the observed rankings may differ from the actual rankings; thus, $\hat{S}_u$ and S_u may differ However, the sets are always of the same size ($\left\| \hat{S}_u \right\| = \|S_u\| = m_u$). If the simulation budget is sufficient and many replications can be conducted for every alternative such that its observed ranking is accurate, n ranked subsets can be correctly selected. However, our objective is to increase the simulation efficiency, which means that the limited number of replications must be appropriately allocated to k alternatives, such that $\hat{S}_1 = S_1$ $\hat{S}_2 = S_2 \dots, \hat{S}_n = S_n$. Assuming that the simulation cost is the same for every replication across all alternatives, this objective can be formulated as follows:

$$\arg\max_{b_1, \dots, b_k} \Pr \left\{ \bigcap_{u=1}^{n} \hat{S}_u = S_u \right\} \quad \text{s.t.} \sum_{i=1}^{k} b_i = B \text{ and } b_i > 0. \quad (3)$$

Here, $\Pr \left\{ \bigcap_{u=1}^{n} \hat{S}_u = S_u \right\}$ indicates $P\{CS_n\}$ In the next section, we propose an efficient R&S method to solve this problem.

III. PROPOSED METHOD

The proposed method, like other R&S methods, divides a given number of total simulation replications B into small units and sequentially allocates the units to efficiently utilize B [8]. In the sequential procedure, a small unit of B is allocated for the alternatives using the observed simulation results. To maximize $P\{CS_n\}$, it is necessary to allocate this small unit to the alternatives that most require further simulation. The proposed method evaluates the uncertainty of each alternative using statistical hypothesis testing and p-value and allocates additional replications accordingly.

A. UNCERTAINTY EVALUATION

The actual relative rankings of the alternatives that are required to select n ranked subsets are determined using the relations between the exact simulation results. If the observed relations based on the sample means are verified to be the same as the actual relations, n ranked subsets can be accurately selected. Statistical hypothesis testing can then be used for the verification.

For example, for two alternatives x_i and x_j, $\hat{\mu}_i$ and s_i^2 and $\hat{\mu}_j$ and s_j^2 are obtained after b_i and b_j replications, respectively. When the observed relation is $\hat{\mu}_i < \hat{\mu}_j$, $\mu_i < \mu_j$ should be considered the alternative hypothesis H_A for the verification and the opposite, which is $\mu_i \geq \mu_j$ should be considered the null hypothesis H_0:

$$H_0 : \mu_i \geq \mu_j, H_A : \mu_i < \mu_j. \tag{4}$$

For this test, the p-value, which represents the probability of obtaining the current results or more extreme results (which are unlikely to be obtained) under the assumption that H_0 is true, is calculated as follows:

$$\delta_{i<j} = \Pr\{X < t\}, \text{ where } X \sim t_v$$
$$t = (\hat{\mu}_i - \hat{\mu}_j) \Big/ \sqrt{s_i^2 \Big/ b_i + s_j^2 \Big/ b_j}$$
$$v = \left\lfloor \frac{\left(s_i^2 \Big/ b_i + s_j^2 \Big/ b_j\right)^2}{\left(s_i^2 / b_i\right)^2 \Big/ (b_i - 1) + \left(s_j^2 \Big/ b_j\right)^2 \Big/ (b_j - 1)} \right\rfloor. \tag{5}$$

The p-value $\delta_{i<j}$ follows a t-distribution with v degrees of freedom; the standard normal distribution can be used instead of a t distribution if the variance of the simulation output distribution is known. As $\delta_{i<j}$ becomes closer to zero, the observation of the current results under the assumption of H_0 being true becomes more difficult; thus, using the current results, H_0 is rejected and $\mu_i < \mu_j$ is verified.

Meanwhile, the value of $\delta_{i<j}$ is equal to the posterior probability that $\mu_i < \mu_j$ is false based on the observed results under the assumption of no prior information This means that the values are the same and not that the definitions are the same. When a noninformative prior assumption is used, the posterior distribution of μ_i follows a t-distribution with $\hat{\mu}_i$, s_i^2/b_i, and $b_i - 1$ degrees of freedom ($\Pr\{\mu_i/D_i\} \sim t_{b_i-1}(\hat{\mu}_i, s_i^2/b_i)$, where D_i denotes a set of observed simulation outputs for x_i) [34]. Subsequently, the posterior probability of $\mu_i < \mu_j$ can be defined as follows:

$$\Pr\{\mu_i < \mu_j/D_i, D_j\} = \Pr\{X < t'\}, \quad \text{where } X \sim t_v$$
$$t' = (\hat{\mu}_j - \hat{\mu}_i) \Big/ \sqrt{s_i^2 \Big/ b_i + s_j^2 \Big/ b_j}. \tag{6}$$

The degrees of freedom v in (6) are the same as those in (5). In both (5) and (6), the random variable X follows the same t-distribution. Because t in (5) is equal to $-t'$, $\delta_{i<j}$ has the same value as $1 - \Pr\{\mu_i < \mu_j/D_i, D_j\}$; alternatively as $\delta_{i<j}$ approaches zero, the posterior probability of $\mu_i < \mu_j$ approaches one. Thus, $\mu_i < \mu_j$ is verified.

Within a statistical hypothesis testing framework, the p-value can be used as an indicator to evaluate the observed simulation results. A value close to zero indicates that the results provide significant evidence for verifying the relation

established as the alternative hypothesis; alternatively the precision of the results ($s_i / \sqrt{b_i}$ and $s_j / \sqrt{b_j}$) is sufficient to statistically verify this relation. In contrast, a value close to 0.5 indicates that the results cannot be considered as significant evidence and that the level of precision is insufficient for verification. A value of 0.5 is achieved when $\hat{\mu}_i = \hat{\mu}_j$; this is the most uncertain case and in this case determining whether $\mu_i < \mu_j$ or $\mu_i > \mu_j$ is difficult Thus, when the p-value is close to 0.5, the precision of the results must be increased by additional replications to ensure that the relation set to H_A can be verified. The uncertainty is defined using this characteristic of the p-value for evaluating the observed simulation results for each alternative [16].

To maximize $P\{CS_n\}$, $\hat{S}_u$ should be equal to S_u for every $u \in \{1, \ldots, n\}$ According to (1), for an alternative x_i included in $\hat{S}_u$, the following relations between x_i and other alternatives should be verified to statistically determine that $x_i \in S_u$:

$$x_i \in \hat{S}_u : \left[\begin{array}{l} \forall x_b \in \bigcup_{j=1}^{u-1} \hat{S}_j : \mu_b < \mu_i \text{ and} \\ \forall x_w \in \bigcup_{j=u+1}^{n} \hat{S}_j : \mu_i < \mu_w \end{array} \right] \Rightarrow x_i \in S_u. \tag{7}$$

