Bi-objective Ranking and Selection Using Stochastic Kriging

Sebastian Rojas Gonzalez^{a,c,*}, Juergen Branke^b, Inneke Van Nieuwenhuyse^{a,c}

^aDepartment of Information Technology, Ghent University, Belgium. ^bDepartment of Operations, Warwick University, United Kingdom ^cFlandersMake@UHasselt and Data Science Institute, Hasselt University, Belgium

Abstract

We consider bi-objective ranking and selection problems, where the goal is to correctly identify the Pareto optimal solutions among a finite set of candidates for which the two objective outcomes have been observed with uncertainty (e.g., after running a multiobjective stochastic simulation optimization procedure). When identifying these solutions, the noise perturbing the observed performance may lead to two types of errors: solutions that are truly Pareto-optimal can be wrongly considered dominated, and solutions that are truly dominated can be wrongly considered Pareto-optimal. We propose a novel Bayesian bi-objective ranking and selection method that sequentially allocates extra samples to competitive solutions, in view of reducing the misclassification errors when identifying the solutions with the best expected performance. The approach uses stochastic kriging to build reliable predictive distributions of the objective outcomes, and exploits this information to decide how to resample. Experimental results show that the proposed method outperforms the standard allocation method, as well as a well-known the state-of-the-art algorithm. Moreover, we show that the other competing algorithms also benefit from the use of stochastic kriging information; yet, the proposed method remains superior.

Keywords: Multiple criteria analysis, Multiobjective Simulation Optimization, Stochastic Kriging, Multiobjective Ranking and Selection

1. Introduction

In multiobjective or multi-criteria optimization problems, the goal is to find the set of solutions that reveal the essential trade-offs between the objectives (i.e., where no single objective can be improved without negatively affecting any other objective). These solutions are referred to as non-dominated or Pareto-optimal, and form the Pareto set, also referred to as the efficient set; the evaluation of these solutions in the objectives corresponds to the Pareto front (see Figure 1 for some examples of continuous fronts). Depending on the type of problem, the Pareto front may have different geometries (e.g. concave, convex, linear, disconnected). In real-life problems, the location and geometry

^{*}Corresponding author

Email address: sebastian.rojasgonzalez@ugent.be (Sebastian Rojas Gonzalez)

of the Pareto front are evidently unknown, and a discrete set of solutions is often used to approximate it (Miettinen, 1999; Hunter et al., 2019).



Figure 1: Bi-objective (left) and tri-objective (right) Pareto fronts with different geometries.

This article considers problems where the outcomes of the objective functions could only be measured through stochastic experiments (e.g., stochastic simulation), i.e., the observations are noisy. Such problems occur frequently in real life. For instance, in manufacturing and distribution problems (e.g., lot sizing in manufacturing plants), it is common to maximize the expected profit of the production system, while also minimizing profit risk; in transportation problems (e.g., constructing optimal routes for dial-a-ride services) it is common to minimize tardiness of deliveries while also using minimal resources; and in the healthcare sector it is often equally important to minimize the waiting time of patients and the idle time of doctors. We assume that a fixed set of candidate solutions is available, for which the objective outcomes have been evaluated (with observational noise). Such a set may result, for instance, from running a multi-objective metaheuristic (see, e.g., Fieldsend & Everson 2015), or a multiobjective simulation optimization procedure (see e.g., Li et al. 2015). The goal of *multi-objective ranking and selection* (MORS) then is to correctly identify the true Pareto-optimal solutions among such a finite set, by adding extra replications to solutions in the most informative way.

In the current literature on multi-objective stochastic simulation optimization (see e.g., Horn et al. 2017; Feliot et al. 2017; Hernández-Lobato et al. 2016), this identification phase is often neglected, and the observed mean performance is directly used to determine the Pareto-optimal set. Evidently, sampling variability may then lead to two types of identification errors: designs that are truly Pareto-optimal can be wrongly considered dominated, or designs that are truly dominated can be considered Pareto-optimal. Chen & Lee (2010) refer to these errors as *Error Type 1* and *Error Type 2* respectively, whereas Hunter et al. (2019) refer to them as misclassification by exclusion (MCE) and misclassification by inclusion (MCI). Throughout this paper, we use the MCE/MCI terminology. The accuracy of the objective estimates can obviously be improved by allocating extra samples, yet it is natural to assume that the resampling budget is limited. Allocating this computational budget among the candidates is not trivial. *Static resampling* (allocating the resampling budget uniformly across all alternatives) tends to be inefficient, in particular in settings with limited budget and complex noise structures, as it often wastes a lot of replications on inferior solutions (Rojas-Gonzalez et al., 2020).

Literature on the MORS problem is relatively recent, and still very limited (Hunter

et al., 2019). In this paper, we propose a sequential MORS method that uses *stochastic kriging* (SK) metamodels (Ankenman et al., 2010) to build predictive distributions of the objectives, based on the sample means and variances after replication. This allows the proposed method to exploit correlations in the objective outcomes across different solutions, even in the presence of heteroscedastic noise. Under a few mild conditions, SK metamodels provide reliable predictions that are superior to using sampling information directly (Staum, 2009), as the noise in the observed performance is considered in the predictions. We exploit the SK information in a novel heuristic to determine which solutions to resample, and show how other MORS algorithms can also benefit from using such metamodel information. In addition, we propose two screening procedures to reduce the computational burden at each iteration. While some preliminary ideas of the method presented here were published in Rojas-Gonzalez et al. (2019), all the allocation decisions we put forward in this paper are novel and shown competitive against the state-of-the-art.

The remainder of this article is organized as follows: Section 2 gives an overview of the relevant literature. Section 3 defines the problem and Section 4 gives the necessary stochastic kriging background. Section 5 explains the proposed sampling criteria and screening procedures; the proposed algorithm, referred to as SK-MORS, is outlined in Section 5.3. We design the experiments to evaluate the performance of the algorithm in Section 6, analyze the results in Section 7 and conclude in Section 8.

2. Related work

The ranking and selection problem has been widely studied in the single-objective case, and has been approached in different ways. One of the most common goals is to maximize the *probability of correct selection* (PCS), where under some mild conditions, convergence to the solution with the true best expected performance can be theoretically guaranteed (see e.g., Boesel et al. 2003; Kim & Nelson 2006; Frazier 2014). In some cases, however, an extremely large replication budget might be required to differentiate two solutions with almost identical performance, while such small differences in performance are likely not relevant to the decision maker. Therefore, *indifference zone* (IZ) approaches have been developed which give a probabilistic guarantee on the selection of the best solution *or* a solution within a given user-defined quantity from the best (known as the *probability of good selection*). This user-specified quantity defines the indifference zone, and represents the smallest difference worth detecting (see e.g., Kim & Nelson 2001; Boesel et al. 2003; Fan et al. 2016).

Relative to the single-objective case, the literature on multiobjective ranking and selection is scarce; a good overview can be found in Hunter et al. (2019). The multi-objective PCS (hereafter denoted mPCS) is defined as the probability of correctly identifying the entire Pareto set, and only this set (Branke & Zhang, 2019). For both the single and multiobjective case, the true PCS must be estimated, usually via Monte Carlo simulation, which often leads to a computational bottleneck (Chen & Lee, 2010). The multiobjective case is evidently more complex, as not only the true Pareto front is often continuous and unknown (and thus approximated with a relatively large discrete set), but the dominance relationships between solutions are extremely sensitive to simulation replications (see e.g. the results in Rojas-Gonzalez et al. 2020).

One of the most widely used MORS methods is the *Multiobjective Optimal Computing Budget Allocation* (MOCBA), proposed in Lee et al. (2010), built upon the well-known single-objective ranking and selection approach OCBA (Chen & Lee, 2010). As discussed by the authors, the MORS problem can be formulated in several ways (e.g., maximizing the mPCS or minimizing the classification errors when identifying the non-dominated set). The proposed allocation rules are complex, and use numerical approximations to estimate which solutions are likely to be dominated. In Chen & Lee (2010) a simplified version of the original MOCBA is proposed, which avoids most of these issues. Another relevant framework is the *SCORE allocations for bi-objective ranking and selection* (Feldman et al., 2015; Feldman & Hunter, 2018), which aims to allocate replications to maximize the rate of decay of the probability of wrongly identifying the true Pareto set, and is asymptotically optimal. Later in Applegate et al. (2020), the work was extended to handle 3 and 4 objectives, showing promising results despite the high computational cost.

