Proceedings of the 2023 Winter Simulation Conference
C. G. Corlu, S. R. Hunter, H. Lam, B. S. Onggo, J. Shortle, and B. Biller, eds.

CLUSTER-BASED SAMPLING ALLOCATION FOR MULTI-FIDELITY SIMULATION OPTIMIZATION

Zirui Cao
Haobin Li
Ek Peng Chew

Department of Industrial Systems Engineering and Management
National University of Singapore
1 Engineering Drive 2
Singapore, 117576, SINGAPORE

Haowei Wang

Rice-Rick Digitalization PTE. Ltd.
51 Goldhill Plaza
Singapore, 308900, SINGAPORE

Kok Choon Tan

Department of Analytics and Operations
National University of Singapore
15 Kent Ridge Drive
Singapore, 119245, SINGAPORE

ABSTRACT

Simulation optimization is widely used to optimize complex systems. High-fidelity simulation can be expensive, especially when the number of designs is large. In practice, fast but less accurate low-fidelity simulation is often available and can provide valuable information. In this paper, we propose a sampling algorithm that utilizes information from multiple fidelity simulation models to improve the efficiency of searching for the best design. A *k*-means algorithm is introduced to help capture the performance clustering phenomenon among designs, and a cluster validity index is proposed to determine the optimal number of clusters. The proposed sampling algorithm can incorporate the information of performance clusters and approximately minimize the expected opportunity cost of the selected best design. Numerical results substantiate the superior performance of the proposed algorithm.

1 INTRODUCTION

Discrete-event simulation (DES) can fully capture the simulated systems' challenging features and provides accurate estimates to designs' performance. By applying DES, managers can select the design with the most promising estimated performance for implementation. In practice, simulation models with different fidelities are often available. A high-fidelity model, e.g., a DES model, is accurate but suffers from high computational cost. On the other hand, low-fidelity simulation models, e.g., analytical models and reduced-order simulation models, are less accurate but cheaper and faster to run. However, due to some significant features being trimmed off by the low-fidelity model, performance estimates provided by it may be biased, and the bias is generally unknown and even large in scale. Therefore, making use of both fidelity simulations is promising as it takes the advantages of the both and can fully utilize the simulation outputs.

The efficiency of searching for the best design via high-fidelity simulation can be substantially improved if the significant information of low-fidelity simulation can be well utilized. The premise of this paper is

979-8-3503-6966-3/23/$31.00 ©2023 IEEE

that low-fidelity results can help classify designs into performance clusters. Specifically, designs within a performance cluster tend to have similar performances, while the behavior of designs in different performance clusters can be drastically different. The clustering phenomenon captured by low-fidelity results provides useful global information that can further enhance the efficiency of searching locally within clusters.

Utilizing the information of multi-fidelity models to improve the efficiency of optimization is a hot topic in simulation optimization literature. Multi-fidelity optimization usually involves constructing surrogate models (Fernández-Godino et al. 2019). Two most widely-used surrogate models are the kriging model (Gaussian process) (Huang et al. 2006; Wang et al. 2022) and the radial basis function (Gutmann 2001; March and Willcox 2012). The intuition of the surrogate method is to fit a surrogate model to approximate the objective and utilizes it to identify the most promising design to sample for the next iteration. The construction of surrogate models usually replies on a desirable domain structure. For example, the kriging model is built based on the assumption that the difference in performance between two designs is small if the two designs are close to each other (Toal 2015). More recently, Xu and Zheng (2023) develop a gradient-based method to solve simulation optimization problems when multi-resolution approximated systems are available. However, the original design space in this paper can be high-dimensional, discrete, and categorical. Therefore, both surrogate methods and gradient-based methods are not a suitable treatment.

Another branch of literature related to this paper follows the path of Ranking and Selection (R&S) literature. The goal of R&S is to identify the best system design from a finite set of alternatives, where "best" is defined with respect to the smallest/largest mean performance. The performance of each design can be learned from samples, e.g., from simulation outputs. The idea of utilizing the information of performance clusters provided by low-fidelity simulations firstly appears in Xu et al. (2016), in which the MO^{2}TOS algorithm is introduced. The MO^{2}TOS algorithm considers deterministic simulation and regards each performance cluster as a "design" in R&S. It uses the OCBA allocation rule in Chen et al. (2000) to balance simulation budget among clusters. Li et al. (2015) and Zhang et al. (2016) go further along this line, and they extend the MO^{2}TOS algorithm to settings where multiple objectives and multiple levels of models are available, respectively. More recently, Song et al. (2019) give a theoretical support to the MO^{2}TOS algorithm. Furthermore, Peng et al. (2018) use Gaussian Mixture Model (GMM) to capture the performance clustering phenomenon among designs and consider the case where simulation realizations are observed with stochastic noise. However, MO^{2}TOS allows only equal-sized clusters; is a pure exploration procedure; and has difficulty in determining the optimal number of clusters, which plays a key role in the clustering problem. As for the GMM method, its applicability is limited by high computational cost.

In this paper, we develop a more general searching algorithm, called cluster-based multi-fidelity optimal sampling (CMFOS). It aims to minimize the expected optimality cost (EOC) of the selected design, that is, to minimize the expected gap between the performances of the selected design and the real best design. In some context, EOC is also referred to as expected optimality gap (EOG) (Song et al. 2019). Theoretical results show that minimizing the expected opportunity cost (EOC) of the selected best design is equivalent to consuming all sampling budget on searching within the best cluster. CMFOS firstly uses a k-means algorithm to partition designs into performance clusters according to low-fidelity outputs. Then, based on the clustering results, CMFOS applies high-fidelity simulation to explore the best cluster (exploration) and further search for the best design (exploitation). In particular, to determine the optimal number of clusters, a modified Davies-Bouldin Index is introduced. Compared with the MO^{2}TOS algorithm in Xu et al. (2016), CMFOS is more flexible in partitioning designs; achieves a better balance between exploration and exploitation; and can determine the optimal number of clusters. Furthermore, in contrast to the GMM method in Peng et al. (2018), CMFOS is more efficient in terms of computational burden.

