



Innovative Applications of O.R.

Efficient estimation of a risk measure requiring two-stage simulation optimization

Tianxiang Wang^a, Jie Xu^{b,*}, Jian-Qiang Hu^c, Chun-Hung Chen^b^a Academy of Statistics and Interdisciplinary Sciences, Faculty of Economics and Management, East China Normal University, Shanghai 200062, China^b Department of Systems Engineering & Operations Research, George Mason University, Fairfax, VA 22030, USA^c School of Management, Fudan University, Shanghai 200433, China

ARTICLE INFO

Article history:

Received 20 September 2021

Accepted 13 June 2022

Available online 17 June 2022

Keywords:

Simulation

Two-stage simulation optimization

Risk measure

Optimal computing budget allocation

ABSTRACT

This paper is concerned with the efficient estimation of the risk measure of a system where the estimation requires solving a two-stage simulation optimization problem. The first stage samples risk factors that specify a second stage simulation optimization problem. The second stage solves a simulation optimization problem and outputs the best performance of the system under the realized risk factors, which are then aggregated across all first stage samples to produce an estimate of the risk measure. Applications of such an estimation scheme arise frequently in important industries such as financial, healthcare, logistics, and manufacturing. Because a large number of first stage samples are typically needed, each of which requires solving a computationally expensive simulation optimization problem, the two-stage simulation optimization approach faces a major computational efficiency challenge. In response to this challenge, this paper proposes a sequential simulation budget allocation procedure that determines the allocation of simulation budget based on a score known as revised probability of sign change for each decision under each scenario. The consistency of the proposed procedure is proved and the computational efficiency gain of the proposed is demonstrated using both benchmark test functions and two test cases in the context of financial portfolio risk estimation and healthcare system resilience estimation.

© 2022 Elsevier B.V. All rights reserved.

1. Introduction

Estimating the risk measure of a complex system's operational performance in an uncertain environment is of great importance in a wide spectrum of economic, engineering, financial, and healthcare applications. Stochastic simulations have long been used for this purpose thanks to its ability to generate realizations of risk factors from often times high-dimensional and complex probability distribution models and then to evaluate the performance of the system under the realized risk factors. When the performance evaluation requires solving a simulation optimization problem, the estimation of risk measures takes the form of a two-stage stochastic simulation optimization, with the first stage generating risk factors and the second stage performing simulation optimization under a realized set of risk factors.

One example is the estimation of the risk measure for a hospital's capability to handle arriving patients with life-threatening conditions that require immediate treatment in either an intensive care unit (ICU) or a coronary care unit (CCU). The risk measure

is defined as the probability of a new patient being denied admission because there is no ICU/CCU bed available. Hospital bed shortage is a widespread problem all over the world, especially in China where the average bed utilization in hospitals reached 83.6% in 2019 Bureau (2020a). In Shanghai, it has reached 93.6% in 2019 Bureau (2020b). In the context of the global COVID-19 pandemic, the hospital bed shortage problem becomes ever more urgent and a disturbingly large number of patients who need critical care are being denied admissions.

One way to decrease the probability of denied admission is to better utilize the limited number of hospital beds, such as a novel decentralized admission control system with partial capacity sharing studied empirically by Zhao, Yu, & Hu (2022) and then theoretically by Wang, Yu, & Hu (2021b) based on the operations of the cardiac surgery department at a renowned general hospital in China. However, designing the right admissions control system is highly dependent on the specific operational conditions and of a hospital and may not be easily adopted at a large scale. There is also clearly a capacity limit beyond which no more benefit may be generated.

A more generally applicable and salable approach is capacity expansion, e.g., to add more beds and corresponding services and

* Corresponding author.

E-mail address: jxu13@gmu.edu (J. Xu).

facilities. However, physical space and financial resources are limited. So any capacity expansion projects must be carefully planned and justified, which may be done by the proposed estimation method using two-stage simulation optimization. Suppose that a hospital plans to add K beds to ICU and CCU together. As part of the project approval process, the hospital management needs to quantify the benefit of the expansion as measured by the probability of denied admissions. For simplicity, consider the case where the overall arrival rate of patients remains steady but the proportion of ICU/CCU patients vary. In the first-stage of the analysis, patient arrival profiles ω are generated, which, for example, represent the proportion of ICU patients among all arriving patients, denoted by ω . The number of beds to add to the ICU/CCU unit, denoted by x , out of the total planned K new beds, will help determine the probability of denied admission, denoted as $Y(x; \omega)$. Given the complexity of the patient flow in the hospital, a stochastic simulation model is used to estimate $Y(x; \omega)$. Once ω is observed, the hospital management wants to know the lowest probability of denied admission that can be achieved with different bed allocations. So the second-stage of the analysis is to determine the lowest denied admissions probability $Y^*(\omega)$ by solving the following simulation optimization problem (we use a maximization formulation for the second stage simulation optimization problem in this paper):

$$Y^*(\omega) = \max_{x \in X(\omega)} -Y(x; \omega), \quad (1)$$

where $X(\omega)$ is the feasible decision space and the expectation is taken with respect to the randomness in patient arrivals and treatment processes once a patient is admitted. In this paper, we focus on the case where the decision space is finite. In general, there is no analytical expression for the objective function in problem (1) and it needs to be estimated by taking the sample average of multiple stochastic simulation replications on a decision x under arrival profile ω . Such a problem is known as simulation-based optimization, or simply simulation optimization (Chen & Lee, 2011; Chen, Gao, Chen, & Shi, 2013; Hu, Fu, & Marcus, 2008; Xu et al., 2016).

Given the uncertainty in the arrival profile ω , the risk measure is defined as the probability that the denied admission rate exceeds a pre-specified threshold α_0 :

$$P^* = \mathbb{E}_{\nu_\omega} [\mathbb{I}_{\{Y^*(\omega) > \alpha_0\}}], \quad (2)$$

where $\mathbb{I}_{\{\cdot\}}$ is an indicator function. The expectation is taken with respect to ν_ω , the probability measure of ω . In this paper, we assume that ω 's are generated from a finite scenario set. The estimation of P^* requires a two-stage procedure because samples of risk factors ω need to be first generated following ν_ω and then $Y^*(\omega)$ has to be estimated by solving a simulation optimization problem (1). As to the best of our knowledge, such a two-stage risk measure estimation with the inner level involving a simulation optimization problem has been rarely studied in the literature, whereas there are many important real world applications that require such an approach, such as determining the resilience level of an infrastructure system (Miller-Hooks, Zhang, & Faturechi, 2012; Zhou et al., 2021). In the healthcare setting, this two-stage risk measure estimation may be easily cast into a variety of decision making contexts. For example, when certain types of medical resources, e.g., oxygen tanks, become scarce and have to be rationed among different departments, a two-stage risk measure estimation may be conducted to answer questions such as the impact of varying demands from different departments or the benefit of acquiring new resources.

Simulation optimization is well-known to be computationally challenging because simulations are often time-consuming to execute and many replications of simulations have to be done to control the accuracy of simulation estimates in the presence of stochastic noise in simulation output data. This computational

challenge is further exacerbated in the proposed two-stage simulation optimization approach because there are now many risk factor scenarios for each of which a simulation optimization problem needs to be solved. In this paper, we refer to the first stage simulation where risk factor scenarios ω are generated as outer level simulation and the second stage where simulation optimizations are conducted as inner level simulation.

In the past couple of decades, there has been a rapidly growing body of simulation optimization literature. For problems where $X(\omega)$ has a moderate number of solutions, Ranking & Selection (R&S) methods can be applied (Branke, Chick, & Schmidt, 2007). Widely used methods include the Optimal Computing Budget Allocation (OCBA) type of algorithms (Chen, Lin, Yücesan, & Chick, 2000; Chen, Yücesan, Dai, & Chen, 2009; Gao, Chen, & Shi, 2017; LaPorte, Branke, & Chen, 2012; 2015; Lee, Chew, & Manikam, 2006; Li, Liu, Pedrielli, Lee, & Chew, 2017; Peng, Chen, Fu, & Hu, 2016; Xiao, Lee, & Chen, 2015; Xiao, Lee, Morrice, Chen, & Hu, 2021; Zhang et al., 2016), Bayesian methods (Groves & Branke, 2019; Qu, Ryzhov, Fu, & Ding, 2015), and frequentist approaches such as indifference-zone R&S algorithms (Kim & Nelson, 2001; Luo, Hong, Nelson, & Wu, 2015; Nelson, Swann, Goldsman, & Song, 2001; Teng, Lee, & Chew, 2010; Zhong, Liu, Luo, & Hong, 2021). When $X(\omega)$ is infinite or finite but too large for a direct application of an R&S procedure, a search algorithm can be used to efficiently search the decision space $X(\omega)$ (Chen et al. (2013); Gao & Chen (2016)). See Xu, Huang, Chen, & Lee (2015) for a comprehensive overview of such algorithms. A new line of research in simulation optimization literature is concerned with the selection of the optimal decision $X(\omega)$ upon the observation of ω (Goodwin, Xu, Celik, & Chen, 2022; Goodwin, Xu, Chen, & Celik, 2021; Pedrielli et al., 2019; Thanos, Bastani, Celik, & Chen, 2015). Furthermore, Liu et al. (2019) design algorithms to efficiently identify whether the performance of the best decision exceeds a pre-specified threshold, which is directly related to our second-stage problem. However, these algorithms all focus on solving a single instance of simulation optimization, either without an outer level risk factor scenario ω , or for a specific risk factor scenario ω .

Recent works by Gao, Du, & Chen (2019); Li, Lam, Liang, & Peng (2020); Shen, Hong, & Zhang (2021) on contextual R&S have a two-stage structure, where the first-stage information is referred to as covariates or contexts and the second stage is an R&S problem dependent on the covariates or contexts. Unlike the estimation problem considered in this paper as given in Eq. (2), their focus is still the selection of the best decision. Gao et al. (2019) and Li et al. (2020) propose sampling schemes to maximize the expected probability of correct selection (PCS) or the worst-case PCS across a finite set of scenarios. In Shen et al. (2021), an indifference-zone R&S algorithm is developed for efficiently solving problem (1) when $Y(x, \omega)$ is assumed to be a linear function of x . In comparison, our objective is to efficiently estimate a risk measure defined as the probability that the performance of the best decision on each scenario exceeds a pre-specified threshold.