More recently, Andradóttir & Lee (2021) proposed a multi-objective IZ procedure to estimate the Pareto set with the true best expected performance with statistical guarantees. The work focuses on providing a stopping criterion for replicating designs until a predetermined desired mPCS is achieved, which is desirable in some MORS settings, but at the expense of a large replication budget due to the approximation of several parameters. Another related work appears in Binois et al. (2015), where by means of kriging metamodels, predictive distributions are built over the objectives and conditional simulations are used to estimate the probability that any given point in the objective space is dominated; the method is found to be very sensitive to the (unknown) geometry of the Pareto front, and is computationally expensive. To the best of our knowledge, there are only two other multi-objective IZ procedures in the literature: Teng et al. (2010) and Branke & Zhang (2019). The latter work clearly highlights the shortcomings of the first, but remains limited to the biobjective case, and is computationally demanding.

A different approach appears in the M-MOBA and M-MOBA-HV algorithms (Branke et al., 2016; Branke & Zhang, 2019), which apply a Bayesian approach to determine the solution that, when replicated further, is expected to yield the maximum *information value* (Chick et al., 2010). In the M-MOBA algorithm, this solution is the one with the highest probability of changing the current Pareto set, whereas in M-MOBA-HV, it is the one leading to the largest change in the observed hypervolume (a widely used performance metric in deterministic multiobjective optimization, that quantifies the volume of the objective space dominated by a given set of points; see Zitzler et al. 2007; Auger et al. 2012 for further details on hypervolume).

Another related stream of research comes from the *evolutionary multiobjective optimization* community, which has to rank the solutions in the population in each iteration, and thus needs to solve a multi-objective ranking and selection problem in each iteration if the underlying optimization problem is noisy. However, this problem is often neglected and the observed mean performance is used to rank the solutions in each iteration. Several authors have looked at formalizing the concept of dominance in noisy multi-objective optimisation (see e.g., Teich 2001; Hughes 2001; Fieldsend & Everson 2005; Basseur & Zitzler 2006). In Trautmann et al. (2009) and Voß et al. (2010), probabilistic dominance is defined by comparing the volume in the objective space of the confidence intervals, and the center point of these volumes is used to determine the dominance relationship. Another related approach is *dynamic resampling*, which varies the additional number of samples based on the estimated variance of the observed objective values. It aims to assess the observed responses at a particular confidence level before determining dominance, and to avoid unnecessary resampling (see e.g., Di Pietro et al. (2004); Syberfeldt et al. (2010)). A relatively simple yet well performing approach is to simply allocate additional samples to solutions that survive from one generation to the next, as is proposed in the Rolling Tide Evolutionary Algorithm (Fieldsend & Everson, 2015).

3. Problem definition

A multiobjective optimization problem can be defined as follows: $\min[f_1(\mathbf{x}), ..., f_m(\mathbf{x})]$, for $m = \{2, 3\}$ objectives. The solution to this problem is a discrete set of *decision vectors* $\mathbf{x}_i = [x_1, ..., x_d]^T$, i = 1, ..., n, also referred to as *designs* or *points*, which are contained in the decision space \mathcal{X} (usually $\mathcal{X} \subset \mathbb{R}^d$), with $f : \mathcal{X} \to \mathbb{R}^m$ the vector-valued function with coordinates $f_1, ..., f_m$ in the objective space. In the stochastic case, the objectives are perturbed by observational noise: $f_j(\mathbf{x}) = f_j(\mathbf{x}) + \epsilon_j$, j = 1, ..., m. The observational noise, denoted by ϵ_j , j = 1, ..., m, is commonly assumed to be independent among the different objectives and identically distributed across replications. In practice, the noise tends to be heteroscedastic (Kim & Nelson, 2006; Ankenman et al., 2010); its level is dependent on the decision variables and thus varies throughout the search space: $f_j(\mathbf{x}) =$ $f_j(\mathbf{x}) + \epsilon_j(\mathbf{x})$, j = 1, ..., m. In this paper, we focus on the bi-objective case (i.e., m = 2objectives); to simplify notation, we sometimes use f_{ij} to denote the noisy performance $f_j(\mathbf{x}_i) = f_j(\mathbf{x}_i) + \epsilon_j(\mathbf{x}_i)$ of point \mathbf{x}_i on objective j.

In a multi-objective setting, usually the concept of *dominance* is used to compare solutions:

Definition 1. For \mathbf{x}_1 and \mathbf{x}_2 two vectors in \mathcal{X} :

- $\mathbf{x}_1 \prec \mathbf{x}_2$ means \mathbf{x}_1 dominates \mathbf{x}_2 iff $f_j(\mathbf{x}_1) \leq f_j(\mathbf{x}_2), \forall j \in \{1, .., m\}$, and $\exists j \in \{1, .., m\}$ such that $f_j(\mathbf{x}_1) < f_j(\mathbf{x}_2)$
- $\mathbf{x}_1 \prec \mathbf{x}_2$ means \mathbf{x}_1 strictly dominates \mathbf{x}_2 iff $f_j(\mathbf{x}_1) < f_j(\mathbf{x}_2), \forall j \in \{1, ..., m\}$.

The set of solutions not dominated by any other solution is called the *Pareto set*, its corresponding image in the objective space is called *Pareto front*. In the noisy case, the

observed performance for a given objective at a given decision vector \mathbf{x}_i is commonly estimated by the sample mean of that objective over r_i replications: $\bar{f}_{ij} = \sum_{k=1}^{r_i} \bar{f}_{ij}^k/r_i$, j = 1, ..., m, where \bar{f}_{ij}^k denotes the performance of the k-th replication of \mathbf{x}_i on objective j. These sample means are then directly used to obtain the *observed* Pareto front, which clearly may incur MCE/MCI errors.

We thus consider the problem of efficiently identifying the designs with the true best expected performance out of a given set of alternatives, when these alternatives are evaluated on multiple objectives using a stochastic simulator. As common in the literature, we assume that the true performance measure f_{ij} for any arbitrary objective j in design ifollows a normal prior distribution, and no prior knowledge is available on the performance before conducting the simulation studies.

Given r_i independent and identically distributed replications (iid) of design *i*, the posterior distribution of the performance is then commonly estimated by

$$\{f_{ij}\} \sim \mathcal{N}(\bar{f}_{ij}, \frac{s_{ij}^2}{r_i}), \forall i = 1, ..., n; j = 1, ..., m$$
(1)

where \bar{f}_{ij} and s_{ij}^2 denote the sample mean and sample variance of objective j in design i, respectively. If every possible solution has been evaluated at least once, the *observed* Pareto set (i.e., the Pareto set based on sample means) can be determined based on the $r_i, i = 1, ..., n$ replications available so far on each objective. After an additional r'_i simulation replications are performed, yielding an average performance of \bar{f}'_{ij} , the observed mean is updated as (Branke & Zhang, 2019):

$$\bar{f}_{ij} = \frac{r_i \bar{f}_{ij} + r'_i \bar{f}'_{ij}}{r_i + r'_i}.$$
(2)

Naturally, when the observed means are updated, the dominance relationships among some alternatives will most likely change, and in some cases change drastically (see Figure 2), thereby also changing the observed Pareto set. We thus focus on solving the following optimization problem (see also Lee et al. 2010):

$$\min_{\substack{r_1,...,r_n}} (\text{MCE}+\text{MCI})$$
(3)
s.t.
$$\sum_{i=1}^n r_i \le R$$
$$r_i \ge 0, i = 1, ..., n$$

where r_i denotes the number of replications on design *i*, *R* is the total replication budget available, and MCE and MCI are the number of misclassification errors by exclusion and inclusion, respectively. Problem 3 is analytically intractable: the exact number of errors cannot be computed, as the true Pareto optimal points are in general unknown. Furthermore, the observed sample means are random variates themselves, given that they are estimated from the samples. In this paper we present a sequential heuristic that aims to solve the problem using *stochastic kriging metamodels* (Ankenman et al., 2010). In what follows, we first present the important background of stochastic kriging in Section 4, succeeded by the description of the proposed approach in Section 5.



Figure 2: Left panel: True performance of a given set of solutions, for a bi-objective minimization problem. Center panel: Observed performance based on the sample means after r replications per solution. Right panel: Observed performance after r + 1 replications per solution.

4. Stochastic kriging

Let $f(\mathbf{x})$ be an unknown objective function, which can only be observed with (heteroscedastic) noise:

$$\tilde{f}(\mathbf{x}_i) = f(\mathbf{x}_i) + \epsilon(\mathbf{x}_i) \tag{4}$$

Assume that we have observed this function in a finite set of d-dimensional designs \mathbf{x}_i , i = 1, ..., n, using r_i replications per design. Stochastic kriging (SK) (Ankenman et al., 2010) then represents the simulation output in replication k by the following model:

$$\tilde{f}^k(\mathbf{x}_i) = \beta + M(\mathbf{x}_i) + \epsilon^k(\mathbf{x}_i).$$
(5)

where β is a constant, and $M(\mathbf{x}_i)$ is a realization of a mean 0 random field. The random field M is assumed to exhibit spatial correlation, meaning that the outcomes $M(\mathbf{x}_i)$ and $M(\mathbf{x}_h)$ will tend to be similar when \mathbf{x}_i is close to \mathbf{x}_h in the design space. Ankenman et al. (2010) refer to the random nature of M as *extrinsic* uncertainty, as it is imposed on the problem in view of developing the model. In the design and analysis of computer experiments, the standard assumption for M is that it is a Gaussian random field. Note that, in stochastic simulation, the *intrinsic* noise terms in Eq. 5 are naturally independent and identically distributed across replications, with $E[\epsilon^k(\mathbf{x}_i)] = 0$ and $Var[\epsilon^k(\mathbf{x}_i)] = \tau^2(\mathbf{x}_i)$; as $\tau^2(\mathbf{x}_i)$ depends on the location of \mathbf{x}_i , the model naturally allows for heterogeneous noise.