The rest of the paper is organized as follows. The problem statement and formulation are presented in Section 2. In Section 3, we apply a k-means algorithm to partition designs into performance clusters and develop an index for determining the optimal number of clusters. In Section 4, the optimal sampling allocation procedure is formally proposed. Numerical results on a synthetic example, two benchmark test functions, as well as a case study are given in Section 5. In the end, Section 6 concludes the paper.

2 PROBLEM STATEMENT

We consider a simple case when only one high- and low-fidelity model are available, and the results can be easily extended to the scenario where one high-fidelity model and multiple low-fidelity models are available (Xu et al. 2016). Suppose that we have m independent designs $x_1, x_2, \ldots, x_m$, and the accurate performance of design x_i, which is denoted by $h(x_i)$, for $i = 1, 2, \ldots, m$, can only be estimated by running a high-fidelity simulation model. The goal is to identify the design with the smallest performance

$$x^* = \arg\min_{i=1,2,\ldots,k} h(x_i),$$

where x^* denotes the best design. In addition to the high-fidelity simulation model, we also have access to a low-fidelity simulation model that provides less accurate approximations for $h(x_i)$. Let $l(x_i)$ be the low-fidelity estimate of designs x_i, for $i = 1, 2, \ldots, m$. Furthermore, we assume the high-fidelity sampling budget N^{max} is much smaller than the number of designs m. As for the low-fidelity model, we assume its sampling budget is large enough for evaluating every design's performance, and therefore, the low-fidelity results $l(x_i)$, for $i = 1, 2, \ldots, m$, are completely observable and known. These assumptions are consistent with the fact that high-fidelity simulation is often time-consuming to run, and high-fidelity sampling budget is limited by its high expense, while low-fidelity simulation is much faster to run and can evaluate a huge number of designs in a very short time.

2.1 Performance Clustering

Suppose that the set of designs and their low-fidelity performances, denoted by $A = \{x_1, x_2, \ldots, x_m\}$ and $L = \{l(x_1), l(x_2), \ldots, l(x_m)\}$, respectively, are given. Without loss of generality, we consider partitioning set A into a given number k of disjoint clusters $\{\pi_1, \pi_2, \ldots, \pi_k\}$, such that

1. $|\pi_j| \geq 1$
2. $\pi_r \cap \pi_s = \emptyset, r \neq s$
3. $\bigcup_{j=1}^{k} \pi_j = A$

Since designs are discrete and categorical, it is important to note that their performances can be drastically different, even if they are located closely together in the original design space. This observation motivates the need of partitioning designs according to their low-fidelity performances. The similarities among designs are measured by Euclidean distances. For each cluster $\pi_j, j = 1, 2, \ldots, k$, its centroid c_j in the low-fidelity performance space is determined by

$$c_j = \arg\min_a \sum_{x_i \in \pi_j} \frac{1}{n_j} \|a - l(x_j)\|^2, \tag{1}$$

where $n_j := |\pi_j|$ is the number of designs in cluster π_j, and $\|\cdot\|$ is the Euclidean norm. In contrast, for a given set of centroids $\mathbf{c} = (c_1, c_2, \ldots, c_k)$, cluster π_j can be defined by

$$\pi_j = \{x_i \in A : \|c_j - l(x_i)\|^2 \leq \|c_s - l(x_i)\|^2, \ \forall s = 1, 2, ..., k\}. \tag{2}$$

The most widely used formulation of the clustering problem is a non-smooth, non-convex optimization problem (Späth 1984; Teboulle 2007)

$$\begin{aligned} \min \ f_k(\mathbf{c}) &= \sum_{i=1}^{m} \min_{j=1,2,\ldots,k} \|c_j - l(x_i)\|^2 \\ \text{s.t. } \mathbf{c} &= (c_1, c_2, \ldots, c_k) \in \mathbb{R}^k, \end{aligned} \tag{3}$$

where $f_k(\mathbf{c})$ is called the total clustering error. Problem (3) is also known as the minimum sum-of-squares clustering problem.

2.2 Sampling Allocation Problem

After designs are partitioned into performance clusters by their low-fidelity approximations, we proceed to search for the best design x^* by randomly sampling high-fidelity model. Assume that in each cluster π_j, designs' high-fidelity performances $h(X)$ are i.i.d. normally distributed, i.e., $h(X) \sim N(h_j, \sigma_j^2), \forall X \in \pi_j, j = 1,2,\ldots,k$, where h_j and σ_j^2 are called group mean and group variance of cluster π_j (Xu et al. 2016).

Given a fixed number of high-fidelity sampling evaluations N^{max}, which are much smaller than the number of design m, the objective is to determine N_j, the number of high-fidelity evaluations allocated to cluster π_j, for $j = 1,2,\ldots,k$, to minimize the EOC of the selected best designs after the high-fidelity sampling budget is exhausted. Therefore, we consider the following sampling allocation problem:

$$\begin{aligned} \min_{N_1,N_2,\ldots,N_k} \quad & \mathbb{E}[\Delta H] = \mathbb{E}\left[h(X_b) - h(X^*)\right] \\ \text{s.t.} \quad & \sum_{j=1}^{k} N_j = N^{max}, \\ & N_j \geq 0, \quad j = 1,2,\ldots,k, \end{aligned} \tag{4}$$

where $h(X_b)$ and $h(X^*)$ denote the accurate performance of the observed best design X_b and the real best design X^*, respectively.

3 K-MEANS CLUSTERING

In this section, a global k-means algorithm is introduced to help partition designs into performance clusters according to their low-fidelity approximations. Furthermore, to determine the optimal number of clusters, a modified Davies-Bouldin Index is developed.