Two-level simulations, or nested simulations have also been well studied in the simulation literature, often in the context of risk measure estimation (Broadie, Du, & Moallemi, 2011; 2015; Dang, Feng, & Hardy, 2019; Gordy & Juneja, 2010; Hong, Juneja, & Liu, 2017; Lan, Nelson, & Staum, 2010; Liu & Staum, 2010; Sun, Apley, & Staum, 2011; Vidyashankar & Xu, 2013; Zhu, Hale, & Zhou, 2018). According to the formulation in Hong et al. (2017), nested simulation is concerned with the estimation of a quantity $\alpha = \mathbb{E}_{\nu_\omega} [g(\mathbb{E}[Y|\omega])]$. When $g(\cdot)$ is an indicator function, it estimates the probability for a system's performance measure to exceed a given threshold. However, the inner simulation considered in nested simulation literature does not need to perform simulation optimization, and are instead just another layer of Monte Carlo sampling as in the first stage. In Broadie et al. (2011), it was shown

that a sequential approach to allocate simulation budget can lead to a significant computational efficiency gain over algorithms that assign the same number of simulation replications to each risk scenario ω . A similarly efficient sequential simulation budget allocation approach is desirable for problem (2). However, the inner level simulation considered in this paper is simulation optimization, which fundamentally changes the nature of the problem and requires new theoretical and algorithmic development. In Wang, Xu, & Hu (2021a), problem (2) was decoupled and a nested simulation procedure similar to Broadie et al. (2011) was used to select a first stage scenario, and then OCBA was used to solve the second-stage simulation optimization under the selected scenario. This decoupling approach lacks a theoretical foundation and leads to sub-optimal solution. As will be discussed in Section 3, the use of OCBA in the second stage actually leads to wasted simulation budget.

Another related research area is the recent work computing budget allocation in stochastic programming. Fei, Gülpınar, & Branke (2019) considered the allocation of a fixed computing budget to a two-stage linear stochastic program. Xie, Yi, & Zheng (2020) introduced the concept of stochastic programming via simulation (SPvS) and proposed a procedure to solve two-stage stochastic programming problems where the second-stage response function requires simulation estimation, similar to our setting here. Their goal is to optimize a first stage decision variable whereas ours is to efficiently estimate a risk metric, e.g., the probability P^* given in Eq. (2). The procedure developed in this paper may be used to help improve the efficiency of an SPvS algorithm, for example, to improve sampling allocation to improve the estimation accuracy when the problem involves chance constraints.

The main contribution of this paper is a new sequential two-stage simulation budget allocation procedure for efficient estimation of the risk measure P^* as given in (2) that requires two-stage simulation optimization. The key idea is to estimate the impact of an additional simulation allocated to a particular risk factor ω as measured by a quantity known as Revised Probability of Sign Change (RPSC). The consistency of the algorithm is proved and its numerical efficiency is demonstrated through benchmark test functions and two case studies, one in a financial setting and the other in a healthcare setting. Through a theoretical analysis of the procedure, we show that fully solving the second stage simulation optimization problem (1) not only hurts the computational efficiency of the procedure, but, in fact, is unnecessary. Instead of trying to solve the second stage simulation optimization problem (1) in its entirety, the developed new procedure uses second-stage simulation budget only to efficiently determine if the indicator function inside the expectation of (2) is 1 or 0 for a particular scenario ω , which then leads to increased computational efficiency.

The rest of the paper is organized as follows. Section 2 formulates the two-stage simulation optimization budget allocation problem. Section 3 develops the two-stage simulation budget allocation procedure and presents an algorithmic implementation. Section 4 shows the consistency of the proposed algorithm. Section 5 describes test problems and experimental results that demonstrate the efficiency of the proposed algorithm. Section 6 concludes the paper.

2. Problem formulation

Let $\Omega = \{\omega_i, i = 1, \dots, n\}$ denote the set of n first-stage scenarios and $X(\omega_i) = \{x_{ij}, j = 1, \dots, k_i\}$ be the set of k_i decision alternatives that can be taken under scenario ω_i . The subset $\Omega^* = \{\omega : Y^*(\omega_i) > \alpha_0, i = 1, \dots, n\}$ contains scenarios under which the system's performance under the best decision exceeds the pre-specified threshold α_0 . Set $n^* = |\Omega^*|$. Then for the given Ω , we have $P^* = n^*/n$.

Here, $Y^*(\omega_i)$ needs to be determined via simulation optimization. We generate m_{ij} IID observations of $Y_l(x_{ij}; \omega_i)$, $l = 1, \dots, m_{ij}$, and use the sample mean of these m_{ij} simulation observations, $\bar{Y}(x_{ij}; \omega_i)$, to estimate the true system performance $Y(x_{ij}; \omega_i)$, which is subject to noise in stochastic simulation outputs. Then we obtain the estimate

$$\hat{Y}^*(\omega_i) = \max_{j \in \{1, 2, \dots, k_i\}} \bar{Y}(x_{ij}; \omega_i), \quad (3)$$

and then estimate n^* by $\hat{n}^* = \sum_{i=1}^n \mathbb{I}_{\{\hat{Y}^*(\omega_i) > \alpha_0\}}$. We can then estimate P^* by

$$\hat{P}^* = \frac{\hat{n}^*}{n} = \frac{\sum_{i=1}^n \mathbb{I}_{\{\hat{Y}^*(\omega_i) > \alpha_0\}}}{n}. \quad (4)$$

One obvious and also widely used allocation scheme first equally allocates second-stage simulation budget among all scenarios, and then equally allocates the simulation budget among all decisions for each scenario. While equal allocation offers a feasible approach, it is clearly not an efficient way to use simulation budget. To see this point, notice that when solving (2), if we know a scenario ω has performance $Y^*(\omega)$ that is significantly lower or higher than α_0 , it is then no longer important to allocate further simulation budget to this scenario. While intuitive, it requires caution to reap this potential computational savings because one must carefully quantify the uncertainty in $\hat{Y}^*(\omega_i)$. This is more challenging than in a classical simulation estimator because it requires solving the second stage simulation optimization problem (1), which is subject to considerable stochastic error when simulation budget is limited.

It is important to observe that while the second stage involves a simulation optimization, we do not need to solve the simulation optimization to the full extent, e.g., using a simulation optimization algorithm like OCBA. To understand why this observation is important, we consider a case where $\bar{Y}(x_{i1}; \omega_i) > \bar{Y}(x_{i2}; \omega_i) > \alpha_0$. For the purpose of solving problem (2), we only need to make sure that $Y(x_{i1}; \omega_i) > \alpha_0$ and thus can focus more on simulating x_{i1} . However, OCBA would focus on if $Y(x_{i1}; \omega_i) > Y(x_{i2}; \omega_i)$ and thus will keep allocating simulation to both x_{i1} and x_{i2} , and thus potentially waste some simulation budget. We derive a new OCBA method referred to as OCBA-2S for the two-stage problem (2). The new procedure sequentially allocates to a chosen decision for a scenario with the largest revised probability of sign change (RPSC), which will be elaborated shortly.

3. A new computing budget allocation procedure for two-stage simulation optimization

Intuitively, if there is one more simulation that can be done, it should be allocated to a decision under a scenario that would be most likely to cause a change in the estimate of P^* . In Broadie et al. (2011), a quantity referred to as Probability of a Sign Change (PSC) was introduced to quantify the significance of collecting an additional simulation observation from a decision alternative for a particular scenario. PSC is conditional on the filtration up to the current iteration and computed as follows. Given a decision x_{ij} that may be taken under scenario ω_i , we denote by $\bar{Y}'(x_{ij}; \omega_i)$ the new simulation sample average with the additional simulation observation. Depending on whether the previous sample average $\bar{Y}(x_{ij}; \omega_i)$ exceeds α_0 or not, PSC_{ij} is given as follows:

$$PSC_{ij} = \begin{cases} P_{\{\bar{Y}'(x_{ij}; \omega_i) \leq \alpha_0 | \bar{Y}(x_{ij}; \omega_i) > \alpha_0\}} & \text{if } \bar{Y}(x_{ij}; \omega_i) > \alpha_0; \\ P_{\{\bar{Y}'(x_{ij}; \omega_i) > \alpha_0 | \bar{Y}(x_{ij}; \omega_i) \leq \alpha_0\}} & \text{if } \bar{Y}(x_{ij}; \omega_i) \leq \alpha_0. \end{cases} \quad (5)$$

Based on this definition, if we simulate a decision x_{ij} with a high PSC_{ij} value, the chance of a change in the sample estimate $\hat{P}^*$

would also be high. In other words, there is still considerable variability associated with the simulation estimate of the performance of decision x_{ij} under scenario ω_i . With this observation, it seems that a justifiable approach is to simulate the decision with the highest PSC. However, considering that we only need to know if $Y^*(\omega_i)$ exceeds α_0 or not, it is not necessary to consider PSC_{ij} 's. For example, if we can make sure that there is at least one decision, which may not even be necessarily the optimal decision, with a performance $Y(x_{ij}; \omega_i) > \alpha_0$, then we do not need to allocate further simulation budget to other decisions under ω_i . Therefore, we introduce a new metric referred to as Revised-PSC (RPSC) to more accurately measure the effect of an additional simulation sample. A sequential allocation rule can then be introduced that allocates the next simulation sample to a decision with the highest RPSC. We now proceed to elaborate on the expression of RPSC and derive the new sequential simulation budget allocation policy.

First, we derive the expression of PSC_{ij} . Without loss of generality, we consider the case that $\bar{Y}(x_{ij}; \omega_i) > \alpha_0$. If one more simulation is done on x_{ij} , the new sample mean is

$$\begin{aligned}\bar{Y}'(x_{ij}; \omega_i) &= \frac{1}{m_{ij} + 1} \sum_{l=1}^{m_{ij}+1} Y_l(x_{ij}; \omega_i) \\ &= \frac{1}{m_{ij} + 1} Y_{m_{ij}+1}(x_{ij}; \omega_i) + \frac{m_{ij}}{m_{ij} + 1} \bar{Y}(x_{ij}; \omega_i).\end{aligned}$$

Therefore, we have

$$\begin{aligned}PSC_{ij} &= P\{\bar{Y}'(x_{ij}; \omega_i) \leq \alpha_0\} \\ &= P\{Y_{m_{ij}+1}(x_{ij}; \omega_i) - Y(x_{ij}; \omega_i) \\ &\quad \leq -m_{ij}(\bar{Y}(x_{ij}; \omega_i) - \alpha_0) - Y(x_{ij}; \omega_i) + \alpha_0\} \\ &\approx P\{Y_{m_{ij}+1}(x_{ij}; \omega_i) - Y(x_{ij}; \omega_i) \leq -m_{ij}|\bar{Y}(x_{ij}; \omega_i) - \alpha_0|\} \\ &\leq \left(1 + \frac{m_{ij}^2}{\sigma_{ij}^2} |\bar{Y}(x_{ij}; \omega_i) - \alpha_0|^2\right)^{-1}.\end{aligned}\quad (6)$$

In the above derivation, we invoke the assumption that $m_{ij} \gg 1$ to obtain the approximation, i.e.,

$$-m_{ij}(\bar{Y}(x_{ij}; \omega_i) - \alpha_0) - (Y(x_{ij}; \omega_i) - \alpha_0) \approx -m_{ij}|\bar{Y}(x_{ij}; \omega_i) - \alpha_0|.$$

The last inequality in (6) is a result of applying the one-sided Chebyshev inequality, where σ_{ij}^2 is the variance of the simulation output of decision x_{ij} under scenario ω_i .