The core of the extrinsic spatial correlation model is the *covariance function* or *kernel*, which is used to quantify the covariance between the outcomes of any two decision vectors \mathbf{x}_i and \mathbf{x}_h . In this work we use the (stationary) squared exponential kernel, also known as Gaussian kernel (for further details on other popular kernels, see Rasmussen & Williams

(2005), Chapter 5):

$$cov[M(\mathbf{x}_i), M(\mathbf{x}_h)] = v^2 \exp\left[-\sum_{q=1}^d \left(\frac{|x_{i,q} - x_{h,q}|}{l_q}\right)^2\right]$$
(6)

where v^2 and $l_q, q = 1, ..., d$ are hyperparameters that denote the process variance and the length-scale of the process along dimension q, respectively, for all i, h = 1, ..., n and $i \neq h$. These hyperparameters, along with the constant β in Eq. 5, need to be estimated from the available data (i.e., the finite set of design points and their corresponding sample means), usually by means of maximum likelihood estimation (MLE).

Using the resulting MLE estimates $\hat{\beta}$, \hat{v}^2 and \hat{l}_q , q = 1, ..., d, the analyst can predict the outcome for the objective function in Eq. 4, at any arbitrary \mathbf{x}_i , by means of the **stochastic kriging predictor**:

$$\hat{f}(\mathbf{x}_i) = \hat{\beta} + \Sigma_M(\mathbf{x}_i, \cdot)^T [\Sigma_M + \Sigma_\epsilon]^{-1} (\mathbf{\bar{f}} - \hat{\beta} \mathbf{1}_n)$$
(7)

where $\overline{\mathbf{f}} = [\overline{f}(\mathbf{x}_1), ..., \overline{f}(\mathbf{x}_n)]^T$ is the $n \times 1$ vector containing the sample means of the available solutions, and $\mathbf{1}_n$ is a $n \times 1$ vector of ones. The $n \times n$ matrix Σ_M contains the kernel results for each couple of available points (i.e., $cov[M(\mathbf{x}_h), M(\mathbf{x}_{h'})]$, for h, h' = 1, ..., n). Analogously, the $n \times 1$ vector $\Sigma_M(\mathbf{x}_i, \cdot)$ contains the kernel results for the given point \mathbf{x}_i and each of the n already sampled points (i.e., $cov[M(\mathbf{x}_i), M(\mathbf{x}_h)]$, for h = 1, ..., n). The $n \times n$ matrix Σ_{ϵ} contains the covariances implied by the intrinsic noise at the available points. As discussed in Ankenman et al. (2010), it is in general not recommended to use common random numbers (CRN) in the simulations when stochastic kriging is used (see also Chen et al. (2012) for further details); consequently, Σ_{ϵ} reduces to the diagonal matrix diag[$\tau^2(\mathbf{x}_1)/r_1, ..., \tau^2(\mathbf{x}_n)/r_n$]. The **predictor uncertainty** is quantified by its mean squared error (MSE), and is given by

$$\hat{s}^{2}(\mathbf{x}_{i}) = \Sigma_{M}(\mathbf{x}_{i}, \mathbf{x}_{i}) - \Sigma_{M}(\mathbf{x}_{i}, \cdot)^{T} [\Sigma_{M} + \Sigma_{\epsilon}]^{-1} \Sigma_{M}(\mathbf{x}_{i}, \cdot) + \frac{\gamma^{T} \gamma}{\mathbf{1}_{n}^{T} [\Sigma_{M} + \Sigma_{\epsilon}]^{-1} \mathbf{1}_{n}}$$
(8)
with $\gamma = 1 - \mathbf{1}_{n}^{T} [\Sigma_{M} + \Sigma_{\epsilon}]^{-1} \Sigma_{M}(\mathbf{x}_{i}, \cdot).$

Note that when Σ_{ϵ} reduces to a zero matrix, the stochastic kriging estimates in Eq. 7 and Eq. 8 reduce to the well-known ordinary kriging expressions (Jones et al., 1998; Kleijnen, 2015), as the output observations are then essentially deterministic. When that happens, the kriging predictor will perfectly coincide with the sample means at all simulated design points, and the predictor uncertainty will reduce to zero. Yet, in stochastic simulation, this would require all points in the design to be evaluated with an infinite replication budget. Consequently, in all practical settings, the SK predictor in Eq. 7 will not coincide with the sample means at the observed points; yet, the difference between them should tend to zero as more replications are added. We exploit this crucial property

in our proposed ranking and selection procedure, as discussed in the next section.

5. Proposed MORS procedure

5.1. Proposed sampling criteria

Let S be the entire set of sampled points. Relying solely on sample means, the analyst could detect an "observed" Pareto set (denoted PS) and a corresponding "observed" Pareto front (denoted PF), among the points in S. Relying on the SK predictors, also a "predicted" \widehat{PS} (and corresponding \widehat{PF}) could be distinguished. Given the presence of noise, the predicted and observed sets (and fronts) are typically not the same; yet, adding more replications will likely reduce the noise, thus reducing the difference between the sample means and the SK predictors, as argued at the end of the previous section. As opposed to previous work, the sampling criteria proposed in our approach rely on *both* types of information, to make decisions on where to allocate extra replications. The proposed criteria are referred to as *expected hypervolume difference* (EHVD), and *posterior distance* (PD). For ease of notation (analogous to the sample means and variances), for point \mathbf{x}_i on objective j, hereafter the stochastic kriging predictor and predictor uncertainties are denoted \hat{f}_{ij} and \hat{s}_{ij} , respectively. Furthermore, $\mathbf{\bar{f}}_i$ and $\mathbf{\hat{f}}_i$ denote the sample mean objective vectors and predicted objective vectors, respectively.

5.1.1. Expected hypervolume difference

The hypervolume dominated by a given Pareto front P with respect to a reference point **r** is defined as the Lebesgue measure, denoted Λ , of the set of objective vectors dominated by the solutions in P, but not by **r**:

$$HV(P,\mathbf{r}) = \Lambda\left(\bigcup_{z\in P} \{z': z \prec z' \prec \mathbf{r}\}\right).$$
(9)

Thus, all the non-dominated vectors contribute to the indicator value, and the dominated vectors do not contribute. For two fronts P and Q and a reference point \mathbf{r} , the expected hypervolume change (Branke & Zhang, 2019) is then defined as:

$$EHVC(P,Q,\mathbf{r}) := HV(P,\mathbf{r}) + HV(Q,\mathbf{r}) - 2 \times \Lambda \left(HV(P,\mathbf{r}) \cap HV(Q,\mathbf{r})\right).$$
(10)

Figure 3 illustrates the concept; for a given Pareto front (set P) based on sample means (left panel), the EHVC calculates how much this set is expected to change (set Q), under the assumption that the performance of each solution follows a certain probability distribution.