3.1 Global K-means Algorithm

The global k-means algorithm (GKM) is an incremental type of clustering algorithm that optimally adds one centroid at each iteration. It can find an at least near optimal partition of a given data set (Likas et al. 2003).

Algorithm 1 Global k-means algorithm

```
Input: $A$, $L$, $k$.
Initialization: Set $q = 1$, $\pi_1 = A$, and calculate the initial centroid $c_1$ by Equation (1).
while $q < k$ do
    for $x_i \in A$ do
        Apply $k$-means algorithm with starting points $(c_1,\ldots,c_{q-1},l(x_i))$, and obtain the updated centroids $\mathbf{y}^{x_i} = (y_1^{x_i}, y_2^{x_i}, \ldots, y_q^{x_i})$.
        Calculate the total clustering error $f_q(\mathbf{y}^{x_i})$ by Equation (3).
    end for
    $x' \leftarrow \arg\min_{x_i \in A} f_q(\mathbf{y}^{x_i})$, and $c_t \leftarrow y_t^{x'}, \forall t \in \{1,2,\ldots,q\}$.
    $q \leftarrow q+1$.
end while
Determine a $k$-partition of $A$ with centroids $\mathbf{c} = (c_1, c_2, \ldots, c_k)$ by Equation (2), and obtain $\{\pi_1, \pi_2, \ldots, \pi_k\}$.
Output: $\{\pi_1, \pi_2, \ldots, \pi_k\}$ and $\mathbf{c}$.
```

3.2 Determining the Number of Clusters

In simple cases, the number of clusters is automatically determined by the features of the problem itself. However, in practice, the number of clusters k is unlikely to be given in advance. To determine the optimal number of clusters, we introduce a modified Davies-Bouldin Index (MDBI)

$$MDBI = \left\{ \frac{1}{k} \sum_{i=1}^{k} \max_{j=1,\ldots,k, j \neq i} \frac{S_k(\pi_i) + S_k(\pi_j)}{d\left(c^i, c^j\right)} \right\} \times \frac{n_b}{N^{max}}, \tag{5}$$

where $S_k(\pi_i)$ is the average distance of designs in cluster π_i to the centroid c_i in the low-fidelity performance space, $d\left(c^i, c^j\right)$ represents the distance between two centroids c_i and c_j in the low-fidelity performance space, n_b denotes the number of designs in the best cluster, and N^{max} is the sampling budget.

The first term in MDBI is the Davies-Bouldin Index (DBI) developed by Davies and Bouldin (1979). The clusters are compact and well-separated if DBI is small. Furthermore, given a sampling budget N^{max}, we expect the number of designs in the best cluster is as small as possible. Therefore, the number of clusters k that minimizes MDBI is taken as the optimal number of clusters.

4 OPTIMAL SAMPLING PROCEDURE

In this section, we first show that the optimal sampling scheme, which can minimize the EOC, is allocating all sampling budget to the best cluster (with the smallest group mean performance). Then, a cluster-based multi-fidelity optimal sampling algorithm is developed.

Lemma 1 is firstly presented in Song et al. (2019), and it gives an explicit expression of EOC as the number of designs n_j in groups π_j, for $j = 1, 2, \ldots, k$, is sufficiently large.

Lemma 1 (Song et al. (2019)) Assume $h(X)$ for all designs $X \in \pi_j$ are independent and identically distributed (i.i.d.) with a cumulative distribution function $F_j(h)$, for $j = 1, 2, \ldots, k$. As the number of designs $n_j \to \infty$, for a given sampling allocation $N_1, N_2, \ldots, N_k$, we have

$$\lim_{\min\limits_{j=1,\ldots,k} n_j \to \infty} \mathbb{E}[\Delta H] \to -\int_{\underline{H}}^{\bar{H}} \prod_{j=1}^{k} \left[F_j(h)\right]^{N_j} dh, \tag{6}$$

where $\bar{H} = \max_{i=1,\ldots,m} h(X_i)$, and $\underline{H} = \min_{i=1,\ldots,m} h(X_i)$.

When the number of designs in each cluster is sufficiently large, minimizing the EOC is equivalent to maximizing the integral in Equation (6). To maximize the integral, we need to maximize the product inside. Therefore, as $n_j \to \infty$, for $j = 1, 2, \ldots, k$, Problem (4) can be rewritten as follows:

$$\begin{aligned} \max_{N_1, N_2, \ldots, N_k} & \prod_{j=1}^{k} \left[\Phi\left(\frac{h - \mu_j}{\sigma_j}\right)\right]^{N_j} \\ \text{s.t.} \quad & \sum_{j=1}^{k} N_j = N^{max}, \\ & N_j \geq 0, \quad j = 1, 2, \ldots, k, \end{aligned} \tag{7}$$

where $\Phi(\cdot)$ is the cumulative distribution function of the standard normal distribution.

Notice that $\Phi(\cdot) \in [0,1]$ is a monotone increasing function of $(h - \mu_j)/\sigma_j$. Suppose that the variances of all designs are upper bounded by a positive constant $\bar{\sigma}^2 > 0$, that is, $0 < \sigma_i^2 \leq \bar{\sigma}^2$, for $i = 1, 2, \ldots, k$. Basically, this assumption requires the best cluster can be distinguished from others. Then, we have $(h - \mu_j)/\bar{\sigma} \leq (h - \mu_j)/\sigma_j$. Moreover, it can be checked that for any $0 < a < b < 1$ and two non-negative integers i and j, we have $a^i b^j > a^j b^i$ if and only if $j > i$. Thus, maximizing the objective of Problem (7) is equivalent to first identifying the cluster with the largest $(h - \mu_j)/\sigma_j$, then allocating all high-fidelity sampling budget N^{max} to it. Since it is difficult to identify the largest $(h - \mu_j)/\sigma_j$, we alternatively turn to identify the largest lower bound for $(h - \mu_j)/\sigma_j$, that is, to identify the cluster with the largest $(h - \mu_j)/\bar{\sigma}$ or equivalently with the smallest μ_j.