A similar derivation for the case $\bar{Y}(x_{ij}; \omega_i) \leq \alpha_0$ leads to the same upper bound for PSC_{ij} :

$$PSC_{ij} \leq APSC_{ij} = \left(1 + \frac{m_{ij}^2}{\sigma_{ij}^2} |\bar{Y}(x_{ij}; \omega_i) - \alpha_0|^2\right)^{-1}.\quad (7)$$

We refer to this upper bound for PSC as the Approximate Probability of Sign Change (APSC), which can be easily computed. To show the efficiency of the approximation, we offer an alternative justification about it. If the simulation output drawn from a location-scale family of distributions, e.g., if $Y_{m_{ij}+1}(x_{ij}; \omega_i)$ is normally distributed, then we have

$$PSC_{ij} \quad (8)$$

$$\begin{aligned}&= P\{Y_{m_{ij}+1}(x_{ij}; \omega_i) - Y(x_{ij}; \omega_i) \\ &\quad \leq -m_{ij}(\bar{Y}(x_{ij}; \omega_i) - \alpha_0) - Y(x_{ij}; \omega_i) + \alpha_0\}\end{aligned}\quad (9)$$

$$\approx P\{Y_{m_{ij}+1}(x_{ij}; \omega_i) - Y(x_{ij}; \omega_i) \leq -m_{ij}|\bar{Y}(x_{ij}; \omega_i) - \alpha_0|\}\quad (10)$$

$$= \Phi\left(\frac{-m_{ij}|\bar{Y}(x_{ij}; \omega_i) - \alpha_0|}{\sigma_{ij}}\right)\quad (11)$$

where $\Phi(x)$ is the cumulative distribution function of the standard normal distribution. Notice that $APSC_{ij} = (1 + \frac{m_{ij}^2}{\sigma_{ij}^2} |\bar{Y}(x_{ij}; \omega_i) - \alpha_0|^2)^{-1}$, therefore, maximizing the PSC_{ij} according to (11) is equivalent to maximizing $APSC_{ij}$.

Next, we compute the RPSC for all second stage decisions x_{ij} . For notational simplicity, we let $J_i = \{j \in \{1, 2, \dots, k_i\} | Y(x_{ij}; \omega_i) > \alpha_0\}$ and $|J_i|$ is the cardinality of the set J_i . In addition, $\hat{J}_i = \{j \in \{1, 2, \dots, k_i\} | \bar{Y}(x_{ij}; \omega_i) > \alpha_0\}$ is the estimate of J_i . We also denote by $b_i = \arg \max_{j \in \{1, 2, \dots, k_i\}} Y(x_{ij}; \omega_i)$ the index for the optimal decision under scenario ω_i , and the index for the estimated optimal decision under scenario ω_i by $\hat{b}_i = \arg \max_{j \in \{1, 2, \dots, k_i\}} \bar{Y}(x_{ij}; \omega_i)$.

We divide scenarios into three categories based on the value of $|J_i|$, and calculate the RPSC for each decision accordingly.

- $|J_i| = 0$: Because $\bar{Y}(x_{ij}; \omega_i) < \alpha_0$ for all x_{ij} , $\mathbb{I}_{\{\bar{Y}(x_{\hat{b}_i}; \omega_i) \geq \alpha_0\}}$ is subject to change with the addition of one sample for each decision, i.e. $\bar{Y}'(x_{ij}; \omega_i) > \alpha_0$, which will change the estimate of P^* . We thus have

$$RPSC_{ij} = APSC_{ij} = \left(1 + \frac{m_{ij}^2}{\sigma_{ij}^2} |\bar{Y}(x_{ij}; \omega_i) - \alpha_0|^2\right)^{-1}.\quad (12)$$

- $|J_i| = 1$: In this case, we have $\bar{Y}(x_{\hat{b}_i}; \omega_i) > \alpha_0$ and $\bar{Y}(x_{ij}; \omega_i) \leq \alpha_0$ for $j \neq \hat{b}_i$. If the additional simulation is allocated to a decision $j \neq \hat{b}_i$, the updated $\bar{Y}'(x_{\hat{b}_i}; \omega_i)$ will always be bigger than α_0 , which means that the additional simulation will not change our estimate of P^* unless it is allocated to $x_{\hat{b}_i}$. Therefore, $RPSC_{ij}$ is defined as:

$$RPSC_{ij} = \begin{cases} \left(1 + \frac{m_{ij}^2}{\sigma_{ij}^2} |\bar{Y}(x_{ij}; \omega_i) - \alpha_0|^2\right)^{-1} & \text{if } j = \hat{b}_i; \\ 0 & \text{if } j \neq \hat{b}_i. \end{cases}\quad (13)$$

- $|J_i| > 1$: There are multiple decisions x_{ij} satisfying $\bar{Y}(x_{ij}; \omega_i) > \alpha_0$. If the additional simulation is allocated to a decision $j \notin \hat{J}_i$, $\bar{Y}'(x_{\hat{b}_i}; \omega_i)$ will not change. In fact, to cause a change in $\mathbb{I}_{\{\bar{Y}(x_{\hat{b}_i}; \omega_i) \geq \alpha_0\}}$, at least $|\hat{J}_i|$ samples need to be allocated, with one to each of the $|\hat{J}_i|$ decisions, and these $|\hat{J}_i|$ updated estimates need to be smaller than α_0 . So $\mathbb{I}_{\{\bar{Y}(x_{\hat{b}_i}; \omega_i) \geq \alpha_0\}}$ changes with a probability of $\prod_{j \in \hat{J}_i} APSC_{ij}$. To facilitate the comparison of $RPSC_{ij}$'s derived for the previous two categories with just one additional simulation sample being allocated, we define RPSCs in this case as follows:

$$RPSC_{ij} = \begin{cases} \frac{APSC_{ij}}{\sum_{j \in \hat{J}_i} APSC_{ij}} \prod_{j \in \hat{J}_i} APSC_{ij} & \text{if } j \in \hat{J}_i; \\ 0 & \text{if } j \notin \hat{J}_i. \end{cases}\quad (14)$$

We need to point out that a weight computed from relative $APSC_{ij}$ is not necessarily the best. We only need to make sure that an additional simulation can be allocated to the most critical decisions in J_i , i.e. the decision has the highest $APSC_{ij}$. In fact, one could adopt a weight computed from the logarithmic $APSC_{ij}$.

Based on the computed $RPSC_{ij}$, we propose OCBA-2S, a new simulation budget allocation rule for the efficient estimation of P^* . OCBA-2S selects a scenario and a decision that has the largest RPSC

to receive the next simulation:

$$(i^*, j^*) \in \arg \max_{(i,j)} RPSC_{ij}. \quad (15)$$

Algorithm 1 presents the new OCBA-2S simulation budget allocation rule that will be tested in the experiments reported in Section 5.

Algorithm 1 (OCBA-2S simulation budget allocation).

INPUT: $\Omega = \{\omega_i, i = 1, \dots, n\}$, $X(\omega_i) = \{x_{ij}, j = 1, \dots, k_i\}$, total second stage simulation budget T , the number of initial simulation replications m_0 .

INITIALIZE:

Perform m_0 simulation replications for $x_{ij}, i = 1, \dots, n, j = 1, \dots, k_i$ and record $Y_l(x_{ij}; \omega_i), l = 1, \dots, m_0$; let $m_{ij} = m_0$.

LOOP: WHILE $\sum_{i=1}^n \sum_{j=1}^{k_i} m_{ij} < T$ **DO**

UPDATE:

Estimate, Allocate and Simulate

- 1: Compute sample variance $\hat{\sigma}_{ij}^2$, sample mean $\bar{Y}(x_{ij}; \omega_i)$, for all $i = 1, \dots, n, j = 1, \dots, k_i$;
- 2: Calculate $RPSC_{ij}$ for all $i = 1, \dots, n, j = 1, \dots, k_i$ using equations (12)–(14).
- 3: Collect an additional simulation sample on scenario ω_{i^*} and decision $x_{i^*j^*}$, where (i^*, j^*) is given by (15), and set $m_{i^*j^*} \leftarrow m_{i^*j^*} + 1$.

END OF LOOP

Output:

Return $\hat{P}^* = \sum_{i=1}^n \mathbb{I}_{\{\bar{Y}(x_{i\hat{j}_i}; \omega_i) > \alpha_0\}} / n$.

4. Analysis of algorithm

In this section, we analyze the asymptotic behavior of the algorithm and show the consistency of the estimate. Since the second-stage simulation is stochastic, there is an induced probability distribution over the set of all sample paths. We denote by $P[\cdot]$ and $E[\cdot]$ the probability and expectation taken with respect to this distribution. The convergence of sequences of random events is to be understood with respect to P . We also define $\mathcal{F}_t$ to be the σ -field generated by the set of sampled decisions up to iteration t and their stochastic simulation output up to iteration $t - 1$, i.e., $\mathcal{F}_t := \sigma\{x(1), y(1), \dots, x(t-1), y(t-1), x(t)\}$, where $x(t)$ is the decision simulated at iteration t and $y(t)$ is the corresponding simulation output. We use $m_{ij}(t)$, $\bar{Y}_{m_{ij}(t)}(x_{ij}; \omega_i)$ and $\hat{\sigma}_{ij}(m_{ij}(t))$ to denote the number of generated samples, estimate sample mean and sample variance under scenario ω_i by taking decision x_{ij} in iteration t . In addition, recall that $J_i = \{j \in \{1, 2, \dots, k_i\} | Y(x_{ij}; \omega_i) > \alpha_0\}$ and $|J_i|$ is the cardinality of the set J_i . To show that our estimate $\hat{P}^*$ is consistent, we first prove the following lemmas.