The EHVC was proposed in Branke et al. (2016) and Branke & Zhang (2019) as a criterion to allocate samples. Analytical expressions for EHVC were proposed for the biobjective case; yet, computational complexity remains high. We use the concept of EHVC to estimate the expected change in the observed HV, if more samples were allocated to a given point \mathbf{x}_i , by replacing the observed objective outcomes (i.e., $\mathbf{\bar{f}}_i$) by their SK



Figure 3: Left panel: Hypervolume (shaded area) dominated by a given non-dominated set (filled points) with respect to a reference point \mathbf{r} . Right panel: the EHVC (shaded area) of a set of non-dominated points with respect to a reference point \mathbf{r} . The filled circles denote the current observed front (set P), and the filled squares the expected new performance after new samples are allocated (set Q). Empty circles represent the dominated points.

predictors (i.e., $\hat{\mathbf{f}}_i$), as we expect these predictors to yield more accurate estimates of the true objective values. We refer to this criterion as the *expected hypervolume difference* (EHVD):

$$\mathrm{EHVD}_{i} = |HV(PF) - HV\left(PF \setminus \mathbf{\bar{f}}_{i} \cup \mathbf{\hat{f}}_{i}\right)|, \qquad \forall i \in S.$$

$$(11)$$

As mentioned before, $\overline{\mathbf{f}}_i$ and $\mathbf{\hat{f}}_i$ will converge as the number of replications for design i grows, and the noise is reduced. As the computation of the HV only uses the non-dominated points, there are five cases to be considered:

- Case 1: $\mathbf{f}_i \prec \mathbf{f}_i$, and \mathbf{f}_i is on the Pareto front. The HV will change (decrease) as the observed vector dominates the predicted vector (see Figure 4(a)).
- Case 2: $\hat{\mathbf{f}}_i \prec \bar{\mathbf{f}}_i$, and $\hat{\mathbf{f}}_i$ is on the Pareto front. The HV will change (increase) as the predicted vector dominates the observed vector (see Figure 4(b)).
- Case 3: $\mathbf{\bar{f}}_i \prec \mathbf{\hat{f}}_i$, and there are one or more points that dominate $\mathbf{\hat{f}}_i$ (denoted $\mathbf{\bar{g}}$ in Figure 4(c)). The HV will change (decrease) as the sample mean vector dominates the predicted vector, but the change is not equally significant as in Case 1.
- Case 4: $\hat{\mathbf{f}}_i \prec \overline{\mathbf{f}}_i$, and there are one or more points that dominate $\overline{\mathbf{f}}_i$ (denoted $\overline{\mathbf{g}}$ in Figure 4(d)). The HV will change (increase) as the predicted vector dominates the sample vector, but the change is not equally significant as in Case 2.
- Case 5: $\overline{\mathbf{f}}_i \prec \widehat{\mathbf{f}}_i$ or viceversa, and both performance vectors are observed in the dominated space. The HV will thus not change and will not impact the EHVD (see Figure 4(e)).

Clearly, the EHVD will tend to allocate more budget to points belonging to Cases 1 and 2; points belonging to Cases 3-4 may have little or no replications allocated, while points in Case 5 will not be considered, though they may as well belong to the true non-dominated set. This is true especially when the noise is high and/or strongly heterogeneous, and

thus the sample and predicted means are both relatively far from the true performance; indeed, an extra replication on a given point might then significantly change its position in the objective space.



Figure 4: The five cases to be considered when calculating the EHVD.

5.1.2. Posterior distance

The EHVD looks only at the points that are likely to change the current HV the most. In a ranking and selection context, all the points sampled have a probability of belonging to the true Pareto front, as they are relatively close to each other. Thus, we propose a criterion to improve the position in objective space of any point in S. We define the *posterior distance (PD)* between the observed and predicted means as:

$$PD_{i} = \sqrt{\sum_{j=1}^{m} \left[\left| \bar{f}_{j}(\mathbf{x}_{i}) - \hat{f}_{j}(\mathbf{x}_{i}) \right| + \hat{s}_{ij} \right]^{2}}, \qquad \forall i \in S.$$
(12)

The PD is the Euclidean distance between the sample means and the predicted means (also referred to as the *posterior means* of the model). If the sample mean and prediction are far away from each other, by running more replications on these points, we expect them to come closer to each other and move towards the true position of the point in the objective space, which aids in minimizing *both* MCE and MCI errors. Note that in Equation 12, we add the posterior uncertainty to the distance between the means, as we intend to inflate the PD when the uncertainty of the predictor is high, such that points with high uncertainty will be rewarded more (see Figure 5 for an illustration).



Figure 5: Left panel: Sample means (black dots), posterior means (red line) and posterior uncertainty (gray area) for a single objective. The distance between the prediction and the sample mean is denoted with a blue line. Right panel: The posterior distance between the sample means (circles) and predicted means (squares). The observed and predicted fronts are depicted with filled circles and squares respectively.

5.2. Proposed screening procedures

As is well-known in ranking and selection, in many cases some of the points included in the solution set will be clearly inferior to other solutions in the set (Boesel et al., 2003). Screening or subset-selection heuristics are widely used prior to running the ranking and selection procedure in order to filter out these clearly inferior solutions. We propose two subset selection procedures that will exploit both the sample variance and prediction uncertainties to enclose confidence regions that reduce the number of points to be considered at each iteration. To do this, we use the *lower confidence bounds* (*LCB*) and *upper confidence bounds* (*UCB*) of the mean performance, defined as $\bar{f}_j(\mathbf{x}) \pm \omega s_j(\mathbf{x}), j = 1, ..., m$, where ω is usually a number in the interval [1,3], to yield the 60%-99% confidence intervals (CI) (Rasmussen & Williams, 2005). Analogously, the CI of the predicted means (denoted \widehat{LCB} and \widehat{UCB}) are defined as $\hat{f}_j(\mathbf{x}) \pm \omega \hat{s}_j(\mathbf{x}), j = 1, ..., m$. Unless stated otherwise, hereafter we will use a value of $\omega = 3$, in order to enclose the 99% CI (i.e., 3 standard deviations from the mean).

The first of the proposed screening procedures is outlined in Algorithm 1. The worst confidence bounds (i.e., the side of the bound that worsens the performance) among all the non-dominated points for each objective j, denoted \bar{u}_j for the sample means, and \hat{u}_j for the predicted means, are used to enclose a confidence region (see left panel of Figure 6). The second screening procedure, outlined in Algorithm 2, will instead enclose a confidence region using the upper confidence bounds of *all* the non-dominated points (see right panel of Figure 6). These confidence regions are computed for *both* the *PF* and \widehat{PF} .

Algorithm 1 Screening Box 1: Input: UCB, LCB, \widehat{UCB} and \widehat{LCB} 2: $\bar{S} = \emptyset$ ▷ Initialize set of clearly inferior points 3: \bar{u}_j : max_{*i*\in PF} UCB_{*i*}, *j* = 1, ..., *m* 4: \hat{u}_j : $\max_{i \in \widehat{PF}} \widehat{UCB}_i, j = 1, ...m$ 5: $NP = S \setminus PS \cap S \setminus \widehat{PS}$. 6: for $i \in NP$ do 7: for j = 1 : m do if $LCB_{ij} > \bar{u}_j$ and $\widehat{LCB}_{ij} > \hat{u}_j$ then 8: $\bar{S} \cup \{\mathbf{x}_i\}$ 9: break 10: end if 11: 12:end for 13: end for 14: return \bar{S}



Figure 6: Illustration of both screening procedures proposed: SCREENING BOX (left panel) and SCREENING BAND (right panel). Filled marks denote the non-dominated points, and unfilled marks the dominated points. The performance is denoted with circles, and confidence bounds with squares.

Subsequently, the performance of every dominated mean objective point (visualized with empty circles) is substituted by the lower confidence bounds (visualized with empty squares). Similarly, the performance of the non-dominated points (denoted with filled

Algorithm 2 Screening Band

1: Input: UCB, LCB, \widehat{UCB} and \widehat{LCB} 2: $\bar{S} = \emptyset$ \triangleright Initialize set of clearly inferior points 3: $NP = S \setminus PS \cap S \setminus \widehat{PS}$ 4: procedure DISTANCE($\mathbf{f}_i, \mathbf{f}_k$) $h_{ik} = \sqrt{\sum_{j=1}^{m} [f_{ij} - f_{kj}]^2}$ 5: 6: return h_{ik} 7: end procedure 8: for $i \in NP$ do $\bar{h}_{ik} = \text{DISTANCE}(LCB_i, UCB_k), \forall k \in PF$ 9: $\hat{h}_{ik} = \text{DISTANCE}(\widehat{LCB}_i, \widehat{UCB}_k), \forall k \in \widehat{PF}$ 10: 11: end for 12: for $i \in NP$ do $u = \arg\min_{k \in PF} h_{ik}$ 13: $v = \arg\min_{k \in \widehat{PF}} h_{ik}$ 14:for j = 1 : m do 15:if $LCB_{ij} > UCB_{uj}$ and $\widehat{LCB}_{ij} > \widehat{UCB}_{vj}$ then 16:17: $S \cup \{\mathbf{x}_i\}$ break 18:end if 19:end for 20: 21: end for 22: return \bar{S}

circles) is substituted by the upper confidence bounds (denoted with filled squares). This is done for both, the performance based on sample means and the performance based on predicted means. If the new position of a given dominated point does not enter any of the confidence regions enclosed by the non-dominated points (i.e., the region enclosed by PF or \widehat{PF}), then this point is screened out from the *current* iteration (see e.g., points *a* and *b* in Figure 6). Otherwise, it will be considered in the current iteration (see e.g., points *c* and *d* in Figure 6). This is to ensure that, even with very high noise levels (as in the experiments in Section 7), with a high confidence, we are not excluding truly non-dominated solutions.