Unfortunately, it is unknown which cluster has the smallest mean performance before sampling. To identify the best cluster (with the smallest group mean), a sampling budget $T, T < N^{max}$, is utilized to evaluate a few designs' high-fidelity performances in each cluster. Based on simulation output samples, the cluster with the smallest sample group mean is selected as the best. Then, the rest of $N^{max} - T$ sampling budget is allocated to the selected best cluster for further searching for the best design. In particular, the OCBA policy in Chen et al. (2000) is one of the most efficient budget allocation rules and can be implemented to maximize the probability of correctly selecting the best cluster. The optimal sampling allocation rule is given in Theorem 1.

Theorem 1 Given N^{max} sampling budget, in which $T, T < N^{max}$, sampling budget is used for identifying the best cluster, as $n_j \to \infty$, for $j = 1, 2, \dots, k$, the sampling allocation $N_1, N_2, \dots, N_k$, which solves Problem (4) and approximately minimizes the EOC, satisfies,

$$N_j = \begin{cases} w_j^* T & \text{if } j = 1, 2, \dots, k \text{ and } j \neq b \\ w_b^* T + (N^{max} - T) & \text{if } j = b \end{cases} \tag{8}$$

where,

$$w_j^* = \frac{I_j^{-1}}{\sum_{j=1}^k I_j^{-1}}, \qquad \text{for } j = 1, 2, \dots, k, \tag{9}$$

$$I_j = \begin{cases} \frac{(\mu_b - \mu_j)^2}{\sigma_j^2} & \text{if } j = 1, 2, \dots, k \text{ and } j \neq b \\ \frac{1}{\sigma_b \sqrt{\sum_{j=1, j \neq b}^k I_j^{-2} / \sigma_j^2}} & \text{if } j = b \end{cases}$$

The optimal sampling procedure includes two-stage. First, a given T sampling budget is used to distinguish clusters and identify the estimated best cluster (exploration). Second, all the rest of $N^{max} - T$ sampling budget is allocated to the best cluster for further searching for the best design (exploitation). In particular, if $T = N^{max}$, the sampling procedure utilizes all budget for exploration, and it becomes the same allocation rule as the OS policy introduced in Xu et al. (2016) and Song et al. (2019). Based on preceding analyses, the cluster-based multi-fidelity optimal sampling (CMFOS) procedure is described with details in Algorithm 2.

Algorithm 2 CMFOS

```
Input: $A$, $l(\cdot)$, $h(\cdot)$, $k$, $N^{max}$, $T$, $n_0$, and $D = \emptyset$
for $x_i \in A$ do
    Apply low-fidelity model to evaluate design $x_i$, and obtain $l(x_i)$
end for
(Partitioning): Run GKM algorithm to obtain a $k$-partition of $A$, i.e., $P = \{\pi_1, \pi_2, \dots, \pi_k\}$
for $\pi_j \in P$ do
    Sample $n_0$ designs from $\pi_j$, and obtain $Y_j = \{X_{j,1}, X_{j,2}, \dots, X_{j,n_0}\}$
    Apply high-fidelity model to obtain $\{h(X_{j,1}), h(X_{j,2}), \dots, h(X_{j,n_0})\}$
    Calculate $\hat{h}_j$, $\hat{\sigma}_j$
    $\pi_j \leftarrow \pi_j \backslash Y_j$, and $D \leftarrow D \cup Y_j$
end for
Set $t = 1$
while $t < T$ do
    Calculate $\mathbf{w} = (w_1^*, w_2^*, \dots, w_k^*)$ by Equation (9)
    Generate a random variable $s$ with $\Pr\{s = j\} = w_j, j = 1, 2, \dots, k$
    Sample a design $X_{s,1}$ from $\pi_j$
    Apply high-fidelity model to obtain $h(X_{s,1})$
    Update $\hat{h}_s$, $\hat{\sigma}_s$
```

```
        $\pi_s \leftarrow \pi_s \backslash X_{s,1}$, $D \leftarrow D \cup \{X_{s,1}\}$, and set $t = t+1$
    **end while**
    Set $b = \arg \min_{j=1,\ldots,k} \hat{h}_j$
    Sample $(N^{max} - n_0 \times k - T)$ designs from $\pi_b$, and obtain $Y_b = \{X_{b,1}, \ldots, X_{b,(N^{max}-n_0 \times k - T)}\}$
    Apply high-fidelity simulation model to obtain $\{h(X_{b,1}), \ldots, h(X_{b,(N^{max}-n_0 \times k - T)})\}$
    $D \leftarrow D \cup Y_b$
**Output:** $X_b = \arg\min_{X \in D} h(X)$, and $h(X_b)$
```

5 NUMERICAL EXPERIMENTS

In this section, we conduct numerical experiments on a synthetic example, two benchmark test functions, and also a case study to show the superior efficiency and applicability of our proposed CMFOS. The following searching algorithms are chosen as benchmarks for comparison:

1. Random sampling (RS): The RS algorithm is a pure random search procedure. It utilizes no information from low-fidelity outputs and randomly samples designs for evaluation.
2. MO^{2}TOS (Xu et al. 2016): The MO^{2}TOS algorithm firstly ranks designs according to their low-fidelity approximations. Then, it partitions designs into equal-sized clusters by ordinal rankings. The OCBA algorithm in Chen et al. (2000) is implemented to balance sampling evaluations among clusters. In particular, the MO^{2}TOS algorithm is a pure exploration procedure.