Lemma 4.1. For any scenario ω_i , if $\lim_{T \rightarrow \infty} \sum_{j=1}^{k_i} m_{ij}(T) = \infty$, then

- (1): if $|J_i| = 0$, then for any $j \in \{1, 2, \dots, k_i\}$, $\lim_{T \rightarrow \infty} m_{ij}(T) = \infty$ w.p.1;
- (2): if $|J_i| = 1$, then for $j^* \in J_i$, $\lim_{T \rightarrow \infty} m_{ij^*}(T) = \infty$ w.p.1;
- (3): if $|J_i| > 1$, then there is at least one $j \in J_i$, s.t. $\lim_{T \rightarrow \infty} m_{ij}(T) = \infty$ w.p.1.

Proof 4.1. First, we denote by A_i the set of decisions receiving an infinite number of replications as $T \rightarrow \infty$, i.e., $A_i = \{j \in \{1, 2, \dots, k_i\} | \lim_{T \rightarrow \infty} m_{ij}(T) = \infty\}$, and $B = \{1, 2, \dots, k_i\} \setminus A_i$. To prove the above three conclusions, it suffices to show that if $A_i \subseteq \{1, 2, \dots, k_i\} \setminus J_i$, then $B_i = \emptyset$. Suppose B_i is not empty, then for any $j \in B_i$, we have $\lim_{T \rightarrow \infty} m_{ij}(T) = M_{ij} < \infty$. In addition, there exists a sufficiently large T_1 , s.t. $\forall j \in B_i, T > T_1, m_{ij}(T) = M_{ij}$. Let

$\hat{j} = \arg \min_{j \in B_i} APSC_{ij}(M_{ij})$. Noticing that $\lim_{T \rightarrow \infty} \sum_{j=1}^{k_i} m_{ij}(T) = \infty$ implies that A_i is not empty. For any $j \in A_i$, according to the Strong Law of Large Numbers (SLLN), $\bar{Y}_{m_{ij}(T)}(x_{ij}; \omega_i)$ converges to $Y(x_{ij}; \omega_i)$ and $\hat{\sigma}_{ij}(m_{ij}(T))$ converges to σ_{ij} with probability one. Furthermore,

$$\begin{aligned} & \lim_{T \rightarrow \infty} APSC_{ij}(m_{ij}(T)) \\ &= \lim_{T \rightarrow \infty} \left(1 + \frac{m_{ij}^2(T)}{\hat{\sigma}_{ij}^2(m_{ij}(T))} |\bar{Y}_{m_{ij}(T)}(x_{ij}; \omega_i) - \alpha_0| \right)^{-1} = 0 \end{aligned}$$

w.p.1. Therefore, we can always find a sufficiently large T_2 ($T_2 > T_1$), s.t. $\forall T > T_2, APSC_{ij}(m_{ij}(T)) < APSC_{\hat{j}}(M_{\hat{j}})$ and $\bar{Y}_{m_{ij}(T)}(x_{ij}; \omega_i) \leq \alpha_0$.

Considering that $\lim_{T \rightarrow \infty} \sum_{j=1}^{k_i} m_{ij}(T) = \infty$, there exists a large T_3 ($T_3 > \max\{T_1, T_2\}$) and in iteration T_3 , the simulation budget has to be allocated to the decision $x_{i\hat{j}'}$ with the largest $RPSC_{ij}(m_{ij}(T_3))$ on scenario ω_i . Obviously, $\hat{j}'$ belongs to B_i , which means that $m_{i\hat{j}'}(T_3) = M_{i\hat{j}'} + 1$. This contradicts $\lim_{T \rightarrow \infty} m_{i\hat{j}'}(T) = M_{i\hat{j}'}$. $\square$

Lemma 4.1 shows that as $T \rightarrow \infty$ the proposed OCBA-2S algorithm allocates an infinite number of simulations to decisions that help determine if $Y^*(\omega) > \alpha_0$ or not for any scenario ω . It should be noted that this is different from requiring the number of simulations executed on each decision under a scenario to go to infinity as $T \rightarrow \infty$. Instead, only the decisions that help determine if $Y^*(\omega) > \alpha_0$ or not need to be simulated infinitely many times. This should be contrasted with using OCBA to solve the second-stage simulation, which would drive the number of simulations allocated to every decision to infinity in the limit in order to correctly identify $x^*(\omega)$. Inevitably, this leads to a great deal of waste on simulation budget. The following proposition formally establishes this property of OCBA-2S, which explains from one perspective why OCBA-2S can significantly outperform PSC-OCBA as demonstrated by our numerical examples in Section 5.

Proposition 4.2. For any scenario ω_i satisfying $|J_i| > 1$ and $j \in \{1, 2, \dots, k_i\} \setminus J_i$, if $Y(x_{ij}; \omega_i) < \alpha_0$, then $\lim_{T \rightarrow \infty} m_{ij}(T) < \infty$ w.p.1.

Proof 4.2. We prove the conclusion by contradiction. Suppose that there exists a $j' \in \{1, 2, \dots, k_i\} \setminus J_i$, s.t. $\lim_{T \rightarrow \infty} m_{ij'}(T) = \infty$. According to the SLLN, $\bar{Y}_{m_{ij'}(T)}(x_{ij'}; \omega_i)$ converges to $Y(x_{ij'}; \omega_i)$ ($Y(x_{ij'}; \omega_i) < \alpha_0$) w.p.1. Hence, there exists a large T_1 , s.t. $\bar{Y}_{m_{ij'}(T)}(x_{ij'}; \omega_i) < \alpha_0$ holds for any $T \geq T_1$. In addition, from **Lemma 4.1**, there exists a $j^* \in J_i$, such that $\lim_{T \rightarrow \infty} m_{ij^*}(T) = \infty$. Furthermore, we can also find a sufficiently large $T_2 \geq T_1$, such that $\bar{Y}_{m_{ij^*}(T)}(x_{ij^*}; \omega_i) > \alpha_0$ and $RPSC_{ij^*}(m_{ij^*}(T)) > 0$ hold for any $T \geq T_2$. It means that $RPSC_{ij^*}(m_{ij^*}(T)) > RPSC_{ij'}(m_{ij'}(T)) = 0$ holds for any $T \geq T_2$. Therefore, according to the OCBA-2S algorithm, we have $\lim_{T \rightarrow \infty} m_{ij}(T) < \infty$, which leads to a contradiction. $\square$

Lemma 4.3. For any scenario ω_i , if $\lim_{T \rightarrow \infty} \sum_{j=1}^{k_i} m_{ij}(T) = \infty$, then

$$P \left\{ \lim_{T \rightarrow \infty} \mathbb{I}_{\{\bar{Y}_{m_{i\hat{j}_i}(T)}(x_{i\hat{j}_i}; \omega_i) > \alpha_0\}} = \mathbb{I}_{\{Y^*(\omega_i) > \alpha_0\}} \right\} = 1.$$

Proof 4.3. We first consider the scenario ω_i satisfying $Y^*(\omega_i) > \alpha_0$. According to **Lemma 4.1** and the SLLN, there exist a $\hat{j} \in J_i$, and $\bar{Y}_{m_{i\hat{j}}(T)}(x_{i\hat{j}}; \omega_i)$ converges with probability one to $Y(x_{i\hat{j}}; \omega_i)$ ($Y(x_{i\hat{j}}; \omega_i) > \alpha_0$). Hence, we have

$$\begin{aligned} & P \left\{ \lim_{T \rightarrow \infty} \mathbb{I}_{\{\bar{Y}_{m_{i\hat{j}_i}(T)}(x_{i\hat{j}_i}; \omega_i) > \alpha_0\}} = \mathbb{I}_{\{Y^*(\omega_i) > \alpha_0\}} \right\} \\ &= P \left\{ \lim_{T \rightarrow \infty} \bar{Y}_{m_{i\hat{j}_i}(T)}(x_{i\hat{j}_i}; \omega_i) > \alpha_0 \right\} \\ &\geq P \left\{ \lim_{T \rightarrow \infty} \bar{Y}_{m_{i\hat{j}}(T)}(x_{i\hat{j}}; \omega_i) > \alpha_0 \right\} = 1. \end{aligned} \quad (16)$$

Next, we consider the other case with $Y^*(\omega_i) \leq \alpha_0$. According to Lemma 4.1 and SLLN, for any $j \in \{1, 2, \dots, k_i\}$, $\bar{Y}_{m_{ij}(T)}(x_{ij}; \omega_i)$ converges to $Y(x_{ij}; \omega_i)$ ($Y(x_{ij}; \omega_i) \leq \alpha_0$) with probability one. Therefore, we have

$$\begin{aligned} & P\left\{\lim_{T \rightarrow \infty} \mathbb{I}_{\{\bar{Y}_{m_{ij}(T)}(x_{ij}; \omega_i) > \alpha_0\}} = \mathbb{I}_{\{Y^*(\omega_i) > \alpha_0\}}\right\} \\ &= P\left\{\lim_{T \rightarrow \infty} \bar{Y}_{m_{ij}(T)}(x_{ij}; \omega_i) \leq \alpha_0\right\} \\ &= P\left\{\bigcap_{j=1}^{k_i} \lim_{T \rightarrow \infty} \bar{Y}_{m_{ij}(T)}(x_{ij}; \omega_i) \leq \alpha_0\right\} = 1. \end{aligned} \quad (17)$$

Combining (16) and (17), Lemma 2 holds. $\square$

Lemma 4.3 shows that we can correctly determine if the system performance under any scenario exceeds the threshold or not as long as we allocate enough second stage simulation budget to the scenario.