In other words, μ_i should be greater than the mean values of the alternatives in $\hat{S}_1 \ldots, \hat{S}_{u-1}$ and it should be less than the mean values of the alternatives in $\hat{S}_{u+1} \ldots, \hat{S}_n$. Each $k - m_u$ relation can be verified through statistical hypothesis testing, as presented in (5). For every test with the alternatives in $\hat{S}_1 \ldots, \hat{S}_{u-1}$ the p-value $\delta_{b<i}$ is obtained, while $\delta_{i<w}$ is obtained from every test with the alternatives in $\hat{S}_{u+1} \ldots, \hat{S}_n$ According to (2), no p-values are greater than 0.5. If every p value equals zero, $x_i \in \hat{\delta}S_u$ can be statistically determined However, if even one p value is close to 0.5, $x_i \in \hat{\delta}S_u$ is uncertain because the relation corresponding this p-value is not verified. Thus, the maximum value among these p-values can be used as a representative for evaluating the observed simulation results for x_i to statistically determine $x_i \in S_u$. This maximum value is defined as the uncertainty of x_i:

$$\omega_i = \max \begin{pmatrix} \delta_{b(1)<i}, \ldots, \delta_{b(h)<i}, \\ \delta_{i<w(l)}, \ldots, \delta_{i<w(k)} \end{pmatrix}, \tag{8}$$

where

$$h = \sum_{j=1}^{u-1} m_j \text{ and } l = \sum_{j=1}^{u} m_j + 1 = h + m_u + 1.$$

Here, the parentheses in the subscript of $\delta_{b<i}$ and $\delta_{i<w}$ indicates the observed relative rankings of $x_b \in \bigcup_{j=1}^{u-1} \hat{S}_j$ and $x_w \in \bigcup_{j=u+1}^{n} \hat{S}_j$. For example, $b(1)$ denotes the identifier of the alternative for which the observed relative ranking is the highest ($\hat{r}_{b(1)} = 1$)

The uncertainty ω_i derived from the combination of p-values indicates the degree to which the precision of $\hat{\mu}_i$ is sufficient to determine $x_i \in S_u$ A value close to zero indicates sufficient precision, whereas a value close to 0.5 indicates insufficient precision; thus, it is necessary to allocate additional replications to alternatives that have high uncertainty

to improve their precision. Combining the *p*-values using the maximal function rather than averaging or adding the *p*-values is semantically appropriate as well as allows a conservative allocation of additional replications using uncertainty; in other words, every *p*-value is considered important. This is a notable factor in increasing the simulation efficiency in high-noise contexts. Furthermore the uncertainty is the *p*-value; normalization is not needed.

According to (8), *p*-value calculations must be performed $k - m_u$ times to evaluate the uncertainty of the x_i in $\hat{S}_u$. Because one *p*-value is used to evaluate the uncertainty of two alternatives, the number of computations that are needed to evaluate the uncertainty of k alternatives is halved. Nevertheless, this computational complexity is derived as $O(k^2)$, which is higher than the complexity $O(k)$ of OCBAsr. To reduce this high complexity, a simple approximation is applied to (8) using the characteristics of *p*-values.

As is shown in (6), $\delta_{i<j}$ is affected by the difference between the sample means of the two alternatives $(\hat{\mu}_j - \hat{\mu}_i)$. When ω_i in (8) is evaluated, the maximum value of $\delta_{b<i}$ is very likely to be $\delta_{b(h)<i}$, which is the *p*-value derived from $\mu_{b(h)} < \mu_i$, because $\hat{\mu}_{b(h)}$ is the largest among the values of the alternatives in $\hat{S}_1 \ldots, \hat{S}_{u-1}$ and is the closest value to $\hat{\mu}_i$. Similarly, the maximum value of $\delta_{i<w}$ is very likely to be $\delta_{i<w(l)}$, which is the *p*-value derived from $\mu_i < \mu_{w(l)}$, because $\hat{\mu}_{w(l)}$ is the smallest among the values of the alternatives in $\hat{S}_{u+1} \ldots, \hat{S}_n$ and is the closest value to $\hat{\mu}_i$. Subsequently, ω_i in (8) can be approximated as follows:

$$\omega_i \approx \max\left(\delta_{b(h)<i}, \delta_{i<w(l)}\right). \tag{9}$$

This approximation introduces negligible error if all the alternatives have similar variances in their simulation outputs. This approximation remains effective as the allocation proceeds even if the variances are quite different, because the effect of the variance on the *p*-value gradually becomes negligible, as indicated in (6). When allocation begins, even if $\delta_{b(h)<i}$ (or $\delta_{i<w(l)}$) is not the maximum value of $\delta_{b<i}$ (or $\delta_{i<w}$), no considerable error is present in the uncertainty derived using (6) as $\delta_{b(h)<i}$ (or $\delta_{i<w(l)}$) is relatively large owing to the small difference between $\hat{\mu}_{b(h)}$ (or $\hat{\mu}_{w(l)}$) and $\hat{\mu}_i$ As a result, this approximation does not considerably affect the UE in (8). However, the complexity is reduced from $O(k^2)$ to $O(k)$ with the use of (9) because only two *p*-values need to be calculated to evaluate the uncertainty of each alternative. Furthermore, as presented in (1) the μ_i of x_i included in S_u is greater than the maximum among those of all alternatives in $S_1 \ldots, S_{u-1}$ and is less than the minimum among those of all alternatives in $S_{u+1} \ldots, S_n$; thus, this approximation is semantically appropriate.

Meanwhile, for the alternatives in $\hat{S}_1$, ω_i is approximated as $\delta_{i<w(l)}$ $(\forall x_i \in \hat{S}_1 : \omega_i \approx \delta_{i<w(l)})$ because $x_{b(h)}$ cannot be determined for them. Similarly, for the alternatives in $\hat{S}_n$, ω_i is approximated as $\delta_{b(h)<i}$ $(\forall x_i \in \hat{S}_n : \omega_i \approx \delta_{b(h)<i})$ because $x_{w(l)}$ cannot be determined for them.

B. ALLOCATION

When the uncertainty is evaluated using the observed simulation results for each alternative, alternatives with low uncertainty have sufficient precision of their simulation results compared to those of other alternatives to determine their relative rankings. In contrast, alternatives with relatively high uncertainty have insufficient precision of their simulation results compared to those of other alternatives.

As ω_i is a *p*-value, it does not represent the absolute precision of the simulation results of x_i. Another alternative for calculating this *p*-value $(= \omega_i)$ is denoted by x_j. The *p*-value indicates if the relative precision of $\hat{\mu}_i$ is sufficient to verify the relation between μ_i and μ_j when the precision of $\hat{\mu}_j$ is considered. Accordingly, even when the absolute precision of $\hat{\mu}_i$ is extremely high (i.e., $s_i / \sqrt{b_i}$ is close to zero), the *p*-value may have a high value that is close to 0.5 if the precision of $\hat{\mu}_j$ is quite low. In this case, these high ω_i values convey distorted information.

However, this is not a major problem when additional replications are allocated using uncertainty. The *p*-value between x_i and x_j is considered at the same time when ω_j and ω_i are evaluated; thus, if this *p*-value is set to ω_i, then ω_j is equal to or greater than ω_i. In other words, x_i and x_j are assigned at least the same number of additional replications according to their evaluated uncertainty. In the extreme case mentioned above, this allocation is not the best strategy that allocates more replications to x_j (because $\hat{\mu}_j$ has low absolute precision compared to $\hat{\mu}_i$). However, this allocation can always prevent the worst case in which more replications are allocated to x_i when $\hat{\mu}_i$ has high absolute precision compared to $\hat{\mu}_j$. Of course, the best strategy may be more efficient than this allocation; however, finding the best strategy would require additional computation, and if the number of additional replications is small, as is typical, the differential allocation between the two alternatives may be meaningless. Herein, we do not consider this uncertainty limitation because our goal is to develop a practical simple method to solve the problem presented in (3). Nevertheless, the proposed method exhibits higher efficiency than the existing methods, as described in the experimental results.