5.3. Algorithm outline

The general steps of the proposed MORS procedure are summarized in Algorithm 3; for ease of reference, we summarize the different sets of points used in Table 1. The initial step of algorithm fits a SK metamodel to each objective based on sample means and variances of the points in S, after b_0 replications performed in each point. The metamodels are then used to predict each response and respective prediction errors on these sampled points. The current observed Pareto-optimal sets are obtained based on both the sample means (PS and PF) and predicted means (\widehat{PS} and \widehat{PF}). In the next step, the upper and lower confidence bounds for each point are calculated. Using these confidence bounds, if a screening procedure is invoked (lines 11-13 in Algorithm 3), the algorithm filters out

from the *current* iteration the points that are observed as clearly inferior. Note that these points might be considered in further iterations.

Notation	Description
S	Entire set of sampled points.
PS	Pareto set based on sample means.
PF	Pareto front based on sample means.
\widehat{PS}	Pareto set based on predicted means.
\widehat{PF}	Pareto front based on predicted means.
UCB	Set of the upper confidence bounds of all points in PS .
\widehat{UCB}	Set of the upper confidence bounds of all points in \widehat{PS} .
LCB	Set of the lower confidence bounds of all points in $S \setminus PS$.
\widehat{LCB}	Set of the lower confidence bounds of all points in $S \setminus \widehat{PS}$.

Table 1: Overview of the different sets of points used in the proposed algorithms.

After screening out a subset of clearly inferior points, the algorithm proceeds to calculate the EHVD values using Equations (9) and (11), and the PD values using Equation (12), only on the points that have not reached the maximum number of replications (i.e., b_{max} replications). As illustrated in Figure 7, a Pareto front can be distinguished (denoted \widetilde{PF}), for which both criteria are maximized. The algorithm then iterates over this nondominated set (sorted in ascending order according to the number of replications allocated to each point so far r_i), allocating one replication per point, until the budget B is depleted. Note the number of points in this Pareto front can differ per iteration.

Selecting the parameters b_0 , B and b_{max} will depend on the specific budget and number of points in a given problem. As discussed in Lee et al. (2010), a value of $b_0 >= 5$ is recommended to approximate the sample means and variances, and b_{max} is used to prevent the algorithms to allocate an extreme number of replications to a given point. In our experiments, we use a value of B = |S| (i.e., the budget per iteration equals the cardinality of the set of points). This is because the standard EQUAL allocation uniformly distributes the replication budget among *all* alternatives, and thus a budget of B = |S|will run 1 additional replication per point per iteration. Furthermore, a value of $b_0 = 5$ and $b_{max} = 100$ are used in our experiments for all competing algorithms.

Recall that the two criteria serve different goals: while the EHVD allocates budget to those points that are expected to change the current HV the most, the PD focuses on improving the prediction accuracy of the performance. As it will be shown in the experimental results, allocating extra samples based on the trade-off between EHVD and PD yields better results than using either of the criteria individually. If the stopping criterion is met, the algorithm returns the *predicted* Pareto set. As the proposed allocation procedure is sequential, after the budget in the current iteration has been depleted, the SK parameters are recomputed with the new (more accurate) sample means and variances to perform a new iteration. Algorithm 3 SK-MORS algorithm 1: Input: 2: S \triangleright Set of candidate points \triangleright Initial replication budget per point 3: b_0 4: b_{max} ▷ Maximum replication budget per point \triangleright Replication budget to be allocated per iteration 5: B6: Output: 7: PS \triangleright The observed Pareto set 8: Perform b_0 replications on all points in S 9: $S_{max} = \emptyset$ \triangleright Initialize set of points that reached b_{max} 10: while stopping criterion not met do $Mat_{reps} \leftarrow$ Matrix of sample means and variances for all candidates 11: $f_j(\mathbf{x}) \leftarrow$ Fit a SK metamodel to each objective j using Mat_{reps} 12: $Mat_{preds} \leftarrow Matrix of predicted means and MSEs for all candidates$ 13:Compute UCB, LCB, $UC\overline{B}$ and $LC\overline{B}$ 14: **procedure** SCREENING(UCB, LCB, \widehat{UCB} , \widehat{LCB}) 15: $\bar{S} \leftarrow$ Set of points screened out based on sample means 16: $\widehat{S} \leftarrow \text{Set of points screened out based on predicted means}$ 17: $\widetilde{S} = S \setminus (\overline{S} \cap \widehat{S})$ ▷ Temporarily remove clearly inferior points 18:end procedure 19: $\tilde{S} = \tilde{S} \setminus S_{max}$ 20: \triangleright Remove points that have reached b_{max} Compute EHVD_i and PD_i, $\forall i \in \widetilde{S}$ \triangleright Eq. 11 and 12 21: $\overline{PF} \leftarrow \text{Pareto front of } S$ ▷ Maximization of PD and EHVD 22: $b \leftarrow$ Vector of the number of replications allocated to each point in \overline{PF} 23: $PF_{sort} \leftarrow PF$ sorted in ascending order of b 24: \triangleright Prioritize allocating to solutions with low bwhile \overline{PF}_{sort} is not empty do 25:26:for $i \in PF_{sort}$ do if B > 0 and $r_i < b_{max}$ then 27:Allocate 1 additional replication on \mathbf{x}_i \triangleright Update Mat_{rens} 28:B = B - 129:else if $r_i == b_{max}$ then 30: 31: $S_{max} \cup \{\mathbf{x}_i\}$ else 32: break 33: end if 34: end for 35: $\widetilde{PF}_{sort} = \widetilde{PF}_{sort} \backslash S_{max}$ 36: end while 37:38: end while 39: Return the PS

6. Test problems

In Section 6.1, we present the implementation details of our experiments on a set of well-known analytical bi-objective test functions. These functions have different Pareto front geometries, and were implemented with varying number of decision variables, and



Figure 7: Pareto front between the maximization of both criteria (normalized in the [0, 1] interval) at two arbitrary iterations.

different levels of noise. They are used to (1) evaluate the performance of the proposed allocation criteria against EQUAL allocation and the MOCBA algorithm (Lee et al., 2010), which we consider state-of-the-art; in Appendix A we provide the allocation ratios, but the full procedure is as described in (Chen & Lee, 2010)), (2) test the impact of the two screening procedures proposed, and (3) examine the impact of using the SK information instead of the sample means and variances. In addition, in Section 6.2 we evaluate the performance of our proposed algorithm against EQUAL and MOCBA on a real life supply chain problem that we obtained from one of our industrial partners.

All three algorithms are stopped when a predetermined number of samples have been allocated; this is a natural criterion, as in most practical settings, the sampling budget will determine the length of the run. We compare their performances by means of the *accuracy of the Pareto set (APS)* metric:

$$APS = 1 - \frac{MCE + MCI}{|S|} \tag{13}$$

where MCI and MCE are the number of misclassification errors by inclusion and exclusion, respectively, and |S| is the total number of designs. The metric is an intuitive measure to evaluate the accuracy of the Pareto set identified by the algorithms. Evidently, it takes a value between 0 and 1: if the algorithm correctly identifies the entire true Pareto set, there is no misclassification of points, and APS = 1. Note that APS implicitly considers both errors to be equally important, which in the MORS context is a desirable property for algorithm evaluation.

6.1. Artificial test problems

We choose three standard test problems from the *deterministic* multi-objective literature (WFG3, WFG4 and DTLZ7) to assess the performance of the proposed procedures. The WFG test suite is state-of-the-art and allows the analyst to construct problems with any number of objectives and decision variables; features such as modality and separability can also be customized, using a set of shape and transformation functions. WFG3 is a non-separable and unimodal problem with a linear Pareto front, WFG4 is a separable and multimodal problem with a concave Pareto front, and DTLZ7 is a disconnected and multi-modal problem (see Huband et al. 2006 for the analytical expressions and detailed characteristics of these problems). The set of points S being considered is discrete and contains a fixed and known number of truly Pareto optimal points, denoted PS_t . Table 2 summarizes the experimental scenarios, and Figure 8 illustrates the objective space of the test functions.



Figure 8: True Pareto front $(PF_t, \text{ black points})$ and dominated points in the bi-objective scenarios.

The true objective outcomes are perturbed with heterogeneous Gaussian noise. Hence, we obtain noisy observations $\tilde{f}_j^k(\mathbf{x}_i) = f_j(\mathbf{x}_i) + \epsilon_j^k(\mathbf{x}_i)$, with $\epsilon_j^k(\mathbf{x}_i) \sim \mathcal{N}(0, \tau_j(\mathbf{x}_i))$ for j = $\{1,2\}$ objectives at the k^{th} replication. In accordance with the previous literature (Picheny et al., 2013; Jalali et al., 2017; Rojas-Gonzalez et al., 2020), we set the standard deviation of the noise $(\tau_i(\mathbf{x}))$ such that it varies linearly with respect to the objective values. The maximum and minimum values of $\tau_j(\mathbf{x})$ are linked to the range of each objective value in the region of interest (i.e., $RF_j = \max_{\mathbf{x}\in S} f_j(\mathbf{x}) - \min_{\mathbf{x}\in S} f_j(\mathbf{x}), j = \{1,2\}$). In the experiments, we set the minimum noise at the individual optima (i.e., the minima) of both objective functions. Because of the trade-off between the function values in the Pareto-optimal points, this assumption automatically leads to a trade-off in the *noise* of the functions at these points (in the extremes of the bi-objective front, the noise is minimal for one objective, while maximal for the other). In that way, the linear structure ensures that the resulting R&S problems are not trivial, as *none* of the Pareto-optimal points has accurate sample means on *both* objectives. Evidently, the same could have been achieved with other monotonous noise structures. As shown in Table 2, we consider three levels of noise perturbing the responses (low, medium and high), varying between $0.001RF_j$ and $2RF_i$.