Lemma 4.4. For any scenario ω_i , if $\lim_{T \rightarrow \infty} \sum_{j=1}^{k_i} m_{ij}(T) = \infty$, then w.p.1.

$$\lim_{T \rightarrow \infty} \max_{j \in \{1, 2, \dots, k_i\}} RPSC_{ij}(m_{ij}(T)) = 0. \quad (18)$$

Proof 4.4. We first consider a scenario ω_i satisfying $|J_i| = 0$, i.e. $Y^*(\omega_i) \leq \alpha_0$. By Lemma 4.1 and the SLLN, for any $j \in \{1, 2, \dots, k_i\}$, $APSC_{ij}(m_{ij}(T))$ converges to 0 with probability one. Furthermore, when T is sufficiently large, $RPSC_{ij}(m_{ij}(T)) = APSC_{ij}(m_{ij}(T))$. Therefore, with probability one, we have

$$\lim_{T \rightarrow \infty} \max_{j \in \{1, 2, \dots, k_i\}} RPSC_{ij}(m_{ij}(T)) = 0.$$

For the other case with $|J_i| \geq 1$, i.e. $Y^*(\omega_i) > \alpha_0$, according to Lemma 4.1 and the SLLN, there exists a $\hat{j} \in A_i \cap J_i$, such that $APSC_{\hat{j}}(m_{\hat{j}}(T))$ converges to 0 with probability one, and $\lim_{T \rightarrow \infty} \bar{Y}_{m_{ij}(T)}(x_{ij}; \omega_i) = Y(x_{ij}; \omega_i) > \alpha_0$, which means that $\bar{Y}_{m_{ij}(T)}(x_{ij}; \omega_i) > \alpha_0$ always holds for a sufficiently large T . In addition, for any $j \in A_i$, $APSC_{ij}(m_{ij}(T))$ also converges to 0 with probability one. On the other hand, for any $j \in B_i$, we claim that $\bar{Y}_{m_{ij}(T)}(x_{ij}; \omega_i) \leq \alpha_0$ holds. Otherwise, similar to the proof of Lemma 4.1, we can also show that an additional simulation will be allocated to some decision x_{ij} , $j \in B_i$, and this will lead to a contradiction. Hence, the $RPSC_{ij}(m_{ij}(T))$ for $j \in B_i$ converges to 0 with probability one. Therefore, $\lim_{T \rightarrow \infty} \max_{j \in \{1, 2, \dots, k_i\}} RPSC_{ij}(m_{ij}(T)) = 0$ also holds. $\square$

Lemma 4.5. For any scenario ω_i , under the OCBA-2S algorithm presented in Algorithm 1,

$$\lim_{T \rightarrow \infty} \sum_{j=1}^{k_i} m_{ij}(T) = \infty \quad \text{w.p.1.}$$

Proof 4.5. We prove the conclusion by contradiction. Suppose that the set

$$C = \left\{ i \in \{1, 2, \dots, n\} \mid \lim_{T \rightarrow \infty} \sum_{j=1}^{k_i} m_{ij}(T) = \sum_{j=1}^{k_i} M_{ij} < \infty \right\}$$

is not empty. We can always find a large T_1 , s.t. $\forall i \in C, j \in \{1, 2, \dots, k_i\}, m_{ij}(T) = M_{ij}$ holds. Let $\epsilon = \max_{i \in C, j \in \{1, 2, \dots, k_i\}} RPSC_{ij}(M_{ij})$ and denote the corresponding scenario as ω_{i^*} . From Lemma 4.4, we know that there exists a sufficiently large T_2 with $T_2 > T_1$, such that $\forall T > T_2, i \in \{1, 2, \dots, n\}, j \in \{1, 2, \dots, k_i\}, RPSC_{ij}(m_{ij}(T)) < \epsilon$ holds. Thus, there exists a large T_3 with $T_3 > T_2$, such that in iteration T_3 ,

the simulation budget has to be allocated to scenario ω_{i^*} , i.e. $\sum_{j=1}^{k_i} m_{i^*j}(T_3) = \sum_{j=1}^{k_i} M_{i^*j} + 1$, which leads to a contradiction. $\square$

Lemma 4.5 ensure that the proposed OCBA-2S algorithm allocates enough budget to each scenario when there is a sufficiently large total budget T . Now we are ready to present the main result on the consistency of OCBA-2S.

Theorem 4.6. The OCBA-2S algorithm presented in Algorithm 1 is strongly consistent, that is,

$$P\left(\lim_{T \rightarrow \infty} \hat{P}^* = P^*\right) = 1. \quad (19)$$

Proof 4.6. The result follows from Lemma 4.3 and Lemma 4.5:

$$\begin{aligned} & P\left(\lim_{T \rightarrow \infty} \hat{P}^* = P^*\right) \\ &= P\left(\lim_{T \rightarrow \infty} \sum_{i=1}^n \mathbb{I}_{\{\bar{Y}_{m_{ij}(T)}(x_{ij}; \omega_i) > \alpha_0\}} = \sum_{i=1}^n \mathbb{I}_{\{Y^*(\omega_i) > \alpha_0\}}\right) \\ &= P\left(\sum_{i=1}^n \left(\lim_{T \rightarrow \infty} \mathbb{I}_{\{\bar{Y}_{m_{ij}(T)}(x_{ij}; \omega_i) > \alpha_0\}} - \mathbb{I}_{\{Y^*(\omega_i) > \alpha_0\}}\right) = 0\right) \\ &\geq P\left(\bigcap_{i=1}^n \left(\lim_{T \rightarrow \infty} \mathbb{I}_{\{\bar{Y}_{m_{ij}(T)}(x_{ij}; \omega_i) > \alpha_0\}} = \mathbb{I}_{\{Y^*(\omega_i) > \alpha_0\}}\right)\right) = 1. \end{aligned}$$

Therefore, we have $P\{\lim_{T \rightarrow \infty} \hat{P}^* = P^*\} = 1$. $\square$

5. Numerical experiments

We evaluate the performance of the new OCBA-2S procedure via comparisons with five other procedures in numerical experiments. The first procedure is equal allocation (EA), which equally allocates the simulation budget to every decision and every scenario. Therefore, we have under EA $m_{ij} = T / \sum_{i=1}^n k_i$ for all i and j . The second procedure equally allocates simulation budget to all scenarios, but then uses the well-known OCBA method to determine the allocation among the decisions within each scenario. So this procedure is referred to as EA-OCBA in the following. The third procedure equally allocates simulation budget to all scenarios, but then uses the method from Liu et al. (2019) which aims to correctly identify whether the performance under the best decision exceeds a threshold. We refer to it as the EA-NkOCBA procedure. The fourth procedure is the algorithm proposed in Gao et al. (2019), which we refer to as cOCBA (covariate OCBA). cOCBA aims to maximize the PCS for all second stage problems across all covariate values (scenarios). The fifth procedure is the algorithm proposed in Wang et al. (2021a). We refer to it as the PSC-OCBA procedure because it first selects a scenario using PSC and then uses OCBA to select the decision to simulate.

For each of the five algorithms tested here, we present the estimation bias achieved by an allocation policy as a function of the number of simulation samples allocated. Because of the randomness in simulation samples, we conducted each experiment 1000 times using different random number seeds, and take the average of the bias recorded in these 1000 experiments as an estimate of the allocation policy's bias.

5.1. Experiment 1: A benchmark test function

We first test the performance of different procedures on a benchmark test function given below:

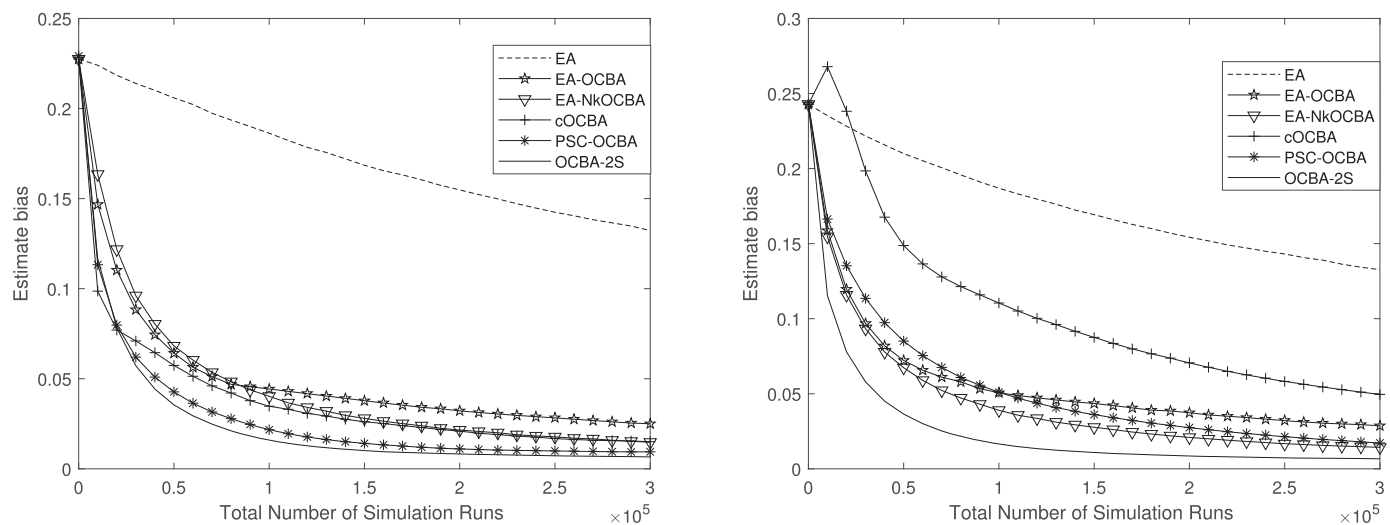


Fig. 1. Estimation bias as a function of the total number of simulation samples for $P^* = 0.9$ (left) and $P^* = 0.1$ (right) for test function (20).

$$Y(x_{ij}; \omega_i) = 0.45 + \frac{i}{5000} - \frac{j-1}{5} + U\left(-\frac{1}{2}, \frac{1}{2}\right),$$

$$i \in \{1, 2, \dots, 500\}, j \in \{1, 2, \dots, 20\}. \quad (20)$$

This function is based on a benchmark test function first reported in Shen et al. (2021). In our experiment, we generated 500 scenarios and set $Y^*(\omega_i) \in [0.45 + 1/5000, 0.45 + 2/5000, \dots, 0.55]$. Each scenario has twenty decision alternatives, with x_{i1} being the best, i.e., $Y^*(\omega_i) = Y(x_{i1}; \omega_i)$. For other decisions x_{ij} , $j = 2, 3, \dots, 20$, we set $Y(x_{ij}; \omega_i) = Y(x_{i1}; \omega_i) - (j-1)/5$. A simulation noise following a uniform distribution $U(-\frac{1}{2}, \frac{1}{2})$ is added each time a simulation observation is collected. In each experiment, every procedure conducted an initial simulation sampling with $m_0 = 10$ on all decisions for every scenario. A total of 300,000 was then expended following the allocation determined by the procedure. At the end of each simulation, we computed the bias of the estimate $\hat{P}^*$ returned by each of the six procedures being tested. We repeated this experiment independently 1000 times and recorded the average bias. Fig. 1 below plots the estimation bias as a function of the total number of simulation samples allocated for two threshold values $\alpha_0 = 0.46$ and $\alpha_0 = 0.54$, which are the 10% and 90% quantiles of $Y^*(\omega)$. So the probabilities we want to estimate are $P^* = 0.9$ and $P^* = 0.1$, respectively.