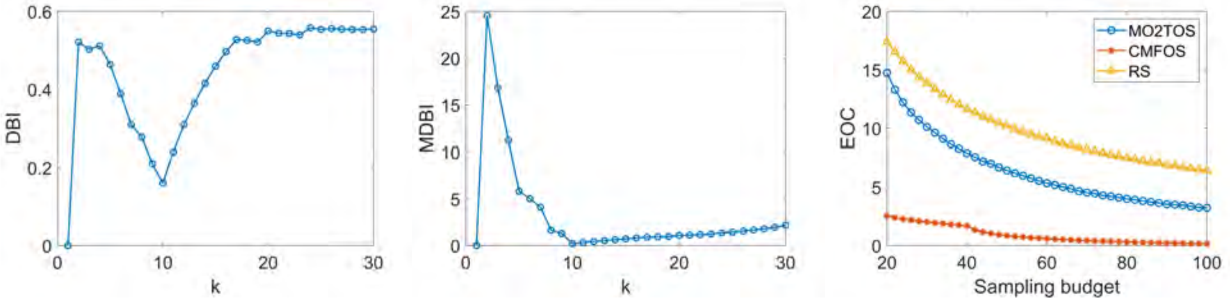
The performance of different procedures is measured by the EOC, i.e., the difference in high-fidelity performance between the selected best design and the truly best design. The empirical EOC is estimated via running 10,000 independent macro replications. In all tested cases, the number of initial sample size for each cluster is set as 2, i.e., $n_0 = 2$, the sampling budget $N^{max} = 100$, and the sampling budget for identifying the best cluster $T = 20$. In particular, the optimal number of clusters is provided by the k with the smallest MDBI.

5.1 A Synthetic Example

In this experiment, we generate 10 groups of designs, and the number of designs in group $1,2,3,\ldots,10$ is $n_1 = 100, n_2 = 300, n_3 = 500, \ldots, n_{10} = 1900$, respectively. Thus, the total number of designs $m = \sum_{i=1}^{10} n_i = 10000$. In each group j, design's low-fidelity performances are normally distributed with $N(h_j, \sigma_j^2)$, i.e., $l(x_{j,t}) \sim N(h_j, \sigma_j^2)$, for $j = 1,2,\ldots,10$ and $t = 1,2,\ldots,n_j$, where $h_1 = 10, h_2 = 20, \ldots, h_{10} = 100$, and $\sigma_1 = \sigma_2 = \cdots = \sigma_{10} = 1$. The approximation errors $\delta(x_{j,t})$ are normally distributed with $N(1,1)$, i.e., $\delta(x_{j,t}) \sim N(1,1)$, for $j = 1,2,\ldots,10$ and $t = 1,2,\ldots,n_j$. We use the equation $h(x_{j,t}) = l(x_{j,t}) + \delta(x_{j,t})$ in Xu et al. (2016) and Song et al. (2019) to generate the high-fidelity performances of design $x_{j,t}$. The correlation ρ between $h(\cdot)$ and $l(\cdot)$ is 0.99.

In Figure 1, two cluster validity indices and performance comparisons of the three tested procedures are reported. Both DBI and MDBI indicate the optimal number of clusters should be 10. As shown in Figure 1c, the behavior of CMFOS changes when the sampling budget grows up to 40. In the first stage, i.e., when $N^{max} \leq 40$, CMFOS consumes sampling budget for identifying the best cluster and is a pure exploration procedure. In the second stage, i.e., when $N^{max} > 40$, CMFOS allocates the rest of sampling budget to the best cluster and becomes a pure exploitation procedure.

CMFOS outperforms both MO^{2}TOS and RS. Compared with MO^{2}TOS, CMFOS performs better in the first stage, i.e., when $N^{max} \leq 40$, because it is more flexible and possesses a higher quality of partition due to allowing to cluster designs into unequal-size groups. Furthermore, CMFOS also achieves a much lower EOC than MO^{2}TOS in the second stage, i.e., when $N^{max} \geq 40$, because the sampling budget is more efficiently used to search for the best design within the identified best cluster by CMFOS. Or in other



(a) Davies-Bouldin index. (b) Modified Davies-Bouldin index. (c) Performance comparison.

Figure 1: Results for the synthetic example.

words, CMFOS better balances exploration and exploitation than MO^{2}TOS, which is a pure exploration procedure. In contrast to RS, which is a pure random search procedure, CMFOS benefits from utilizing the information of low-fidelity results to improve the efficiency of searching for the best design. Therefore, in this tested example, CMFOS has the best performance and dominates both MO^{2}TOS and RS.

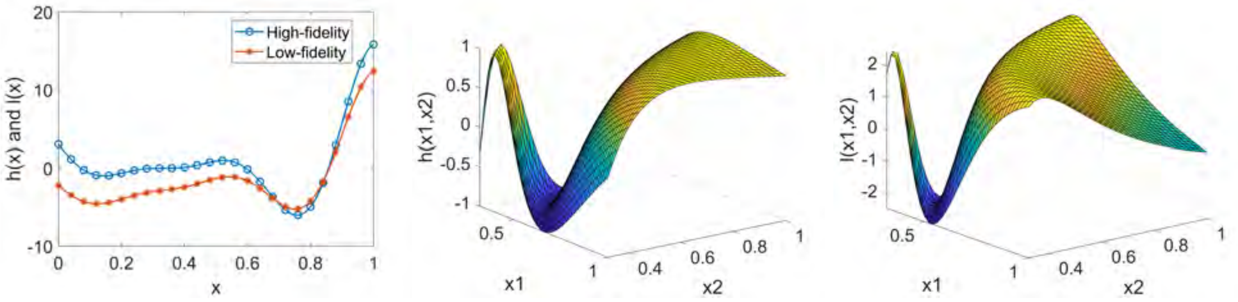
5.2 Two Benchmark Test Functions

In this section, we conduct experiments on two benchmark test functions, both of which can be found in Mainini et al. (2022), and they are discretized to be suitable for discrete optimization problems. In particular, the Gaussian sampling distribution assumption does not hold for both test function examples, and it is often the case in practice.