According to the meaning of uncertainty, it is necessary to allocate additional replications to alternatives that have high uncertainty. This is also supported by the posterior probability. According to the Fréchet inequality, which states that the upper bound of the posterior probability that $x_i \in \hat{S}_u$ is included in S_u is defined as (10), shown at the bottom of the next page.

As previously mentioned, under the no prior information assumption used in this study, $\Pr\{\mu_i < \mu_j / D_i, D_j\}$ has the same value as $1 - \delta_{i<j}$; thus, we obtain the following:

$$x_i \in \hat{S}_u : \Pr\left\{x_i \in S_u \Big/ \bigcup_{j \in \{1,\ldots,k\}} D_j\right\}$$

$$\leq \min\left(\begin{array}{c} 1 - \delta_{b(1)<i}, \ldots, 1 - \delta_{b(h)<i}, \\ 1 - \delta_{i<w(l)}, \ldots, 1 - \delta_{i<w(k)} \end{array}\right)$$

$$= 1 - \max \begin{pmatrix} \delta_{b(1)<i}, \ldots, \delta_{b(h)<i}, \\ \delta_{i<w(l)}, \ldots, \delta_{i<w(k)} \end{pmatrix}. \quad (11)$$

Applying the definition of uncertainty in (8), we obtain the following:

$$x_i \in \hat{S}_u : 1 - \Pr\left\{ x_i \in S_u \Big/ \bigcup_{j \in \{1, \ldots, k\}} D_j \right\} \geq \omega_i. \quad (12)$$

In other words, the lower bound of the posterior probability that $x_i \in \hat{S}_u$ is not included in S_u has the same value as ω_i A high value of ω_i means that the probability that the current observed decision (i.e., $x_i \in \hat{S}_u$) is false is confirmed to be high. In contrast, a low value of ω_i implies that the probability that $x_i \in \hat{S}_u$ is false may be high or low. Subsequently, if the simulation budget is limited, it is efficient to allocate additional replications to alternatives for which the posterior probability of being false is confirmed to be high (i.e., ω_i is high).

If the number of additional replications is limited, it is evident that most of them should be allocated to alternatives that have high uncertainty; however, a few additional replications should also be allocated to those alternatives having low uncertainty to consider the type-I error contained in the *p*-value. This type of error frequently occurs at the beginning of a sequential allocation procedure in extremely noisy contexts, where highly precise sample means are needed to distinguish between the actual relative rankings owing to the notable variation in the simulation output compared with the small differences between the means of alternatives.

For example, suppose we partition k alternatives into two ranked subsets, where the first set has one alternative ($m_1 = 1$) and all other alternatives belong to the second set ($m_2 = k - 1$). The alternative in S_1 is denoted as x_b, with $r_b = 1$. In extremely noisy contexts, $\hat{\mu}_b$ may be highly biased after a few replicated simulations. In this case, x_b which belongs to $\hat{S}_2$ has relatively low uncertainty owing to the large difference between $\hat{\mu}_b$ and the sample mean of the alternative in $\hat{S}_1$; thus no additional replications are allocated to x_b. Accordingly, the biased $\hat{\mu}_b$ cannot be updated and two ranked subsets cannot be correctly selected ($\hat{S}_1 \neq S_1$ and $\hat{S}_2 \neq S_2$). Although x_b requires additional replications as soon as possible, it fails to receive allocated replications because of the type-I error. If the sequential procedure continues, additional replications may eventually be allocated to x_b owing to its increased uncertainty (i.e., the precision of $\hat{\mu}_b$ is relatively low). However, many replications are inefficiently allocated in the meantime To prevent this wastage in extremely noisy contexts, one or two additional replications must be allocated to alternatives that have relatively low uncertainty. This is because, as in this example, their sample mean is highly biased and the number of output samples is small Therefore,

even one additional output sample can considerably change the sample mean

In summary, to increase $P\{CS_n\}$ using uncertainty, most of additional replications should be allocated to alternatives that have relatively high uncertainty However, a few additional replications should be simultaneously allocated to alternatives that have relatively low uncertainty. This strategy is expressed in the following allocation policy [16]:

$$a_i = (\omega_i)^C \Big/ \sum_{j=1}^{k} (\omega_j)^C \cdot \Delta, \quad (13)$$

where ω_i denotes the evaluated uncertainty of x_i, a_i denotes the number of additional replications to be allocated to x_i, and Δ denotes the additional simulation budget ($\sum_{i=1}^{k} a_i = \Delta$), which represents the small unit of B in the sequential allocation procedure.

The parameter C is used to control this strategy. As the value of C increases, additional replications are concentrated on alternatives having relatively high uncertainty. However, as the value of C decreases, alternatives that have a relatively low uncertainty are more likely to have additional replications allocated to them. The optimal value of C for maximizing $P\{CS_n\}$ is problem-specific. If the problem has a low level of stochastic noise, the frequency of the type-I error is low; thus, a high C value that allocates additional replications to alternatives that have a relatively high uncertainty may be effective. Conversely, for a high level of noise, a low C value may effectively cope with type-I errors.

Herein, we set the default C value to one. Although this value may not be optimal for all problems, as is described in the next section, the proposed method with this default setting exhibits superior efficiency over existing methods. Moreover, the proposed method partitions k alternatives into n ($3 < n < k$) ranked subsets by generalizing the existing R&S studies that select between two, three, and k ranked subsets, as presented in Table 1. Accordingly, in each special case, the uncertainty evaluation of the proposed method in (8), or the approximated version in (9), becomes identical to that of the existing UE-based method designated for this special case. For example, when $n = 2$ and $m_1 = 1$, the problem in (3) falls into the single optimal alternative selection and the uncertainty evaluation in (8) becomes identical to that of [16]. Similarly, when $n = k$ and $m_{\forall u} = 1$, the problem is the same as the complete ranking identification, and the uncertainty evaluation in (9) becomes identical to that of [27]. Because the size of every subset is one, the approximation in (9) becomes meaningless and (8) can be used instead. In the existing methods for special cases [16], [17], [24], [26], and [27], the value of C is empirically derived as one through a simulation-based optimization of benchmarks or randomly

$$x_i \in \hat{S}_u : \Pr\left\{ x_i \in S_u \Big/ \bigcup_{j \in \{1, \ldots, k\}} D_j \right\} \leq \min \begin{pmatrix} \Pr\left\{ \mu_{b(1)} < \mu_i \Big/ D_{b(1)}, D_i \right\}, \ldots, \Pr\left\{ \mu_{b(h)} < \mu_i \Big/ D_{b(h)}, D_i \right\}, \\ \Pr\left\{ \mu_i < \mu_{w(l)} \Big/ D_i, D_{w(l)} \right\}, \ldots, \Pr\left\{ \mu_i < \mu_{w(k)} \Big/ D_i, D_{w(k)} \right\} \end{pmatrix}. \quad (10)$$

Algorithm 1 Partitioning Θ (k Alternatives) Into n ($2 \le n \le k$) Ranked Subsets, Each Comprising $m_{u \in \{1, \ldots, n\}}$ Alternatives ($\sum_{u=1}^{n} m_u = k$)

Input parameters: Θ, n, $\{m_1, \ldots, m_n\}$, and B.
Control parameters: α, Δ, and C (default setting is 1).
Output: $\hat{S}_1, \ldots, \hat{S}_n$.