6.2. Practical application from industry

We also test the competing algorithms on a real-life multi-period supply chain problem, obtained from one of our industrial partners in the chemical process industry. The company has several plant sites, each of them equipped with several processing units for production. Each of these units has unique processing capabilities and is responsible for a different category of products. Each category consists of products showing similarities in

	WFG3	WFG4	DTLZ7
Number of objectives m	2	2	2
Number of decision variables d	5	5	2
Number of points in $S(S)$	100	100	100
Size of the true Pareto set $ PS_t $	20	20	50
Noise level	High	Medium	Low
Total number of iterations	15	30	30
Stochastic kriging metamodels	Squared exponential kernel: Eq. 6		
Low noise std. dev.	$0.001RF_j \le \tau_j(\mathbf{x}) \le 0.5RF_j$		
Medium noise std. dev.	$0.01RF_j \le \tau_j(\mathbf{x}) \le 1RF_j$		
High noise std. dev.	$1RF_j \le \tau_j(\mathbf{x}) \le 2RF_j$		

Table 2: Summary of the experimental scenarios for the artificial test functions

their physical and chemical properties. The customers are separated into sales regions, and the demand forecasts per period (for all products) are generated at the level of these sales regions. The demand uncertainty for each product is modeled using a normal distribution, where the mean of this distribution is given by the forecast, and the standard deviation is approximated using historical data. Transportation is assumed to always be available, on each route from the sites to the regions. This problem setting has already been used as a real-life case in other articles (e.g., Jung et al. 2004), but then as a single-objective problem with a stochastic constraint (i.e., minimize the cost function subject to a service level constraint).

The company needs to decide the number of units produced per week, for each product and each facility. Any excess product is stored in a warehouse next to the production facility. Once demand in each sales region is known, it is satisfied by the available supply from the facility closest to the sales region (i.e., pairs of sales region and facility are sorted in increasing distance, and demand is satisfied in this order). Any demand that cannot be satisfied by the company is outsourced and satisfied by a third party at a higher cost. There are two objectives: minimizing total cost (consisting of production cost, outsourcing cost, transportation cost, and inventory holding cost), and maximize customer service level which is defined as the percentage of periods that the company does *not* need outsourcing to meet customer demand. In the example below we use 3 production facilities and 5 products, so a total of 15 decision variables. Based on Jung et al. (2004), we provide a more detailed formulation of the problem in Appendix B. Each simulation period is a week and the simulation runs over 1 year (i.e., 52 periods). Evidently, the demand from the customers is uncertain, and we consider different levels of uncertainty by varying the coefficient of variation of the demand, denoted χ (see Appendix B). We evaluate 4 different scenarios: $\chi = 0.1, 0.3, 0.6$ and 0.9. Table 3 summarizes the parameters of the scenarios.

	Scenario 1	Scenario 2	Scenario 3	Scenario 4
Number of points in $S(S)$	125	96	93	99
Size of the true Pareto set $ PS_t $	33	32	22	25
Coefficient of variation of demand	$\chi = 0.1$	$\chi = 0.3$	$\chi = 0.6$	$\chi = 0.9$

Table 3: Summary of the experimental scenarios for the practical problem



Figure 9: Pareto front for the simultaneous minimization of cost (denoted f_1) and negative service level (denoted f_2). The left panel shows the approximation of the true performance, and the right panel the misclassification errors at an arbitrary iteration.

7. Results

We first evaluate the performance of the proposed screening procedures in Section 7.1. In Section 7.2, we compare the SK-MORS algorithm against EQUAL allocation and the MOCBA algorithm. Furthermore, we show that resampling points based on the combination of EHVD and PD outperforms a resampling approach based on either of these criteria in isolation. In Section 7.3, we analyze the benefit of using the SK information instead of the sample means in the EQUAL allocation and MOCBA algorithms. Finally, in Section 7.4 we evaluate the performance on the practical problem. In all cases, we run the algorithms 30 times on the same instance (i.e., 30 macroreplications); in the figures, we report the average APS value obtained at each iteration of the algorithms, along with the 95% confidence intervals. Table 4 summarizes the algorithms evaluated.

7.1. Impact of the screening procedures

The impact of the proposed screening procedures is shown in Figure 10; the metric |S| shows the number of points that remain in the set after screening, at a given iteration. Both screening procedures succeed in sequentially reducing the number of candidate points considered as the observed performance becomes more accurate (see (a), (c) and (e) in Figure 10); evidently, the biggest reduction is obtained with SCREENING BAND. Such reduction significantly reduces the computational cost of executing the hypervolume calculations in Eq. 11 for every candidate point at each iteration.

As clear from Figures 10(b), (d) and (f), the noise levels for the different scenarios have an impact on the performance of the proposed algorithm w.r.t. the the convergence to the

Algorithm	Description
SKMORS _{none}	SK-MORS algorithm without using a screening heuristic.
$SKMORS_{box}$	SK-MORS algorithm using the SCREENING BOX heuristic (Algorithm 1).
SKMORS _{band}	SK-MORS algorithm using the SCREENING BAND heuristic (Algorithm 2).
SKMORS-PD	SK-MORS algorithm using only the PD criterion.
SKMORS-HV	SK-MORS algorithm using only the EHVD criterion.
EQUAL	Allocate replications uniformly to all sampled points.
EQUAL-SKi	EQUAL allocation using the SK predictions for identification.
MOCBA	Multiobjective Optimal Computing Budget Allocation algorithm.
MOCBA-SK	MOCBA algorithm using the SK predictions for allocation and identification.
MOCBA-SKi	MOCBA algorithm using the SK predictions only for identification.

Table 4: Overview of the different algorithms tested.

true front. While for low and medium levels of noise (i.e., scenarios DTLZ7 and WFG4 respectively) no particular differences are observed between both proposed methods, for a high level of noise (i.e., scenario WFG3) the performance of SCREENING BAND clearly worsens. This is not entirely surprising, as filtering out some points may result in leaving out truly non-dominated points (i.e., MCE points) in one or more iterations, especially for the BAND procedure. Moreover, in Figure 10(e), we observe that the BAND procedure screens out a very large number of points, including some of the MCE errors. This happens because the noise is low and the dominated points are relatively far from the non-dominated points. SCREENING BOX on the other hand, shows consistent performance; thus in further sections we use SK-MORS_{box} as the default proposed algorithm.

7.2. Proposed allocation criteria

Fig 11 clearly shows that, for all test problems considered, our algorithm converges much faster towards the correct identification of the true Pareto front than EQUAL or MOCBA. Note that the superiority of MOCBA vs. EQUAL is not entirely clear for all scenarios (e.g., Figures 11(a) and 11(c)), as in these scenarios MOCBA and EQUAL require a much larger budget to converge. On the other hand, the proposed algorithm shows consistent performance. It is also remarkable that in scenario DTLZ7 the SK predictions provide clearly superior estimates already from the beginning of the procedure, which we attribute to the low level of noise compared to the other scenarios.

Figure 12 shows that the convergence is faster when both criteria are combined, as opposed to using the PD and EHVD separately (see Figure 7). This is expected, given that one criterion complements the limitations of the other. An advantage of the PD criterion is its very low computational cost, although the EHVD is shown to be more efficient in convergence (also observed in further experiments with other scenarios).

7.3. Impact of stochastic kriging

As evident from Figure 13, feeding SK information to the identification phase of EQUAL and MOCBA (i.e., using SK predictions only when computing the misclassifi-





(a) Average number of points considered at each iteration for scenario WFG3 (high noise)



(b) Performance on WFG3 (high noise)



(c) Average number of points considered at each iteration for scenario WFG4 (medium noise)

(d) Performance on WFG4 (medium noise)



(e) Average number of points considered at each iteration for scenario DTLZ7 (low noise)

Figure 10: Performance of the proposed screening procedures.

cation errors), leads to a drastic improvement in both algorithms (referred to as EQUAL-SKi and MOCBA-SKi respectively); yet, they still show worse performance than the proposed method. The bad performance of MOCBA-SK (i.e., feeding SK information to the



Figure 11: Convergence to the true Pareto front: the proposed algorithm versus standard EQUAL allocation, and the state-of-the-art MOCBA procedure, for the 3 test problems considered.