As can be seen from Fig. 1, in general, OCBA-2S clearly outperforms EA, EA-OCBA, EA-NkOCBA, cOCBA and PSC-OCBA. For example, Fig. 1 shows that for $P^* = 0.1$, after exhausting the total simulation budget of 300,000 simulations, EA, EA-OCBA, EA-NkOCBA, cOCBA and PSC-OCBA reduced the estimation bias to 0.132, 0.029, 0.014, 0.049 and 0.017, all of which are much larger than a bias of 0.007 achieved by OCBA-2S. Another way to understand the computational efficiency gain of OCBA-2S is to examine the *speed-up factor*, which is calculated as the ratio of the total simulation budget to the number of simulations used by OCBA-2S to reach the same level of bias achieved by the other policy after all simulation budget was expended. For example, EA achieved a bias of 0.132 with 300,000 simulations. In comparison, OCBA-2S only used about 10,000 simulations. Thus the speed-up factor is 30. The speed-up factors of OCBA-2S vs. EA-OCBA, EA-NkOCBA, cOCBA and PSC-OCBA are 4.3, 2.5, 7.5 and 3.0, respectively. We have similar observations for $P^* = 0.9$.

We also make the following observations on the performance of various algorithms tested. First, cOCBA performs very differently in the two cases for $P^* = 0.1$ and $P^* = 0.9$ as can be very clearly seen in Fig. 1. To understand why this happens, we first consider a

scenario ω with $Y^*(\omega) < \alpha_0$. We argue it would require much less simulation effort to correctly determine if $Y^*(\omega)$ exceeds α_0 than for a scenario with $Y^*(\omega) > \alpha_0$. To see this point, consider a sub-optimal decision $X \neq X^*(\omega)$. If X is incorrectly selected as the best decision because simulation noise makes $\tilde{Y}(X^*(\omega); \omega) < \tilde{Y}(X; \omega)$, we know $Y(X; \omega) < Y^*(\omega) < \alpha_0$, and thus it would likely take less simulation effort to determine $Y(X; \omega) < \alpha_0$. In other words, an incorrect selection of the best decision could more easily lead to a correct estimation of the indicator function in Eq. (2). In contrast, when only the best decision's response exceeds the threshold, e.g., only $Y^*(\omega) > \alpha_0$, incorrect selection of the best decision would very likely lead to an incorrect estimate of the indicator function in Eq. (2), and thus more simulation budget would be needed for such scenarios. In the case with $P^* = 0.1$, there are 450 scenarios with $Y^*(\omega) < \alpha_0$ and thus would actually need much less simulation effort to correctly determine if $Y^*(\omega) < \alpha_0$ or not. However, because cOCBA maximizes the PCS for all scenarios, cOCBA wastes much budget on these 450 scenarios, which hurts its efficiency. However, in the case $P^* = 0.9$, there are 450 scenarios with $Y^*(\omega) > \alpha_0$. For these scenarios, maximizing PCS directly helps with determining if $Y^*(\omega) > \alpha_0$. Therefore, cOCBA performs much better in this case.

We also notice that although EA-NkOCBA specializes in determining whether $Y^*(\omega) > \alpha_0$ for each ω in the second stage, it is still dominated by OCBA-2S because EA-NkOCBA does not consider budget allocation across scenarios. Finally, we notice in the case of $P^* = 0.9$, when the simulation budget is relative small, cOCBA outperforms OCBA-2S. As will be seen in Fig. 2 with increased simulation noise variance, the lead of cOCBA grows even larger. This makes us believe that the cause is the sensitivity of OCBA-2S to the quantity $RPCS_{ij}$, which is computed using sample statistics and is thus susceptible to simulation noise, especially in the early stage of the algorithm.

Because all algorithms except EA use sample statistics to sequentially determine allocation, we also test their sensitivities to simulation noise by making the noise follow a uniform distribution $U(-\frac{5}{2}, \frac{5}{2})$, which has a variance 25 times of that in the previous set of experiments. Results are shown in Fig. 2. In general, OCBA-2S outperforms the other five methods as long as the total budget is not too limited. For the case $P^* = 0.9$, as discussed previously, cOCBA's allocation aims to ensure high values of PCS across all scenarios, which leads to more favorable outcomes. The high noise makes this benefit even more noticeable here and thus cOCBA maintains the lead over OCBA-2S until there is a reasonably

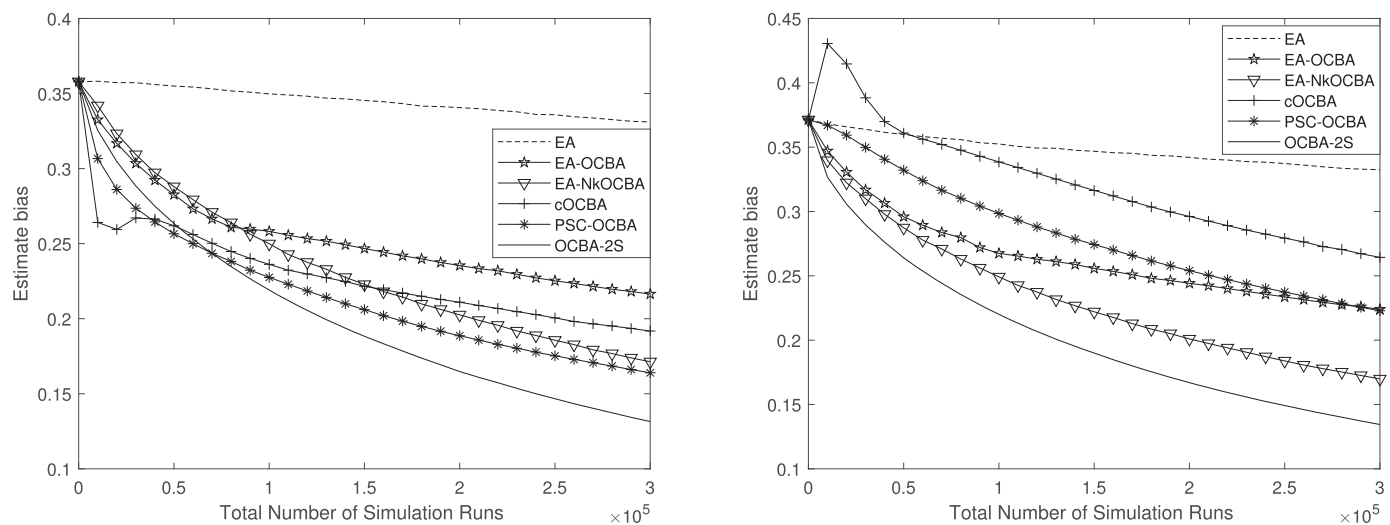


Fig. 2. Estimation bias as a function of the total number of simulation samples for $P^* = 0.9$ (left) and $P^* = 0.1$ (right) under high variance.

large budget. As the expended simulation budget increases, OCBA-2S again outperforms all other five methods.

5.2. Experiment 2: Risk analysis of a put option portfolio

The test problem in Experiment 2 is a put option example from Broadie et al. (2011). Specifically, we assume that the portfolio consists of three long position in a single put option. We are interested in computing the probability of a loss by taking the worst decision. We consider loss thresholds corresponding to 10% loss probabilities. The underlying asset follows a geometric Brownian motion with an initial price of $S_0 = 100$. The drift of this process under the real-world distribution used in the outer stage of simulation is $\mu = 8\%$. The annualized volatility is $\sigma = 20\%$. The risk-free rate is $r = 3\%$. The strike of the put option is $K = 95$, and the maturity is $T = 0.25$ years (i.e., three months). The risk horizon is $\tau = 1/52$ years (i.e., one week). With these parameters, the initial value of the put is $X_0 = 1.699$ given by the Black-Scholes formula. Denote by $S_\tau(\omega)$ the underlying asset price at the risk horizon τ . This random variable is generated according to $S_\tau(\omega) \triangleq S_0 e^{(\mu - \sigma^2/2)\tau + \sigma\sqrt{\tau}\omega}$, where the real-valued risk factor ω is a discrete random variable with equal probability. We assume that at the risk horizon τ , the investor have a chance to sell j ($j \in \{0, 1, 2, 3\}$) of his options. In addition, we assume the difference between the theoretical and real price of a single option is constant, i.e.

$$X_\tau^i(\omega) - X_\tau^j(\omega) = X_\tau^i(\omega) - E[e^{-r(T-\tau)} \max(K - S_\tau(\omega, W), 0) | \omega] \equiv \alpha_0,$$

which means that investors are all positive about the put option. Therefore, the portfolio loss at the risk horizon τ on scenario ω and decision j is given by

$$L(\omega, j) = 3X_0 - 3E[e^{-r(T-\tau)} \max(K - S_\tau(\omega, W), 0) | \omega] - ja_0,$$

where the expectation is taken over the random variable W , which is an independently distributed standard normal, and $S_\tau(\omega, W)$ is given by

$$S_\tau(\omega, W) \triangleq S_\tau(\omega)e^{(r - \sigma^2/2)(T-\tau) + \sigma\sqrt{T-\tau}W}.$$

Note that, given a fixed value of ω and a standard normal W , the random variable $S_\tau(\omega, W)$ is distributed according to the risk-neutral distribution of underlying asset price at the option maturity T , conditional on asset price $S_\tau(\omega)$ at the risk horizon τ . Given

an outer scenario ω_i and decision j , each inner loss sample takes the form

$$\hat{Z}_{i,j,l} = 3X_0 - 3e^{-r(T-\tau)} \max(K - S_\tau(\omega_i, W_{i,l}), 0) - ja_0,$$

where $W_{i,l}$ is an independent standard normal random variable. It is not difficult to see that the loss $L(\omega, j^*)$ ($j^* = 0$) by taking worst decision is strictly increasing in the risk factor ω . Hence, the probability of a loss exceeding a threshold α_0 can be computed according to $\alpha = P(L \geq \alpha_0) = P(\omega \geq \omega^*)$, where ω^* is the unique solution to $L(\omega^*) = \alpha_0$. We conduct Experiment 2 using a total computing budget of 250,000 simulation samples and assume there are 500 scenarios. To be consistent with the example in Broadie et al. (2011), we let $\omega \in [1.2816 - \frac{899}{50}, \dots, 1.2816 - \frac{3}{50}, 1.2816 - \frac{1}{50}, 1.2816 + \frac{1}{50}, 1.2816 + \frac{3}{50}, \dots, 1.2816 + \frac{99}{50}]$ and choose the values 0.859 for the loss threshold α_0 , corresponding to ω of 1.2816 and loss probabilities P^* of 10%. In addition, we also repeat this whole procedure 1000 times and then calculate the estimation bias obtained from these 1000 independent macro replications for each method. We also provide 10 initial runs for each decision among all scenarios ($m_0 = 10$) for all the four methods.