Initialization:
1: **for each** $x_i \in \Theta$ **do**:
2:     **simulate** α times
3:     **set** $b_i \leftarrow \alpha$; **calculate** $\hat{\mu}_i$ and s_i^2
4: **select** $\hat{S}_1, \ldots, \hat{S}_n$ using (2)
Loop:
5: **while** $\sum_{i=1}^{k} b_i < B$ **do**:
6:     **set** $\Delta \leftarrow \min \left(B - \sum_{i=1}^{k} b_i, \Delta \right)$
7:     **for each** $x_i \in \Theta$: **do**:
8:        **evaluate** ω_i using (9)
9:     **for each** $x_i \in \Theta$ **do**:
10:       **calculate** a_i using (13)
11:       **simulate** a_i times
12:       **set** $b_i \leftarrow b_i + \text{round}(a_i)$[a]; **update** $\hat{\mu}_i$ and s_i^2
13:     **select** $\hat{S}_1, \ldots, \hat{S}_n$ using (2)
14: **return** $\hat{S}_1, \ldots, \hat{S}_n$

---

[a]The rounding error ($\Delta - \sum_{i=1}^{k} \text{round}(a_i)$) is distributed to the alternatives that are allocated additional replications (at most ± 1 for each alternative).

generated problems. This consistency supports the default setting of the proposed method.

Meanwhile, a_i should be an integer when additional simulations are conducted; however, a_i is typically not an integer according to (13) Thus, a conversion method should be used when the proposed method is implemented. Fortunately, owing to the advantage of the sequential procedure [8], any reasonable approach that allows a_i/a_j to be similar to $\left(\omega_i/\omega_j \right)^C$ would not considerably change the efficiency of the proposed method. Herein, a rounding function is used to convert a_i to an integer and the error ($\Delta - \sum_{i=1}^{k} \text{round}(a_i)$) is distributed to the alternatives that are allocated additional replications (at most ± 1 for each alternative).

C. SEQUENTIAL PROCEDURE

As we previously mentioned, the proposed method divides B into smaller units and sequentially allocates the units to efficiently utilize B. Alternatively, it repeats 1) evaluating the uncertainty using (9) based on the observed simulation results, 2) allocating Δ using (13) based on the evaluated uncertainty and 3) updating the simulation results. Algorithm 1 presents this sequential allocation procedure

Algorithm 1 takes k alternatives, the number of ranked subsets, the size of each subset, and the simulation budget as input; it returns $\hat{S}_1, \ldots, \hat{S}_n$ that are correctly selected to be $\forall u \in \{1, \ldots, n\} : \hat{S}_u = S_u$ Algorithm 1 comprises two phases: the initialization and the loop. In this study,

we assume that no prior information about the output distribution of each alternative is provided before the simulation is performed. Thus, to obtain the initial simulation results for evaluating uncertainty, every alternative is simulated α times during the initialization phase. Then, in the loop phase, Δ replications are iteratively allocated using the evaluated uncertainty to gradually update the simulation results until the remaining budget ($B - k\alpha$) is depleted. Finally, $\hat{S}_1 \ldots, \hat{S}_n$ obtained from the observed simulation results are returned as the output.

Algorithm 1 has three parameters that control the sequential allocation procedure: α, Δ, and C (C is described in the previous subsection). α denotes the number of initial replications required by every alternative to obtain the initial simulation results. If the α value is too small, the initial results may be poor, making the subsequent iterative allocations inefficient. Furthermore, a value of α that is too large relative to B (i.e., close to B/k) reduces the efficiency of the algorithm as it does not allow a sufficient number of iterations in the loop phase. The literature suggests that α values between 5 and 20 are appropriate [35]. However, when the problem has a lot of noise and B is large, a larger value of α than suggested may reduce the type-I error and increase the algorithm efficiency. Meanwhile, Δ denotes the number of additional replications to be distributed to the alternatives using the uncertainty evaluated in each iteration of the loop phase. If the Δ value is too small relative to that of k, the efficiency of the algorithm may be reduced because a few additional replications are allocated only to alternatives with relatively high uncertainty. Furthermore, a value that is too large relative to B hides the advantage of the sequential allocation procedure. Empirical studies suggest that Δ values between $0.5k$ and $0.05B$ are appropriate [16]. In extremely noisy contexts, setting a large Δ value within the suggested range may improve the algorithm efficiency by increasing the possibility of allocating a few additional replications to alternatives that have relatively low uncertainty.

IV. EXPERIMENTS
In this section, our experimental results for the numerical and practical problems are presented to demonstrate the high efficiency of the proposed method.