The first test function is the Forrester function introduced in Forrester et al. (2007), which is a well-known test function for optimization using multi-fidelity models. The one-dimensional Forrester function is defined by the following equations (from high-fidelity to low-fidelity) and is illustrated in Figure 2a,

$$h(x) = 6(x-2)^2 \sin(12x-4),\; x \in [0,1],$$
$$l(x) = 0.75h(x) + 5(x-0.5) - 2,\; x \in [0,1].$$

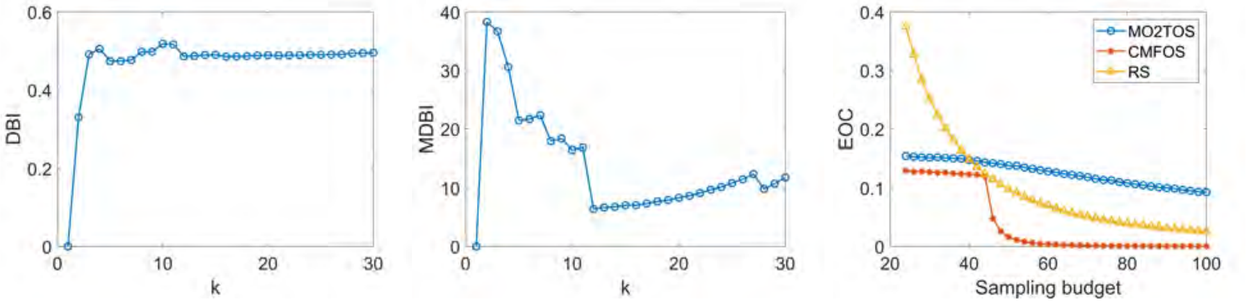
The global minimum of the Forrester function is given by $h(x^*) = -6.020740$, where $x^* = 0.75724876$. In this example, we randomly generate 10,000 designs from the interval $[0,1]$. The correlation ρ between $h(\cdot)$ and $l(\cdot)$ for this tested example can be calculated as 0.93.



(a) Forrester function. (b) Paciorek function: high-fidelity. (c) Paciorek function: low-fidelity.

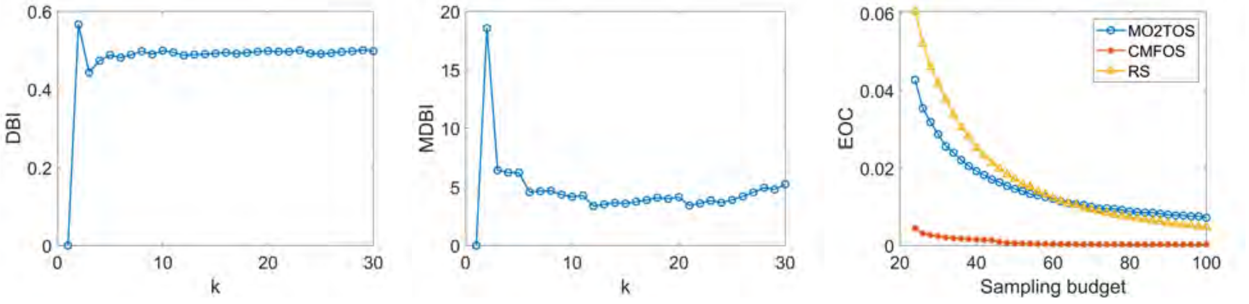
Figure 2: Benchmark test functions.

Figure 3 gives the two cluster validity indices and performance comparisons of the 3 tested procedures on the Forrester test function example. MDBI suggests $k = 12$ should be the optimal number of clusters, while DBI gives $k = 2$. When $k = 2$, the clusters are more compact and separable, however, the size of the best cluster is much larger than the case when $k = 12$. This leads to an inefficient searching. Therefore, we take $k = 12$ as the optimal number of clusters according to the MDBI introduced in Section 3.2.



(a) Davies-Bouldin index. (b) Modified Davies-Bouldin index. (c) Performance comparison.

Figure 3: Results for the Forrester test function example.



(a) Davies-Bouldin index. (b) Modified Davies-Bouldin index. (c) Performance comparison.

Figure 4: Results for the Paciorek test function example.

As shown in Figure 3c, the proposed CMFOS performs the best. In the first stage of CMFOS, it converges slowly because the sampling budget is used to identify the best cluster. After the best cluster is specified, the EOC of CMFOS approaches 0 much faster. Furthermore, MO^{2}TOS performs better than RS when the sampling budget is relatively small, but it is surpassed by RS when the sampling budget grows up to 40. The reason why MO^{2}TOS doesn't perform well in this example could be because it allows only equal-sized-groups clustering, which often leads to a large clustering error if designs should have been partitioned into unequal-sized groups. This results in MO^{2}TOS assigning too many sampling evaluations to clusters that are highly unlikely to contain the best design. However, CMFOS still dominates both MO^{2}TOS and RS in this example, verifying the benefits of unequal-sized-groups clustering, better balancing exploration and exploitation, and utilizing information of low-fidelity results.

The second benchmark test function is the Paciorek function, which is a multi-modal function (Toal 2015). The two-dimensional Paciorek function without stochastic noise is defined by the following equations (from high-fidelity to low-fidelity) and is plotted in Figure 2b and 2c,

$$h(x_1,x_2) = \sin\left(\frac{1}{x_1x_2}\right), \; x_1,x_2 \in [0.3,1],$$

$$l(x_1,x_2) = h(x_1,x_2) - 9A^2\cos\left(\frac{1}{x_1x_2}\right), \; x_1,x_2 \in [0.3,1],$$

where A is a parameter that varies between 0 and 1 and limits the approximation error of low-fidelity model. Specifically, the approximation error of the low-fidelity model increases as A increases. When $A = 0$, the low-fidelity model has no approximation error, and identifying the best design is equivalent to identifying the design with the best low-fidelity performance. In this experiment, we set $A = 0.5$, and randomly generate $m = 10000$ pairs of (x_1, x_2) from the interval $[0.3, 1]$ as designs. The correlation ρ between $h(\cdot)$ and $l(\cdot)$ for this tested example can be calculated as 0.38.