Figure 12: Individual performance of the proposed resampling criteria on WFG4 (medium noise).

MOCBA algorithm for allocation and identification) is somewhat surprising; however, as the sample variance and prediction MSE don't reflect the same information (i.e., sampling variability and prediction error respectively). Overall, it is clear that the efficient identification of the solutions with the true best expected performance is facilitated by exploiting the stochastic kriging information (as opposed to relying on the sample means).



Figure 13: Performance on WFG4 (medium noise).

7.4. Performance on the supply chain problem

Figure 14 shows that the proposed method remains superior to MOCBA and EQUAL; with increasing noise levels, the superiority becomes more evident. This result is analogous to those observed with analytic test functions. A relevant observation that is not visible from the figure, all algorithms struggle to converge to zero errors on this problem, even with very large budgets of 100 or more iterations, because of several solutions with only marginally different performance (conversely, see the results in Figure 11). Indeed, in practice we will often encounter several solutions whose performance is marginally different, and thus also incorporating a *multiobjective indifference zone* procedure can be beneficial. Yet again, as discussed in Section 2, multiobjecive IZ procedures remain very scarce in the literature. Finally, in Table 5 we report the overhead running time per iteration for all the algorithms. Evidently, fitting the metamodels and quantifying the acquisition functions implies a significant computational effort, especially in this case with 15 input dimensions). In fact, 15 input dimensions is already close to the limit for Gaussian Process regressors Rasmussen & Williams (2005); Kleijnen (2015); higher dimensions would require implementing a dimensionality reduction technique (see e.g., (Binois & Wycoff, 2022), which in turn incorporates yet another source of error.

Table 5: Average (std. dev.) running time per macroreplication in seconds for the practical application (including the simulation evaluations).

$SK-MORS_{none}$	$SK-MORS_{box}$	$SK-MORS_{band}$	MOCBA	EQUAL
57.77 (12.01)	52.44 (9.97)	31.21 (8.40)	7.73 (0.44)	5.77 (0.21)



Figure 14: Performance of the proposed algorithm versus standard EQUAL and MOCBA, for the 4 scenarios considered.

8. Conclusions and future research

This paper proposed a bi-objective ranking and selection technique that can deal with heteroscedastic noise and exploit spatial correlations between the alternatives by using stochastic kriging metamodels. The proposed sampling criteria are able to allocate samples efficiently in view of minimizing both misclassification errors. A high value for the EHVD criterion at a given point indicates that extra replications at that point are expected to generate a drastic change to the estimated front, implying that it is interesting for the decision maker to spend additional replication budget there. This criterion tends to be highest for points that are expected to be on the true front. The PD on the other hand, focuses on improving the accuracy of all the alternatives in objective space, and explicitly neglects the dominance relationships between the alternatives. The PD has the additional advantage of being significantly cheaper to compute. By selecting the points that are on the Pareto front when simultaneously maximizing both criteria, we obtain an algorithm that efficiently allocates the available replication budget in view of reducing misclassification errors.

To reduce the computational effort, we proposed two screening procedures: as shown,

the use of these procedures did not significantly impact the performance of the algorithm in terms of convergence to the true front with low and medium noise levels. The performance of the SCREENING BAND procedure was observed to deteriorate with higher noise levels, as it has a higher risk of filtering out truly non-dominated points (i.e., MCE points). The choice of which one to use in practice will depend on the specific problem at hand. For example, if detecting small differences in performance for the observed non-dominated solutions is not crucial for the decision-maker, using SCREENING BAND can be helpful to reduce the computational overhead significantly, while still considering well-performing solutions. Moreover, both procedures may be combined, first using SCREENING BOX and then switching to SCREENING BAND in later iterations.

Especially in settings with high noise, the sample variance can be inaccurate with respect to the true variance if only a small number of replications is taken. Yet, the results obtained on the real-life supply chain problem illustrate that the proposed algorithm consistently performs well, even if the (endogenous) noise on the objective values is increased by manipulating the coefficient of variation of the demand. We are quite confident that the assumption of linear noise structure in the analytical test problem does not significantly influence our results; yet, other noise structures may be tested. This poses an interesting avenue for further work.

Extending the algorithm to more objectives seems straightforward, as we can build metamodels for as many objectives as is required. However, the computational cost for the hypervolume computations in Equation 11 would increase substantially. Higherdimensional objective spaces are an open question in MORS research, as MORS procedures tend to invest an extremely large replication budget to differentiate solutions that are marginally different (Feldman & Hunter, 2018; Cooper & Hunter, 2020), especially when the number of alternatives are in the hundreds or thousands (as with many objectives). In this context, analyzing the parallelization of the resampling criteria and/or resampling in batches of solutions (e.g., by computing the batch of solutions that jointly maximize the EHVD) would evidently be valuable. We thus believe that a thorough study with 3 or more objectives is non-trivial, and poses a major challenge to the MORS community as well.

References

- Andradóttir, S., & Lee, J. S. (2021). Pareto set estimation with guaranteed probability of correct selection. *European Journal of Operational Research*, 292, 286–298.
- Ankenman, B., Nelson, B. L., & Staum, J. (2010). Stochastic kriging for simulation metamodeling. Operations Research, 58, 371–382.
- Applegate, E. A., Feldman, G., Hunter, S. R., & Pasupathy, R. (2020). Multi-objective ranking and selection: Optimal sampling laws and tractable approximations via score. *Journal of Simulation*, 14, 21–40.

- Auger, A., Bader, J., Brockhoff, D., & Zitzler, E. (2012). Hypervolume-based multiobjective optimization: Theoretical foundations and practical implications. *Theoretical Computer Science*, 425, 75–103.
- Basseur, M., & Zitzler, E. (2006). A preliminary study on handling uncertainty in indicator-based multiobjective optimization. In *Applications of Evolutionary Computing* (pp. 727–739). Springer Berlin Heidelberg.
- Binois, M., Ginsbourger, D., & Roustant, O. (2015). Quantifying uncertainty on pareto fronts with gaussian process conditional simulations. *European Journal of Operational Research*, 243, 386–394.
- Binois, M., & Wycoff, N. (2022). A survey on high-dimensional gaussian process modeling with application to bayesian optimization. ACM Transactions on Evolutionary Learning and Optimization, 2, 1–26.
- Boesel, J., Nelson, B. L., & Kim, S.-H. (2003). Using ranking and selection to "clean up" after simulation optimization. Operations Research, 51, 814–825.
- Branke, J., & Zhang, W. (2019). Identifying efficient solutions via simulation: myopic multi-objective budget allocation for the bi-objective case. OR Spectrum, 41, 831–865.
- Branke, J., Zhang, W., & Tao, Y. (2016). Multiobjective ranking and selection based on hypervolume. In T. Roeder, P. Frazier, R. Szechtman, E. Zhou, T. Huschka, & S. Chick (Eds.), *Proceedings of the 2016 Winter Simulation Conference* (pp. 859–870). Piscataway, New Jersey: IEEE.
- Chen, C.-H., & Lee, L. H. (2010). Stochastic Simulation Optimization volume 1. Singapore: World Scientific. doi:10.1142/7437.
- Chen, X., Ankenman, B. E., & Nelson, B. L. (2012). The effects of common random numbers on stochastic kriging metamodels. ACM Trans. Model. Comput. Simul., 22, 7:1–7:20.
- Chick, S. E., Branke, J., & Schmidt, C. (2010). Sequential sampling to myopically maximize the expected value of information. *INFORMS Journal on Computing*, 22, 71–80. doi:10.1287/ijoc.1090.0327.
- Cooper, K., & Hunter, S. R. (2020). PyMOSO: software for multiobjective simulation optimization with R-PERLE and R-MinRLE. *INFORMS Journal on Computing*, 32, 1101–1108.
- Di Pietro, A., While, L., & Barone, L. (2004). Applying evolutionary algorithms to problems with noisy, time-consuming fitness functions. In *Proceedings of the 2004 Congress* on Evolutionary Computation (pp. 1254–1261). IEEE volume 2.