A. NUMERICAL PROBLEMS
The same numerical problems that were used to test OCBAsr were used herein to evaluate the proposed method [32]. Table 3 presents the two types of stochastic numerical models for these problems: equal variance (EV) and unequal variance (UV). For each model, μ_i is equal to the index i. The EV model has the same variance across all simulation outputs, whereas the UV model has different variances. Furthermore, there are two versions of each model. The high-noise version has a higher noise level than the normal version owing to reduction in the difference between the means of the alternatives. The high-noise versions were not employed to test OCBAsr; however, they are introduced in this study
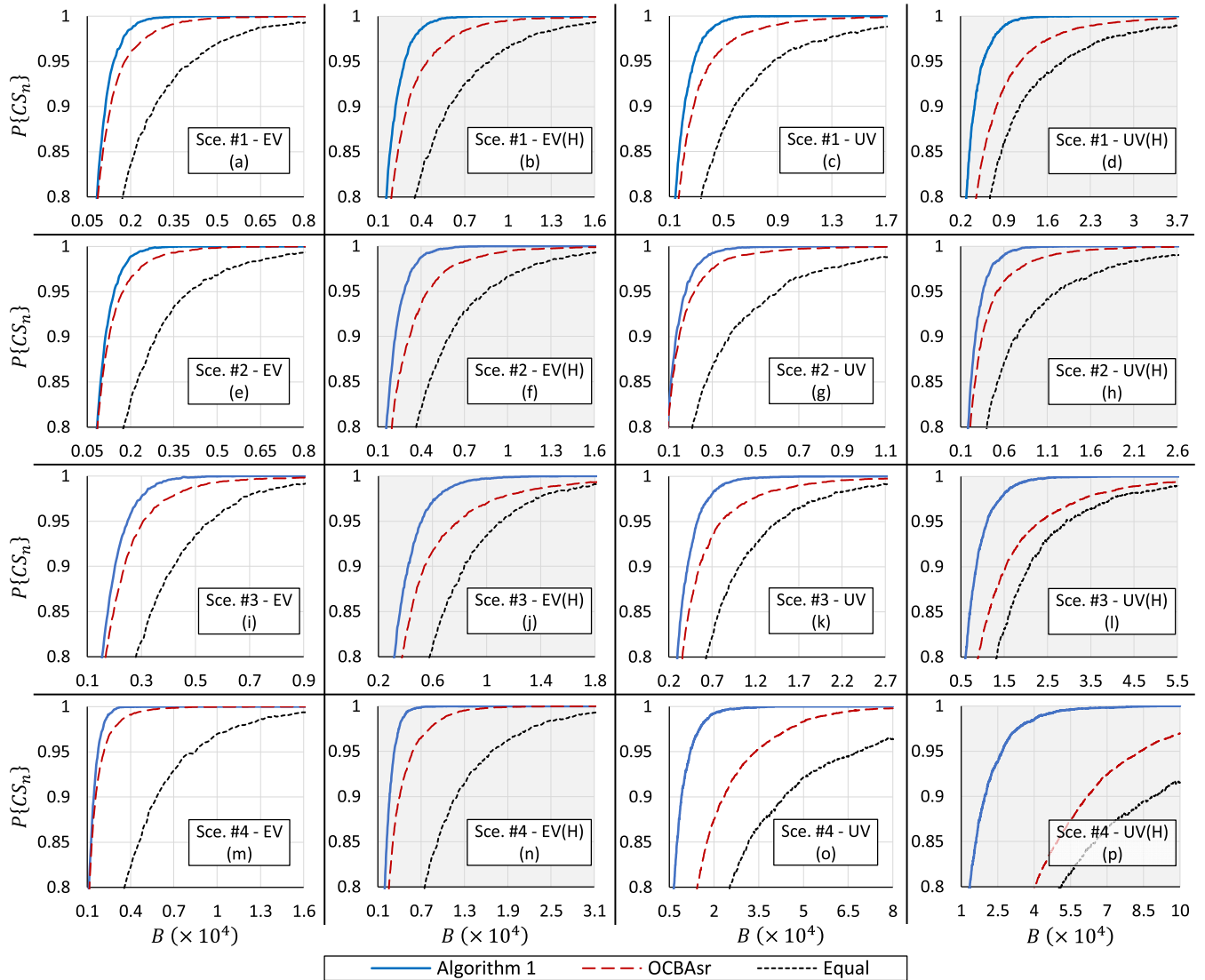
FIGURE 1. Convergence curves of $P\{CS_n\}$ against B for 16 numerical problems, where the shaded graphs correspond to the results for the high-noise versions.

TABLE 3. Two stochastic models for numerical problems.

Description	Normal ver.	High-noise ver. (H)
Equal variance (EV)	$Y_i \sim \mathcal{N}(i, 6^2)$	$Y_i \sim \mathcal{N}(0.7i, 6^2)$
Unequal variance (UV)	$Y_i \sim \mathcal{N}(i, i^2)$	$Y_i \sim \mathcal{N}(0.7i, i^2)$

TABLE 4. Four scenarios for numerical problems.

No.	Description	k	n	$\{m_1, \dots, m_n\}$
1	Equal division	15	3	$\{5, 5, 5\}$
2	Unequal division	15	3	$\{3, 5, 7\}$
3	Large number of divisions	15	5	$\{3, 3, 3, 3, 3\}$
4	Large number of alternatives	30	3	$\{10, 10, 10\}$

to demonstrate the practicality of the proposed method in extremely noisy contexts

By combining these four stochastic models with the four scenarios presented in Table 4, 16 numerical problems were generated. The experimental results of the proposed method for the extreme cases (e.g., $n = 2$ or $n = k$) can be found in [16] ($n = 2$ and $m_1 = 1$), [17] ($n = 2$ and $m_1 = m$), and [27] ($n = k$ and $m_{\forall u} = 1$) For all 16 problems, Algorithm 1 was compared with OCBAsr and the equal allocation method. For OCBAsr, OCBAsr$_{mp}$ was used because it exhibited the best

overall performance among the various versions of OCBAsr presented in [32]. For both Algorithm 1 and OCBAsr, α and Δ were set to 20 and 50 respectively except in Scenario 4 (in which Δ was set to 100 because $k = 30$). To compare the efficiency of each method, $P\{CS_n\}$ was calculated according to the simulation budget, as depicted in Fig. 1, where each $P\{CS_n\}$ value was estimated in 10,000 independent experiments. Table 5 shows the average budget required to achieve $0.99\ P\{CS_n\}$.

TABLE 5. Average simulation budget required to reach 0.99 $P\{CS_n\}$ for 16 numerical problems.

Problem			R&S methods	
Sec.	Model	Algorithm 1	OCBAsr	Equal
1	EV	2,150	3,350 (*1.56*)[a]	7,230 (*3.36*)
	EV(H)	4,300	8,100 (*1.88*)	14,520 (*3.38*)
	UV	4,400	8,900 (*2.02*)	17,490 (*3.98*)
	UV(H)	9,150	23,750 (*2.60*)	36,750 (*4.02*)
2	EV	2,050	3,100 (*1.51*)	7,320 (*3.57*)
	EV(H)	4,150	8,500 (*2.05*)	14,970 (*3.61*)
	UV	2,800	4,300 (*1.54*)	11,460 (*4.09*)
	UV(H)	5,950	11,450 (*1.92*)	25,320 (*4.26*)
3	EV	3,500	5,250 (*1.50*)	8,580 (*2.45*)
	EV(H)	7,650	15,750 (*2.06*)	17,850 (*2.33*)
	UV	8,100	16,850 (*2.08*)	26,310 (*3.25*)
	UV(H)	17,450	46,650 (*2.67*)	56,280 (*3.23*)
4	EV	2,400	3,900 (*1.63*)	14,640 (*6.10*)
	EV(H)	4,700	10,800 (*2.30*)	28,500 (*6.06*)
	UV	18,700	57,300 (*3.06*)	>10^5[0.9792] (*>5.35*)
	UV(H)	42,750	>10^5[0.9697] (*>2.34*)	≫10^5[0.9148] (*≫2.34*)

[a]The numbers in parentheses indicate the proportions of the budgets required for Algorithm 1.

The experimental results show the superior efficiency of Algorithm 1 over the other methods. In particular, when the results of the normal and high-noise versions are compared, the improvement in the efficiency of Algorithm 1 is more pronounced in the high-noise version. For example, as presented in Table 5, in Scenario 2, the simulation budget required by OCBAsr to reach 0.99 $P\{CS_n\}$ for the EV model is approximately 1.51 times of that required by Algorithm 1 but 2.05 times of the Algorithm 1 budget for the EV(H) model. In Scenario 3, the OCBAsr budget for the UV model is approximately 2.08 times of the Algorithm 1 budget and 2.67 times of the Algorithm 1 budget for the UV(H) model. In Scenario 4, Algorithm 1 achieved 0.99 $P\{CS_n\}$ within 42,750 replications for the UV(H) model; however, OCBAsr did not achieve this value even after allocating 10^5 replications.