Fig. 3 plots the simulation results. OCBA-2S clearly outperforms EA, EA-OCBA, EA-NkOCBA, cOCBA and PSC-OCBA. Specifically, after exhausting the total simulation budget of 250,000 simulations, EA, EA-OCBA, EA-NkOCBA, cOCBA and PSC-OCBA reduced the estimation bias to 0.0025, 0.0016, 0.0013, 0.0086 and 0.0012 respectively. In contrast, OCBA-2S reduced the bias to 0.0004. On the other hand, OCBA-2S used about 14,000, 32,000, 44,000, 2000 and 50,000 simulation budget to achieve the same bias by EA, EA-OCBA, EA-NkOCBA, cOCBA and PSC-OCBA, respectively. That is to say, OCBA-2S achieved a speed-up factor of more than 18, 8, 5, 125 and 5 over EA, EA-OCBA, EA-NkOCBA, cOCBA and PSC-OCBA, respectively.

5.3. Experiment 3: Resource allocation in a healthcare system

The test problem in Experiment 3 studies the resource allocation problem in a critical care facility depicted in Fig. 4. This test case was first introduced in Schruben & Margolin (1978) and then widely by other researchers, for example Ng & Chick (2006) and Xie, Nelson, & Barton (2014). We use the expected number of patients per month that are denied a bed as the system's performance. It is a function of the number of beds in the intensive care unit (ICU), coronary care unit (CCU), and intermediate care units. Patients arrive according to a Poisson process and are routed

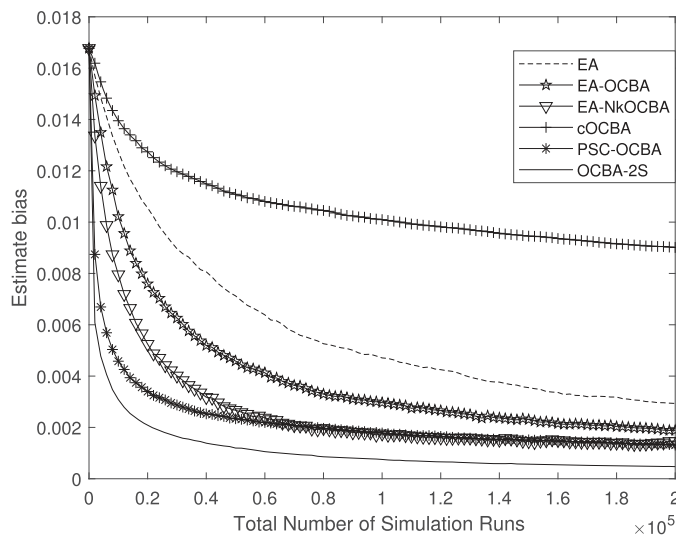


Fig. 3. Estimation bias as a function of the total number of simulation samples in experiment 2.

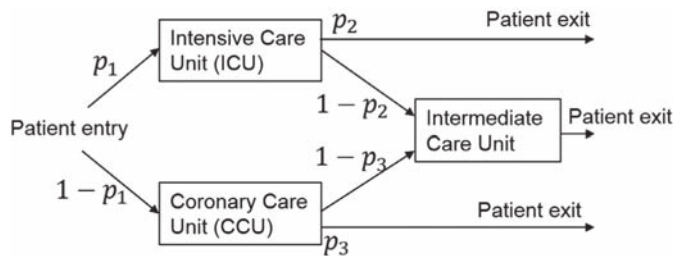


Fig. 4. Patient flows through different units of a critical care facility.

through the system depending upon their specific health condition. Specifically, we assume the patients are placed to ICU and CCU with probability p_1 and $1 - p_1$, respectively. In addition, after a period of treatment, a patient in ICU can exit the system or transferred to intermediate care units with probability p_2 and $1 - p_2$, respectively. A patient in CCU can exit the system or transferred to intermediate care units with probability p_3 and $1 - p_3$, respectively. The patient flows through the facility is depicted in Fig. 4.

We fix ten parameters of nine different sources of randomness: the patient arrival process (Poisson arrivals, mean day), ICU stay duration (lognormal mean and standard deviation), lognormal service times at the intermediate ICU (with mean and standard deviation), intermediate CCU (with mean and standard deviation), and CCU processes (with mean and standard deviation). The number of beds in ICU, CCU and intermediate care units are 4, 2 and 5, respectively.

We regard different probabilities $p = (p_1, p_2, p_3)$ as different scenarios in the system. We consider 8 possible values for p_1 , p_2 and p_3 : $p_1 \in \{0.55, 0.60, \dots, 0.90\}$, $p_2 \in \{0.10, 0.15, \dots, 0.45\}$, and $p_3 \in \{0.10, 0.15, \dots, 0.45\}$. So together are $8^3 = 512$ scenarios. In each scenario, we need to decide how to allocate two additional beds to ICU, CCU or intermediate care units, which gives a total of 6 feasible decisions in each scenario. Because of the complexity of the problem, we do not know the true system performance in each scenario and instead use a stochastic simulation model to estimate it. For the purpose of benchmarking, we generated 5000 simulation samples for each of the 6 decisions in each of the 512 scenarios and take the sample means as a reasonably accurate estimate for the true system performance. We set the threshold $\alpha_0 = 2.086$, which corresponds to $P^* = 0.1$. We set a total computing budget of

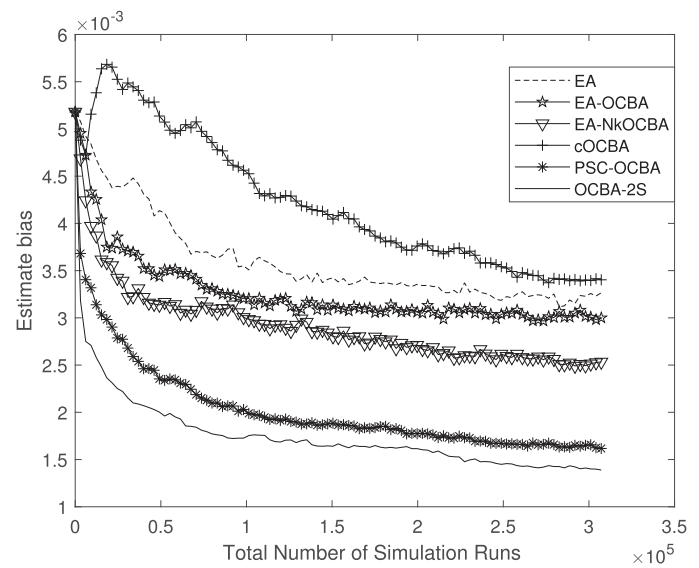


Fig. 5. Experiment 3: $\alpha_0 = 2.086$, 90% Quantile.

307,200 simulation samples. We again conducted $m_0 = 10$ initial runs for each decision for all four allocation policies.

Fig. 5 plots the simulation results. OCBA-2S again clearly outperformed EA, EA-OCBA, EA-NkOCBA, cOCBA and PSC-OCBA. Specifically, after exhausting the total simulation budget of 307,200 simulations, EA, EA-OCBA, EA-NkOCBA, cOCBA and PSC-OCBA reduced the estimation bias to 0.0032, 0.0030, 0.0025, 0.0034 and 0.0016, respectively. In contrast, OCBA-2S reduced the bias to 0.0014. On the other hand, OCBA-2S used about 3000, 6000, 15,000, 3000 and 206,000 simulation budget to achieve the same bias by EA, EA-OCBA, EA-NkOCBA, cOCBA and PSC-OCBA, respectively. That is to say, OCBA-2S achieved a speed-up factor of more than 102, 51, 20, 102 and 1.5 over EA, EA-OCBA, EA-NkOCBA, cOCBA and PSC-OCBA, respectively.

6. Conclusions

In this paper, we propose the OCBA-2S simulation budget allocation policy for the efficient estimation of a risk measure that requires solving a two-stage simulation optimization problem. OCBA-2S is built upon a new metric known as RPSC, and uses it to guide a sequential allocation of simulation budget to first-stage scenarios and then second stage decisions. The consistency of OCBA-2S is proved and its computational efficiency is demonstrated using three sets of test problems in comparison with three other simulation budget allocation policies. As to the best of our knowledge, this paper makes a first step towards the development of a rigorous and efficient simulation budget allocation procedure for estimation of a risk measure that requires a solving two-stage simulation optimization problems.

There are multiple venues for us to expand research on this topic. This paper is concerned with the estimation of a risk probability. We plan to extend the procedure to cover other widely used risk measures such as value-at-risk (VaR) and conditional value-at-risk (CVaR). In this paper, we consider problems where the scenario space is finite, and we focus on the allocation of second-stage simulation to a finite number of decisions under each scenario. An important direction to pursue in the future is when the scenario space is infinite, either countably infinite or continuous, how one should balance the sampling of the scenario space and the second stage decision space. Without sampling enough scenarios, the accuracy of the estimate would certainly be poor. However, sampling too many scenarios would also hurt the estimate's accuracy be-

cause there would not be enough simulation budget to estimate $Y^*(\omega)$. We believe that combining the basic principle of OCBA-2S method and some good criterion of balancing the sampling of the first-stage scenario space and the second stage decision space, there is a good chance of designing more generally applicable algorithms. Another potentially fruitful venue for research is the development of two-stage simulation budget allocation for the new problem known as two-stage SPvS (Xie et al., 2020). For example, such procedures can be used to efficiently estimate chance constraints in two-stage SPvS. Finally, it is also possible to extend this work for general two-stage simulation optimization problems where the simulation evaluation of a solution requires a two-stage simulation. We are not aware of any work in this direction. The closest paper is by Zhu, Xu, Chen, Lee, & Hu (2016), who studied for classical simulation optimization how to balance the number of solutions generated by a random search algorithm and the number of simulation replications for each sampled solution. The development of an efficient two-stage simulation optimization computing budget allocation procedure may help solve important problems such as optimal infrastructure investment to maximize the system's resilience against natural hazards Miller-Hooks et al. (2012).

Acknowledgments

Wang and Hu are supported in part by the National Natural Science Foundation of China (NSFC) under Grants 71720107003 and 72033003. Xu is supported in part by the National Science Foundation (NSF) under grant DMS-1923145 and UChicago Argonne LLC under grant 1F-60250. Chen is supported in part by the National Science Foundation under grant FAIN-2123683.