The two cluster validity indices and performance comparisons of the tested procedures on the Paciorek test function example are presented in Figure 4. According to Figure 4a and 4b, we take $k = 12$ as the optimal number of clusters. As shown in Figure 4c, CMFOS outperforms the other two algorithms from the beginning to the end. Again, MO^{2}TOS outperforms RS when the sampling budget is small, while it is surpassed by RS when the sampling budget is relatively large.

5.3 A Machine Allocation Problem

The machine allocation problem has been studied by Xu et al. (2016), Peng et al. (2018), and Song et al. (2019). The jobs flow of a manufacturing system is shown in Figure 5. There are two products P1 and P2, and P1 has priority over P2 in the system. When both products are waiting for being processed by the same workstation, the processing of P2 will be deferred. The inter-arrival and service time of both products are non-exponential. There are 5 workstations, and each of them contains a flexible number of machines. These machines can perform serial batches with at most two products of the same type to save the machine setup time. The total number of machines is 37, and the number of machines in each workstation must be between 5 and 10. By enumeration, it is straightforward to verify there are 780 feasible machine allocation plans. The objective is to determine the machine allocation plan that can minimize the average cycle time of the manufacturing system.

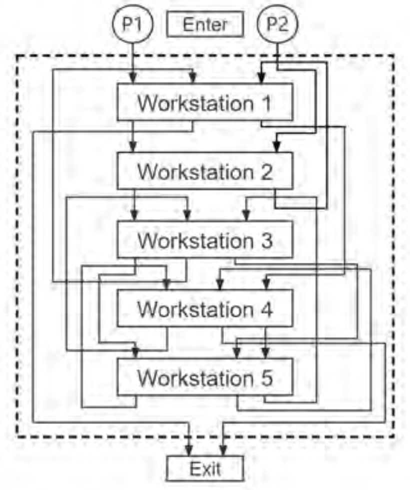
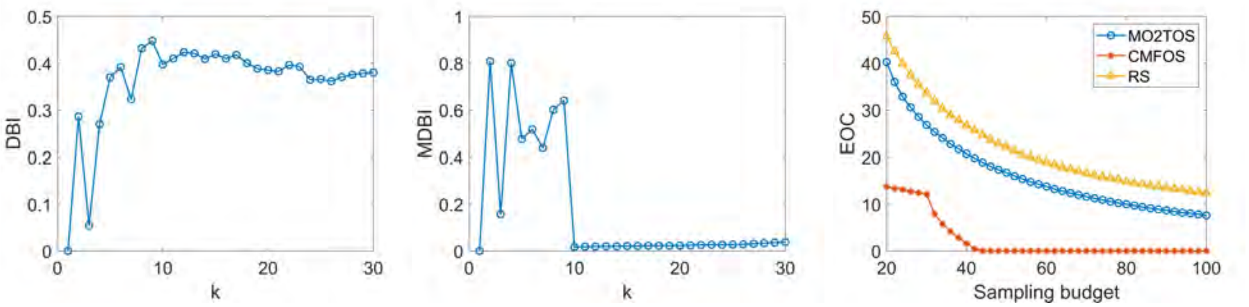


Figure 5: Jobs flow through a manufacturing system with two products.

A discrete-event simulation (DES) model is constructed to fully capture the complex features of the manufacturing system, such as the re-entrant jobs flow, priority scheduling, etc. The DES model is a high-fidelity model and provides accurate estimates for the performance of machine allocation plans. The low-fidelity model simplifies the manufacturing system and provides approximate estimates for the performance of machine allocation plans. We set T, the sampling budget for identifying the best cluster, as 10. In this experiment, there are 780 designs, and the correlation ρ between $h(\cdot)$ and $l(\cdot)$ for this tested example can be calculated as 0.94.

Figure 6 illustrates the two cluster validity indices and performance comparisons of the three tested procedures on the machine allocation problem. According to Figure 6a and 6b, we take $k = 10$. As shown in Figure 6c, CMFOS performs the best, followed by MO^{2}TOS, and RS has the worst performance. This observation corresponds to the results in the synthetic example and both benchmark test function examples, and it verifies the superior efficiency and applicability of the proposed CMFOS.



(a) Davies-Bouldin index. (b) Modified Davies-Bouldin index. (c) Performance comparison.

Figure 6: Results for the machine allocation problem.

6 CONCLUSION

In this paper, we propose a cluster-based multi-fidelity optimal sampling (CMFOS) procedure that utilizes information from the low-fidelity model to further enhance the efficiency of searching for the best design. The proposed CMFOS algorithm is flexible, i.e., allows unequal-sized-groups clustering; can determine the optimal number of clusters; achieves a balance between exploration and exploitation; and is computationally efficient. Furthermore, the sampling allocation determined by CMFOS can approximately minimize the EOC of the selected best design. Numerical experiments, including a synthetic example, two benchmark functions, and also a machine allocation problem, substantiate the superior efficiency and applicability of the proposed CMFOS.

ACKNOWLEDGMENTS

This research paper has been made possible by the funding support from the Singapore Maritime Institute & the Center of Excellence in Modelling and Simulation for Next Generation Ports (Singapore Maritime Institute grant: SMI-2022-SP-02).

REFERENCES

Chen, C.-H., J. Lin, E. Yücesan, and S. E. Chick. 2000. “Simulation Budget Allocation for Further Enhancing the Efficiency of Ordinal Optimization”. *Discrete Event Dynamic Systems* 10(3):251–270.