This improved efficiency compared with that of OCBAsr is owing to the fact that Algorithm 1 does not rely on heuristic constants to select ranked subsets; however, the fundamental reason is that Algorithm 1 considers the precision of the observed simulation results during the allocation. Although the allocation by OCBAsr considers the $\hat{\mu}_i$ and s_i^2 values for each x_i based on the constants, the allocation by Algorithm 1 considers the value of b_i as well. Thus, Algorithm 1 considers $\hat{\mu}_i$ and the precision of $\hat{\mu}_i$ (s_i^2/b_i) during the allocation, as described in (6). If the $\hat{\mu}_i$ value for an alternative x_i is considerably biased after the initial simulation, OCBAsr does not allocate additional replications to x_i because of the poor value of $\hat{\mu}_i$. Subsequently, $\hat{\mu}_i$ and s_i^2 are not updated and the incorrect selections of $\hat{S}_1 \ldots, \hat{S}_n$ continue. For example, in the practical problem of the next subsection, alternative that is misclassified in this way received only 21 replications on average after 23,000 replications were allocated by OCBAsr; there was negligible additional allocation after the initial allocation. On the other hand, as the sequential

allocation procedure continues, the precision of $\hat{\mu}_i$ that does not receive additional replications becomes relatively lower compared to that of the others. Thus, considering the precision, Algorithm 1 certainly allocates additional replications to x_i unlike OCBAsr so that incorrect selections are quickly corrected. The likelihood of obtaining poor initial simulation results increases with increasing noise. Algorithm 1, which overcomes such situations well, is more efficient in high-noise contexts.

The problem of poor initial results can be solved by increasing the number of initial replications (α). However, in practice, identifying the noise level associated with a problem is difficult before the simulations are performed; thus, setting an unconditionally high α value may result in the wastage of the simulation budget without any improvement in the efficiency. Algorithm 1, which is relatively independent of the accuracy of the initial results compared with OCBAsr and which reliably and quickly corrects errors through additional allocations using uncertainty, is more efficient and practical, as demonstrated by the results of the practical problem described in the next subsection.

Meanwhile, in low-noise contexts, as a notable bias of $\hat{\mu}_i$ is unlikely to happen, the allocation of Algorithm 1 considering the precision of $\hat{\mu}_i$ may be redundant and may consume more simulation budget compared to OCBAsr. However, these increments may be inconsiderable because the budget required to accurately select ranked subsets is small when the noise level is low. Furthermore, the practical simulation models targeted by Algorithm 1 typically involve high stochastic noise to reflect the high levels of uncertainty in complex systems; thus, Algorithm 1 is still effective.

B. PRACTICAL PROBLEM—RELOCATIONZONE DISTRIBUTION

Public bicycle-sharing systems (PBSSs) are an ecofriendly means of transportation in many cities. A major operational concern of PBSSs is the relocation of bicycles accumulated in a specific area according to rental and return locations [36]. Eoulling, the PBSS in Sejong City, which is the administrative capital of South Korea, is no exception. For the efficient relocation of bicycles, as depicted in Fig. 2, the Eoulling managers divide bicycle stations into 16 zones and distribute the relocation task for each zone to five companies. However, the companies have different relocation capabilities. For example, company A can simultaneously perform relocation work for four zones, but the efficiency of their work per hour is low as most of the employees are beginners. Conversely, company B can handle only two zones simultaneously, but its efficiency is high because of the availability of skilled workers. Each zone requires a different amount of relocation work to minimize the inconvenience to citizens. Thus, given the capabilities of the five companies, if these 16 zones are partitioned into five ranked subsets according to the workload and are then distributed to these companies, the effectiveness of the relocation task can be maximized. The workload of each zone can be evaluated using the stochastic
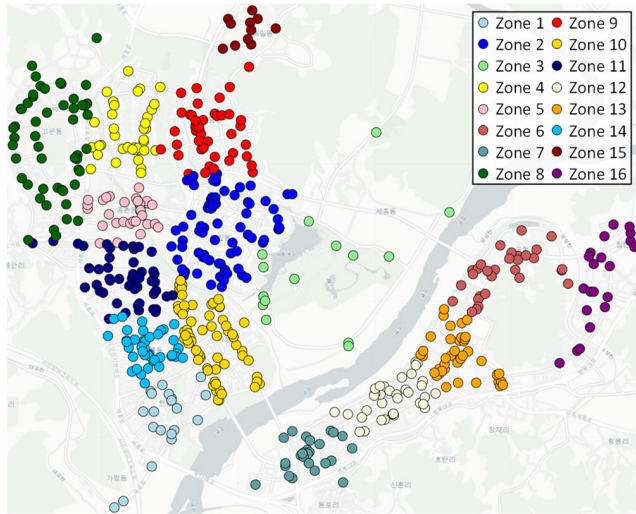
FIGURE 2. Eoulling's 16 relocation zones in Sejong City.

TABLE 6. Evaluated relocation workload for 16 relocation zones.

Zone (i)	$\hat{u}_i$ ($\approx \mu_i$)	s_i^2 ($\approx \sigma_i^2$)	Zone (i)	$\hat{u}_i$ ($\approx \mu_i$)	s_i^2 ($\approx \sigma_i^2$)
1	92.294	1048.071	9	97.83	563.278
2	330.119	4892.501	10	175.926	1604.786
3	49.305	359.713	11	86.903	396.266
4	62.725	296.839	12	126.495	902.337
5	64.947	313.586	13	84.037	447.731
6	91.093	675.505	14	57.835	245.376
7	109.619	991.014	15	25.422	140.760
8	75.393	267.883	16	77.194	677.363

TABLE 7. Average simulation budget required to reach a specific $P\{CS_n\}$ in the practical problem.

$P\{CS_n\}$	Algorithm 1	OCBAsr	Equal
0.90	2,620	3,320 (*1.27*)	8,510 (*3.25*)
0.95	3,470	4,920 (*1.42*)	12,860 (*3.71*)
0.99	5,520	26,470 (*4.80*)	24,380 (*4.42*)

simulation [29], and Table 6 presents the precisely evaluated workload.

Algorithm 1 was employed to efficiently solve this practical simulation problem. In addition, OCBAsr and the equal allocation method were applied to demonstrate the utility and improved efficiency of Algorithm 1. Considering the capabilities of the five companies, $\{m_1, \ldots, m_5\}$ were set to $\{4, 2, 4, 3, 3\}$. For both Algorithm 1 and OCBAsr, α and Δ were set to 20 and 50 respectively. As in the numerical problems, $P\{CS_n\}$ was measured according to the simulation budget, as depicted in Fig. 3, where every $P\{CS_n\}$ was estimated over 10,000 independent experiments. Table 7 presents the budgets required to achieve 0.90, 0.95, and 0.99 $P\{CS_n\}$.