References

- Branke, J., Chick, S. E., & Schmidt, C. (2007). Selecting a selection procedure. *Management Science*, 53, 1916–1932.
- Broadie, M., Du, Y., & Moallemi, C. C. (2011). Efficient risk estimation via nested sequential simulation. *Management Science*, 57, 1172–1194.
- Broadie, M., Du, Y., & Moallemi, C. C. (2015). Risk estimation via regression. *Operations Research*, 63, 1077–1097.
- Bureau, N. S. (2020a). National Bureau of Statistics of China 2020. URL: <http://www.stats.gov.cn/tjsj/ndsj/2020/indexch.htm>.
- Bureau, S. S. (2020b). Shanghai statistical yearbook 2020. URL: <http://tjj.sh.gov.cn/tjnj/20210303/2abf188275224739bd5bce9bf128aca8.html>.
- Chen, C. H., & Lee, L. H. (2011). *Stochastic simulation optimization: An optimal computing budget allocation*: vol. 1. World scientific.
- Chen, C. H., Lin, J., Yücesan, E., & Chick, S. E. (2000). Simulation budget allocation for further enhancing the efficiency of ordinal optimization. *Discrete Event Dynamic Systems*, 10, 251–270.
- Chen, C. H., Yücesan, E., Dai, L., & Chen, H. C. (2009). Optimal budget allocation for discrete-event simulation experiments. *IIE Transactions*, 42, 60–70.
- Chen, W., Gao, S., Chen, C. H., & Shi, L. (2013). An optimal sample allocation strategy for partition-based random search. *IEEE Transactions on Automation Science and Engineering*, 11, 177–186.
- Dang, O., Feng, M., & Hardy, M. R. (2019). Efficient nested simulation for conditional tail expectation of variable annuities. *North American Actuarial Journal*, 24, 1–24.
- Fei, X., Gülpınar, N., & Branke, J. (2019). Efficient solution selection for two-stage stochastic programs. *European Journal of Operational Research*, 277, 918–929.
- Gao, S., & Chen, W. (2016). A partition-based random search for stochastic constrained optimization via simulation. *IEEE transactions on Automatic Control*, 62, 740–752.
- Gao, S., Chen, W., & Shi, L. (2017). A new budget allocation framework for the expected opportunity cost. *Operations Research*, 65, 787–803.
- Gao, S., Du, J., & Chen, C. H. (2019). Selecting the optimal system design under covariates. In *2019 IEEE 15th international conference on automation science and engineering (CASE)* (pp. 547–552). IEEE.
- Goodwin, T., Xu, J., Celik, N., & Chen, C. H. (2022). Real-time digital twin-based optimization with predictive simulation learning. *Journal of Simulation*, 1–18.
- Goodwin, T., Xu, J., Chen, C. H., & Celik, N. (2021). Efficient simulation optimization with simulation learning. In *2021 IEEE international conference on automation science and engineering (CASE)* (pp. 1–6). IEEE.
- Gordy, M. B., & Juneja, S. (2010). Nested simulation in portfolio risk measurement. *Management Science*, 56, 1833–1848.
- Groves, M., & Branke, J. (2019). Top- k selection with pairwise comparisons. *European Journal of Operational Research*, 274, 615–626.
- Hong, L. J., Juneja, S., & Liu, G. (2017). Kernel smoothing for nested estimation with application to portfolio risk measurement. *Operations Research*, 65, 657–673.
- Hu, J., Fu, M., & Marcus, S. (2008). A model reference adaptive search method for stochastic global optimization. *Communications in Information and Systems*, 8, 245–276.
- Kim, S., & Nelson, B. L. (2001). A fully sequential procedure for indifference-zone selection in simulation. *ACM Transactions on Modeling and Computer Simulation*, 11, 251–273.
- Lan, H., Nelson, B. L., & Staum, J. (2010). A confidence interval procedure for expected shortfall risk measurement via two-level simulation. *Operations Research*, 58, 1481–1490.
- LaPorte, G. J., Branke, J., & Chen, C. H. (2012). Optimal computing budget allocation for small computing budgets. In *Proceedings of the 2012 winter simulation conference (WSC)* (pp. 1–13). IEEE.
- LaPorte, G. J., Branke, J., & Chen, C. H. (2015). Adaptive parent population sizing in evolution strategies. *Evolutionary Computation*, 23, 397–420.
- Lee, L. H., Chew, E. P., & Manikam, P. (2006). A general framework on the simulation-based optimization under fixed computing budget. *European Journal of Operational Research*, 174, 1828–1841.
- Li, H., Lam, H., Liang, Z., & Peng, Y. (2020). Context-dependent ranking and selection under a Bayesian framework. In *2020 winter simulation conference (WSC)* (pp. 2060–2070). IEEE.
- Li, J., Liu, W., Pedrielli, G., Lee, L. H., & Chew, E. P. (2017). Optimal computing budget allocation to select the nondominated systems a large deviations perspective. *IEEE Transactions on Automatic Control*, 63, 2913–2927.
- Liu, M., & Staum, J. (2010). Stochastic kriging for efficient nested simulation of expected shortfall. *Journal of Risk*, 12, 3.
- Liu, Y., Pedrielli, G., Li, H., Lee, L. H., Chen, C. H., & Shortle, J. F. (2019). Optimal computing budget allocation for stochastic $n - k$ problem in the power grid system. *IEEE Transactions on Reliability*, 68, 778–789.
- Luo, J., Hong, L. J., Nelson, B. L., & Wu, Y. (2015). Fully sequential procedures for large-scale ranking-and-selection problems in parallel computing environments. *Operations Research*, 63, 1177–1194.
- Miller-Hooks, E., Zhang, X., & Faturechi, R. (2012). Measuring and maximizing resilience of freight transportation networks. *Computers and Operations Research*, 39, 1633–1643.
- Nelson, B. L., Swann, J., Goldsman, D., & Song, W. (2001). Simple procedures for selecting the best simulated system when the number of alternatives is large. *Operations Research*, 49, 950–963.
- Ng, S. H., & Chick, S. E. (2006). Reducing parameter uncertainty for stochastic systems. *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, 16, 26–51.
- Pedrielli, G., Sencuk Candan, K., Chen, X., Mathesen, L., Inanlouganji, A., Xu, J., et al., (2019). Generalized ordinal learning framework (golf) for decision making with future simulated data. *Asia-Pacific Journal of Operational Research*, 36, 1940011.
- Peng, Y., Chen, C. H., Fu, M. C., & Hu, J. Q. (2016). Dynamic sampling allocation and design selection. *INFORMS Journal on Computing*, 28, 195–208.
- Qu, H., Ryzhov, I. O., Fu, M. C., & Ding, Z. (2015). Sequential selection with unknown correlation structures. *Operations Research*, 63, 931–948.
- Schruben, L. W., & Margolin, B. H. (1978). Pseudorandom number assignment in statistically designed simulation and distribution sampling experiments. *Journal of the American Statistical Association*, 73, 504–520.
- Shen, H., Hong, L. J., & Zhang, X. (2021). Ranking and selection with covariates for personalized decision making. *INFORMS Journal on Computing*.
- Sun, Y., Apley, D. W., & Staum, J. (2011). Efficient nested simulation for estimating the variance of a conditional expectation. *Operations Research*, 59, 998–1007.
- Teng, S., Lee, L. H., & Chew, E. P. (2010). Integration of indifference-zone with multi-objective computing budget allocation. *European Journal of Operational Research*, 203, 419–429.
- Thanos, A. E., Bastani, M., Celik, N., & Chen, C. H. (2015). Dynamic data driven adaptive simulation framework for automated control in microgrids. *IEEE Transactions on Smart Grid*, 8, 209–218.
- Vidyashankar, A. N., & Xu, J. (2013). Adaptive nested rare event simulation algorithms. In *2013 winter simulations conference (WSC)* (pp. 736–744). IEEE.
- Wang, T., Xu, J., & Hu, J. Q. (2021a). A study on efficient computing budget allocation for a two-stage problem. *Asia-Pacific Journal of Operational Research*, 38, 2050044.
- Wang, T., Yu, P., & Hu, J. (2021b). Admission control game with capacity borrowing. *Production and Operations Management*, 31(2), 547–560.
- Xiao, H., Lee, L. H., & Chen, C. H. (2015). Optimal budget allocation rule for simulation optimization using quadratic regression in partitioned domains. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 45, 1047–1062.
- Xiao, H., Lee, L. H., Morrice, D., Chen, C. H., & Hu, X. (2021). Ranking and selection for terminating simulation under sequential sampling. *IIE Transactions*, 53, 735–750.
- Xie, W., Nelson, B. L., & Barton, R. R. (2014). A Bayesian framework for quantifying uncertainty in stochastic simulation. *Operations Research*, 62, 1439–1452.
- Xie, W., Yi, Y., & Zheng, H. (2020). Global-local metamodel-assisted stochastic programming via simulation. *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, 31, 1–34.
- Xu, J., Huang, E., Chen, C. H., & Lee, L. H. (2015). Simulation optimization: A review and exploration in the new era of cloud computing and big data. *Asia-Pacific Journal of Operational Research*, 32, 1550019.
- Xu, J., Huang, E., Hsieh, L., Lee, L. H., Jia, Q. S., & Chen, C. H. (2016). Simulation optimization in the era of industrial 4.0 and the industrial internet. *Journal of Simulation*, 10, 310–320.

- Zhang, S., Xu, J., Lee, L. H., Chew, E. P., Wong, W. P., & Chen, C. H. (2016). Optimal computing budget allocation for particle swarm optimization in stochastic optimization. *IEEE Transactions on Evolutionary Computation*, 21, 206–219.
- Zhao, Y., Yu, P., & Hu, J. (2022). Strategic admission behavior and its implications: Evidence from a cardiac surgery department. *Journal of the Operations Research Society of China*, 1–21.
- Zhong, Y., Liu, S., Luo, J., & Hong, L. J. (2021). Speeding up paulsons procedure for large-scale problems using parallel computing. *INFORMS Journal on Computing*, 34(1), 586–606.
- Zhou, C., Xu, J., Miller-Hooks, E., Zhou, W., Chen, C. H., Lee, L. H., et al., (2021). Analytics with digital-twinning: A decision support system for maintaining a resilient port. *Decision Support Systems*, 143, 113496.
- Zhu, C., Xu, J., Chen, C. H., Lee, L. H., & Hu, J. Q. (2016). Balancing search and estimation in random search based stochastic simulation optimization. *IEEE Transactions on Automatic control*, 61, 3593–3598.
- Zhu, H., Hale, J., & Zhou, E. (2018). Simulation optimization of risk measures with adaptive risk levels. *Journal of Global Optimization*, 70, 783–809.