Davies, D. L., and D. W. Bouldin. 1979. “A Cluster Separation Measure”. *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-1(2):224–227.

Fernández-Godino, M. G., C. Park, N. H. Kim, and R. T. Haftka. 2019, may. “Issues in Deciding Whether to Use Multifidelity Surrogates”. *AIAA Journal* 57(5):2039–2054.

Forrester, A. I., A. Sóbester, and A. J. Keane. 2007. “Multi-fidelity Optimization via Surrogate Modelling”. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 463(2088):3251–3269.

Gutmann, H.-M. 2001. “A Radial Basis Function Method for Global Optimization”. *Journal of Global Optimization* 19(3):201–227.

Huang, D., T. T. Allen, W. I. Notz, and R. A. Miller. 2006. “Sequential Kriging Optimization Using Multiple-fidelity Evaluations”. *Structural and Multidisciplinary Optimization* 32(5):369–382.

Li, H., Y. Li, L. H. Lee, E. P. Chew, G. Pedrielli, and C.-H. Chen. 2015. “Multi-objective Multi-fidelity Optimization with Ordinal Transformation and Optimal Sampling”. In *Proceedings of the 2015 Winter Simulation Conference*, edited by

L. Yilmaz, W. K. V. Chan, I. Moon, T. M. K. Roeder, C. Macal, and M. D. Rossetti, 3737–3748. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Likas, A., N. Vlassis, and J. J. Verbeek. 2003. “The Global K-means Clustering Algorithm”. *Pattern Recognition* 36(2):451–461.

Mainini, L., A. Serani, M. P. Rumpfkeil, E. Minisci, D. Quagliarella, H. Pehlivan, S. Yildiz, S. Ficini, R. Pellegrini, F. Di Fiore et al. 2022. “Analytical Benchmark Problems for Multifidelity Optimization Methods”. *arXiv preprint arXiv:2204.07867*.

March, A., and K. Willcox. 2012. “Provably Convergent Multifidelity Optimization Algorithm not Requiring High-fidelity Derivatives”. *AIAA Journal* 50(5):1079–1089.

Peng, Y., J. Xu, L. H. Lee, J. Hu, and C.-H. Chen. 2018. “Efficient Simulation Sampling Allocation Using Multifidelity Models”. *IEEE Transactions on Automatic Control* 64(8):3156–3169.

Song, J., Y. Qiu, J. Xu, and F. Yang. 2019. “Multi-fidelity Sampling for Efficient Simulation-based Decision Making in Manufacturing Management”. *IISE Transactions* 51(7):792–805.

Späth, H. 1984. “Cluster-Formation Und-Analyse-Ein Zweiter Blick”. In *DGOR*, 300–300. Springer.

Teboulle, M. 2007. “A Unified Continuous Optimization Framework for Center-Based Clustering Methods”. *Journal of Machine Learning Research* 8(1):65–102.

Toal, D. J. 2015. “Some Considerations Regarding the Use of Multi-fidelity Kriging in the Construction of Surrogate Models”. *Structural and Multidisciplinary Optimization* 51(6):1223–1245.

Wang, S., S. H. Ng, and W. B. Haskell. 2022. “A Multilevel Simulation Optimization Approach for Quantile Functions”. *INFORMS Journal on Computing* 34(1):569–585.

Xu, J., S. Zhang, E. Huang, C.-H. Chen, L. H. Lee, and N. Celik. 2016. “MO2TOS: Multi-fidelity Optimization with Ordinal Transformation and Optimal Sampling”. *Asia-Pacific Journal of Operational Research* 33(03):1650017.

Xu, J., and Z. Zheng. 2023. “Gradient-Based Simulation Optimization Algorithms via Multi-Resolution System Approximations”. *INFORMS Journal on Computing* 35(3):633–651.

Zhang, S., J. Xu, E. Huang, C.-H. Chen, and S. Gao. 2016. “Improving Ordinal Transformation Through Optimal Combination of Multi-model Predictions”. In *2016 IEEE International Conference on Industrial Technology (ICIT)*, 1545–1549. IEEE.

AUTHOR BIOGRAPHIES

ZIRUI CAO is a Ph.D. candidate in the Department of Industrial Systems Engineering and Management at National University of Singapore. He received his B.Mgt. degree in the Department of Management Science and Engineering from Southeast University in 2020. His research interests include simulation optimization, parallel computing, and large-scale optimization. His email address is zirui@u.nus.edu.

HAOWEI WANG is currently a research scientist at Rice-Rick Digitalization PTE. Ltd. Before joining Rice-Rick, he received the B.Eng. degree in industrial engineering from Nanjing University in 2016 and the Ph.D. degree in industrial and systems engineering from National University of Singapore (NUS) in 2021. His research interest includes simulation optimization, Bayesian optimization under uncertainties. His email address is haowei_wang@ricerick.com.

HAOBIN LI is a Senior Lecturer in the Department of Industrial Systems Engineering and Management at National University of Singapore, and he received his Ph.D. degree in 2014 from the same department. He has research interests in operations research and simulation optimization with applications in logistics and maritime studies. His email address is li_haobin@nus.edu.sg.

EK PENG CHEW is a Professor and Deputy Head of the Department of Industrial Systems Engineering and Management, National University of Singapore. He received his Ph.D. degree from the Georgia Institute of Technology, USA. His research interests include logistics and inventory management, system modeling and simulation. His email address is isecep@nus.edu.sg.

KOK CHOON TAN is an Associate Professor and Deputy Head of the Department of Analytics and Operations, National University of Singapore. He received his Ph.D. degree from the Massachusetts Institute of Technology, USA. His research interests include logistics, supply chain optimization, and system modeling. His email address is kokchoon@nus.edu.sg.