The experimental results, like the results of the previous numerical problems, demonstrate the superior efficiency of Algorithm 1. As shown in Table 6, the variance is greater than the interval between the means of the alternatives. In this high-noise practical problem OCBAsr required an enormous
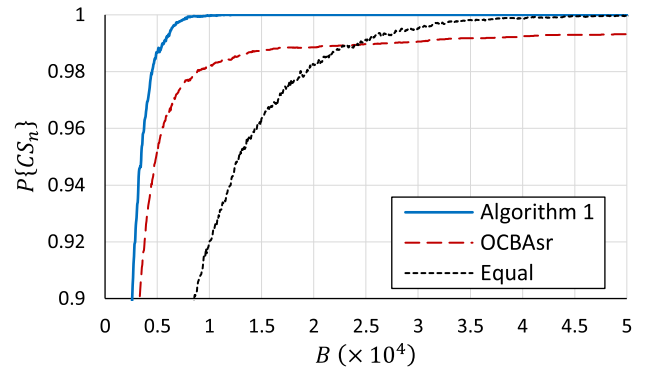

FIGURE 3. Convergence curve of $P\{CS_n\}$ against B for the practical problem.

budget to reach a high $P\{CS_n\}$ value and it did not converge to the maximum $P\{CS_n\}$ after allocating 50,000 replications. After 23,000 replications, the $P\{CS_n\}$ of OCBAsr was lower than that of the equal allocation method, as depicted in Fig. 3. As mentioned previously, this is because the biased $\hat{\mu}_i$ of x_i is not updated by OCBAsr after the initial simulation, resulting in continuous incorrect selections of $\hat{S}_1 \ldots, \hat{S}_n$. Among 10,000 experiments on a budget of 23,000, the average number of replications allocated to the misclassified alternative was only 21.16 (approximately 0.092% of the allocated budget) for 102 cases in which OCBAsr persisted with an incorrect selection until the end; in other words, after the initial allocation ($\alpha = 20$), this alternative received little additional replication over 454 sequential allocations of Δ because of its poor initial results. In contrast, Algorithm 1 reached 0.99 $P\{CS_n\}$ after allocating only 5,520 replications. In addition, its $P\{CS_n\}$ converged to the maximum value within 10,000 replications, as depicted in Fig. 3.

The simulation model [29] was synchronized with the operational data collected from Eoulling in real time, and the relocation workload was evaluated through simulations of the synchronized model. Relocations were typically performed in the morning, afternoon, and evening; thus, each workload for these three times was evaluated using the model synchronized with the data collected in the evening of the previous day, morning, and afternoon, respectively. Within approximately one hour after synchronization, the 16 zones had to be distributed among the five companies according to the simulation results before the relocation could begin. Because the model simulates 260,000 citizens of Sejong City as agents, the process is complex and expensive (requiring approximately 0.5 s per replication on an i5-10400 2.9 GHz CPU, with 16 GB RAM and CentOS 7). As presented in Table 7, Algorithm 1 can accurately partition the 16 zones into five ranked subsets using the simulation model within one hour; however, the OCBAsr and equal allocation methods are time-consuming because of their low efficiencies As such, Algorithm 1 is essential for increasing the simulation efficiency in practical systems where digital twins are employed.

V. CONCLUSION

We proposed an R&S method that efficiently allocates a simulation budget to partition k alternatives into n exclusive ranked subsets when alternatives are evaluated using a stochastic simulation model. To improve the simulation efficiency for complex practical problems involving high noise, the proposed method is based on UE that has high robustness to noise. Using hypothesis testing and *p*-value, the uncertainty was defined to evaluate if the observed simulation results of each alternative had sufficient precision to accurately select the ranked subsets. The proposed method splits a given budget into a small number of simulation replications and sequentially allocates them using the evaluated uncertainty for each alternative, thereby gradually improving the insufficient precision to maximize $P\{CS_n\}$ within the budget. Our numerical experiments demonstrated the superior efficiency of the proposed method compared with that of OCBAsr Notably, the superiority was more pronounced in high-noise contexts. In addition, the practical problem of relocation-zone distribution demonstrated the utility of the proposed method in practical scenarios.

Owing to its superior efficiency in high-noise contexts, the proposed method is expected to be effective in situations that require high simulation efficiency such as digital twins for complex systems However, further studies to mitigate the rather strict assumptions are needed to improve the practicality of the proposed method. First, this study assumes that the stochastic simulation output follows a normal distribution; however, in reality, this may not be the case [37]. Moreover, in practice, an output distribution is typically unknown; thus, it is necessary to generalize the proposed method to the assumption of a general underlying distribution. Second, this study did not consider the correlation between alternatives. The proposed method, which is based on the assumption of independence between alternatives, is applicable even if the alternatives are correlated, but it may be inefficient because it does not use this correlation information. As the quantified correlation information (i.e., the covariance matrix) is typically unknown, the simulation budget can be further reduced in practical scenarios if correlation information is estimated and utilized during the sequential allocation procedure Finally, a method for determining the optimal C value to maximize $P\{CS_n\}$ according to the characteristics of the problem is left for future research.

ACKNOWLEDGMENT

REFERENCES

[1] S. Yang, K. Song, and G. Zhu, "Stochastic process and simulation of traction load for high speed railways," *IEEE Access*, vol. 7, pp. 76049–76060, 2019.

[2] B. G. Kang, K.-M. Seo, and T. G. Kim, "Model-based design of defense cyber-physical systems to analyze mission effectiveness and network performance," *IEEE Access*, vol. 7, pp. 42063–42080, 2019.

[3] L. Wang, R. Iida, and A. M. Wyglinski, "Vehicular network simulation environment via discrete event system modeling," *IEEE Access*, vol. 7, pp. 87246–87264, 2019.

[4] A. C. Lisboa, F. H. B. De Souza, C. M. Ribeiro, C. A. Maia, R. R. Saldanha, F. L. B. Castro, and D. A. G. Vieira, "On modelling and simulating open pit mine through stochastic timed Petri nets," *IEEE Access*, vol. 7, pp. 112821–112835, 2019.

[5] Y. Liu, K. Wang, L. Ge, L. Ye, and J. Cheng, "Adaptive evaluation of virtual machine placement and migration scheduling algorithms using stochastic Petri nets," *IEEE Access*, vol. 7, pp. 79810–79824, 2019.

[6] M. C. Fu, "Optimization for simulation: Theory vs. practice," *INFORMS J. Comput.*, vol. 14, no. 3, pp. 192–215, Aug. 2002.

[7] S. H. Choi and T. G. Kim, "Enhancing the noise robustness of the optimal computing budget allocation approach," *IEEE Access*, vol. 8, pp. 25749–25763, 2020.

[8] C.-H. Chen and L. H. Lee, *Stochastic Simulation Optimization: An Optimal Computing Budget Allocation*, vol. 1. Singapore: World Scientific, 2011.

[9] Y. C. Ho, R. S. Sreenivas, and P. Vakili, "Ordinal optimization of DEDS," *Discrete Event Dyn. Syst.*, vol. 2, no. 1, pp. 61–88, Jul. 1992.

[10] H. Xiao, L. H. Lee, and K. M. Ng, "Optimal computing budget allocation for complete ranking," *IEEE Trans. Autom. Sci. Eng.*, vol. 11, no. 2, pp. 516–524, Apr. 2014.

[11] S.-H. Kim and B. L. Nelson, "Selecting the best system," in *Handbooks in Operations Research and Management Science: Simulation*, S. G. Henderson and B. L. Nelson, Eds. Oxford, U.K.: Elsevier, 2006, pp. 501–534.

[12] C.-H. Chen, J. Lin, E. Yücesan, and S. E. Chick, "Simulation budget allocation for further enhancing the efficiency of ordinal optimization," *Discrete Event Dyn. Syst.*, vol. 10, no. 3, pp. 251–270, Jul. 2000.

[13] S. E. Chick and K. Inoue, "New two-stage and sequential procedures for selecting the best simulated system," *Oper. Res.*, vol. 49, no. 5, pp. 732–743, Oct. 2001.

[14] J. Branke, S. E. Chick, and C. Schmidt, "Selecting a selection procedure," *Manage. Sci.*, vol. 53, no. 12, pp. 1916–1932, Dec. 2007.

[15] J. Xu, E. Huang, C.-H. Chen, and L. H. Lee, "Simulation optimization: A review and exploration in the new era of cloud computing and big data," *Asia–Pacific J. Oper. Res.*, vol. 32, no. 3, Jun. 2015, Art. no. 1550019.

[16] S. H. Choi and T. G. Kim, "Efficient ranking and selection for stochastic simulation model based on hypothesis test," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 48, no. 9, pp. 1555–1565, Sep. 2018.

[17] S. H. Choi and T. G. Kim, "Optimal subset selection of stochastic model using statistical hypothesis test," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 48, no. 4, pp. 557–564, Apr. 2018.

[18] S. H. Choi and T. G. Kim, "Pareto set selection for multiobjective stochastic simulation model," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 50, no. 11, pp. 4256–4269, Nov. 2020.

[19] S. H. Choi, B. G. Kang, and T. G. Kim, "A heuristic procedure for selecting the best feasible design in the presence of stochastic constraints," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 51, no. 2, pp. 1016–1026, Feb. 2021.

[20] C.-H. Chen, D. He, M. Fu, and L. H. Lee, "Efficient simulation budget allocation for selecting an optimal subset," *INFORMS J. Comput.*, vol. 20, no. 4, pp. 579–595, Nov. 2008.

[21] S. Zhang, L. H. Lee, E. P. Chew, J. Xu, and C.-H. Chen, "A simulation budget allocation procedure for enhancing the efficiency of optimal subset selection," *IEEE Trans. Autom. Control*, vol. 61, no. 1, pp. 62–75, Jan. 2016.

[22] S. Gao and W. Chen, "Efficient subset selection for the expected opportunity cost," *Automatica*, vol. 59, pp. 19–26, Sep. 2015.

[23] J. Zhang, L. Zhang, C. Wang, and M. Zhou, "Approximately optimal computing budget allocation for selection of the best and worst designs," *IEEE Trans. Autom. Control*, vol. 62, no. 7, pp. 3249–3261, Jul. 2017.

[24] S. H. Choi and C. Choi, "A simulation budget allocation procedure for finding both extreme designs simultaneously in discrete-event simulation," *IEEE Access*, vol. 8, pp. 93978–93986, 2020.

[25] H. Xiao, S. Gao, and L. H. Lee, "Simulation budget allocation for simultaneously selecting the best and worst subsets," *Automatica*, vol. 84, pp. 117–127, Oct. 2017.

[26] S. H. Choi and B. S. Kim, "A robust method for identifying the best and worst subsets in stochastic simulation," *Int. J. Simul. Model.*, vol. 20, no. 3, pp. 477–488, Sep. 2021.

[27] S. H. Choi and T. G. Kim, "A heuristic approach for selecting best-subset including ranking within the subset," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 50, no. 10, pp. 3852–3862, Oct. 2020.

[28] B. P. Zeigler, T. G. Kim, and H. Praehofer, *Theory of Modeling and Simulation: Integrating Discrete Event and Continuous Complex Dynamic Systems*, 2nd ed. San Diego, CA, USA: Academic, 2000.

[29] J. W. Bae, C.-H. Lee, J.-W. Lee, and S. H. Choi, "A data-driven agent-based simulation of the public bicycle-sharing system in Sejong City," *Simul. Model. Pract. Theory*, to be published.

[30] S. Greco, V. Mousseau, and R. Słowiński, "Ordinal regression revisited: Multiple criteria ranking using a set of additive value functions," *Eur. J. Oper. Res.*, vol. 191, no. 2, pp. 416–436, Dec. 2008.

[31] D. Cossock and T. Zhang, "Statistical analysis of Bayes optimal subset ranking," *IEEE Trans. Inf. Theory*, vol. 54, no. 11, pp. 5140–5154, Nov. 2008.

[32] J. Zhang, Z. Li, C. Wang, D. Zang, and M. Zhou, "Approximate simulation budget allocation for subset ranking," *IEEE Trans. Control Syst. Technol.*, vol. 25, no. 1, pp. 358–365, Jan. 2017.

[33] E. Negri, L. Fumagalli, and M. Macchi, "A review of the roles of digital twin in CPS-based production systems," *Proc. Manuf.*, vol. 11, pp. 939–948, Sep. 2017.

[34] K. Inoue and S. E. Chick, "Comparison of Bayesian and frequentist assessments of uncertainty for selecting the best system," in *Proc. Winter Simulation Conf.*, Washington, DC, USA, 1998, pp. 727–734.

[35] R. E. Bechhofer, T. J. Santner, and D. M. Goldsman, *Design and analysis of Experiments for Statistical Selection, Screening, and Multiple Comparisons*. New York, NY, USA: Wiley, 1995.

[36] C. S. Shui and W. Y. Szeto, "A review of bicycle-sharing service planning problems," *Transp. Res. C, Emerg. Technol.*, vol. 117, Aug. 2020, Art. no. 102648.

[37] M. Broadie, M. Han, and A. Zeevi, "Implications of heavy tails on simulation-based ordinal optimization," in *Proc. Winter Simulation Conf.*, Dec. 2007, pp. 439–447.

MOON GI SEOK (Member, IEEE) received the B.S. degree in electronics engineering from Korea University, Seoul, South Korea, in 2009, and the M.S. and Ph.D. degrees in electrical engineering from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea, in 2011 and 2017, respectively. From 2017 to 2019, he was a Postdoctoral Researcher with KAIST and Arizona State University (ASU), Tempe, AZ, USA. From 2019 to 2023, he was a Research Fellow and a Senior Research Fellow with Nanyang Technological University (NTU), Singapore. Since 2023, he has been an Assistant Professor with the Department of Computer Science and Engineering, Dongguk University, Seoul. His research interests include digital twinning for manufacturing systems, parallel and distributed simulations, model verification, and hardware/software co-design.

SEON HAN CHOI (Member, IEEE) received the B.S., M.S., and Ph.D. degrees in electrical engineering from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea, in 2012, 2014, and 2018, respectively. In 2018, he was a Postdoctoral Researcher with the Information and Electronics Research Institute, KAIST. From 2018 to 2019, he was a Senior Researcher with the Korea Institute of Industrial Technology, Ansan, South Korea. From 2019 to 2022, he was an Assistant Professor with the Department of IT Convergence and Application Engineering, Pukyong National University, Busan, South Korea. In 2022, he joined the Department of Electronic and Electrical Engineering, Ewha Womans University, Seoul, South Korea, as an Assistant Professor. His current research interests include the modeling and simulation of discrete-event systems, efficient simulation optimization under stochastic noise, evolutionary computing, and machine learning.

● ● ●