- Fan, W., Hong, L. J., & Nelson, B. L. (2016). Indifference-zone-free selection of the best. Operations Research, 64, 1499–1514.
- Feldman, G., & Hunter, S. R. (2018). Score allocations for bi-objective ranking and selection. ACM Transactions on Modeling and Computer Simulation, 28, 7:1–7:28.
- Feldman, G., Hunter, S. R., & Pasupathy, R. (2015). Multiobjective simulation optimization on finite sets: Optimal allocation via scalarization. In Y. Yilmaz, W. Chan, I. Moon, T. Roeder, C. Macal, & M. Rosseti (Eds.), *Proceedings of the 2015 Winter Simulation Conference (WSC)* (pp. 3610–3621). Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.
- Feliot, P., Bect, J., & Vazquez, E. (2017). A bayesian approach to constrained single-and multi-objective optimization. Journal of Global Optimization, 67, 97–133.
- Fieldsend, J. E., & Everson, R. M. (2005). Multi-objective optimisation in the presence of uncertainty. In 2005 IEEE Congress on Evolutionary Computation (pp. 243–250). volume 1.
- Fieldsend, J. E., & Everson, R. M. (2015). The rolling tide evolutionary algorithm: A multiobjective optimizer for noisy optimization problems. *IEEE Transactions on Evolutionary Computation*, 19, 103–117.
- Frazier, P. I. (2014). A fully sequential elimination procedure for indifference-zone ranking and selection with tight bounds on probability of correct selection. *Operations Research*, 62, 926–942.
- Hernández-Lobato, D., Hernández-Lobato, J. M., Shah, A., & Adams, R. P. (2016). Predictive entropy search for multi-objective bayesian optimization. In Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48 ICML'16 (pp. 1492–1501). JMLR.
- Horn, D., Dagge, M., Sun, X., & Bischl, B. (2017). First investigations on noisy modelbased multi-objective optimization. In *International Conference on Evolutionary Multi-Criterion Optimization* (pp. 298–313). Springer.
- Huband, S., Hingston, P., Barone, L., & While, L. (2006). A review of multiobjective test problems and a scalable test problem toolkit. *IEEE Transactions on Evolutionary Computation*, 10, 477–506.
- Hughes, E. J. (2001). Evolutionary multi-objective ranking with uncertainty and noise. In International Conference on Evolutionary Multi-Criterion Optimization (pp. 329–343). Springer.
- Hunter, S. R., Applegate, E. A., Arora, V., Chong, B., Cooper, K., Rincón-Guevara, O., & Vivas-Valencia, C. (2019). An introduction to multiobjective simulation optimization. ACM Trans. Model. Comput. Simul., 29, 7:1–7:36. doi:10.1145/3299872.

- Jalali, H., Van Nieuwenhuyse, I., & Picheny, V. (2017). Comparison of kriging-based algorithms for simulation optimization with heterogeneous noise. *European Journal of* Operational Research, 261, 279 – 301.
- Jones, D. R., Schonlau, M., & Welch, W. J. (1998). Efficient global optimization of expensive black-box functions. *Journal of Global optimization*, 13, 455–492.
- Jung, J. Y., Blau, G., Pekny, J. F., Reklaitis, G. V., & Eversdyk, D. (2004). A simulation based optimization approach to supply chain management under demand uncertainty. *Computers & chemical engineering*, 28, 2087–2106.
- Kim, S.-H., & Nelson, B. L. (2001). A fully sequential procedure for indifference-zone selection in simulation. ACM Transactions on Modeling and Computer Simulation (TOMACS), 11, 251–273.
- Kim, S.-H., & Nelson, B. L. (2006). Selecting the best system. In Handbooks in Operations Research and Management Science (pp. 501–534). North Holland: Elsevier volume 13.
- Kleijnen, J. P. C. (2015). *Design and Analysis of Simulation Experiments*. (2nd ed.). NY: Springer.
- Lee, L. H., Chew, E. P., Teng, S., & Goldsman, D. (2010). Finding the non-dominated pareto set for multi-objective simulation models. *IIE Transactions*, 42, 656–674.
- Li, H., Lee, L. H., Chew, E. P., & Lendermann, P. (2015). MO-COMPASS: a fast convergent search algorithm for multi-objective discrete optimization via simulation. *IIE Transactions*, 47, 1153–1169.
- Miettinen, K. (1999). Nonlinear multiobjective optimization volume 12. Springer Science & Business Media.
- Picheny, V., Wagner, T., & Ginsbourger, D. (2013). A benchmark of kriging-based infill criteria for noisy optimization. *Structural and Multidisciplinary Optimization*, 48, 607– 626.
- Rasmussen, C. E., & Williams, C. K. I. (2005). Gaussian Processes for Machine Learning (Adaptive computation and machine learning). (1st ed.). Cambridge, Massachusetts, USA: The MIT Press.
- Rojas-Gonzalez, S., Branke, J., & Nieuwehuyse, I. V. (2019). Multiobjective ranking and selection with correlation and heteroscedastic noise. In 2019 Winter Simulation Conference (WSC) (pp. 3392–3403).
- Rojas-Gonzalez, S., Jalali, H., & Nieuwenhuyse, I. V. (2020). A multiobjective stochastic simulation optimization algorithm. *European Journal of Operational Research*, 284, 212 – 226.

- Staum, J. (2009). Better simulation metamodeling: The why, what, and how of stochastic kriging. In Proceedings of the 2009 Winter Simulation Conference (WSC) (pp. 119–133). IEEE.
- Syberfeldt, A., Ng, A., John, R. I., & Moore, P. (2010). Evolutionary optimisation of noisy multi-objective problems using confidence-based dynamic resampling. *European Journal of Operational Research*, 204, 533–544.
- Teich, J. (2001). Pareto-front exploration with uncertain objectives. In *International Conference on Evolutionary Multi-Criterion Optimization* (pp. 314–328). Springer.
- Teng, S., Lee, L. H., & Chew, E. P. (2010). Integration of indifference-zone with multiobjective computing budget allocation. *European Journal of Operational Research*, 203, 419 – 429.
- Trautmann, H., Mehnen, J., & Naujoks, B. (2009). Pareto-dominance in noisy environments. In Proceedings of the Eleventh Conference on Congress on Evolutionary Computation CEC'09 (pp. 3119–3126). Piscataway, NJ, USA: IEEE Press.
- Voß, T., Trautmann, H., & Igel, C. (2010). New uncertainty handling strategies in multiobjective evolutionary optimization. In *Parallel Problem Solving from Nature*, *PPSN XI* (pp. 260–269). Berlin, Heidelberg: Springer Berlin Heidelberg.
- Zitzler, E., Brockhoff, D., & Thiele, L. (2007). The hypervolume indicator revisited: On the design of pareto-compliant indicators via weighted integration. In *Evolutionary multi-criterion optimization* (pp. 862–876). Springer.

Appendix A MOCBA allocation rules

The allocation rules for the simplified MOCBA algorithm Chen & Lee (2010) are summarized below:

 \bar{f}_{ij} : The averaged observed performance of design *i* for objective *j* after a certain number of replications.

 p_i : The design that dominates design *i* with the highest probability.

 $j_{p_i}^i$: The objective j of p_i that dominates the corresponding objective of design i with the lowest probability.

 τ_{ij} : The observed intrinsic standard deviation of design *i* for objective *j* after a certain number of replications.

- α_i : The budget allocation for design *i*.
- S: The entire set of sampled points.
- S_A : The subset of designs in S labeled as being dominated.

 S_B : The subset of designs in S labeled as being non-dominated.

For any given design $g, h \in S_A$ and $d \in S_B$:

$$\alpha_h = \left(\frac{\tau_{hj_{p_h}^h} / \delta_{hp_h j_{p_h}^h}}{\tau_{gj_{p_g}^g} / \delta_{gp_g j_{p_g}^g}}\right)^2 \tag{14}$$

$$\alpha_d = \sqrt{\sum_{h \in D_d} \frac{\tau_{dj_d^h}^2}{\tau_{hj_d^h}^2} \alpha_h^2} \tag{15}$$

where

$$\delta_{ipj} = \bar{f}_{pj} - \bar{f}_{ij} \tag{16}$$

$$j_{p}^{i} = \underset{j \in \{1,\dots,m\}}{\arg\min} P(\bar{X}_{pj} \le \bar{X}_{ij}) = \underset{j \in \{1,\dots,m\}}{\arg\max} \frac{\delta_{ipj} |\delta_{ipj}|}{\tau_{ij}^{2} + \tau_{pj}^{2}}$$
(17)

$$p_{i} = \underset{\substack{p \in S \\ p \neq i}}{\arg\max} \prod_{j=1}^{m} P(\bar{f}_{pj} \le \bar{f}_{ij}) = \underset{\substack{p \in S \\ p \neq i}}{\arg\min} \frac{\delta_{ipj_{p}^{i}} |\delta_{i}pj_{p}^{i}|}{\tau_{ij_{p}^{i}}^{2} + \tau_{pj_{p}^{i}}^{2}}$$
(18)

$$S_{A} = \left\{ h|h \in S, \frac{\delta_{hp_{h}j_{p_{h}}}^{2}}{\tau_{hj_{p_{h}}}^{2} + \tau_{p_{h}j_{p_{h}}}^{2}} < \min_{i \in D_{h}} \frac{\delta_{ihj_{h}}^{2}}{\tau_{ij_{h}}^{2} + \tau_{hj_{h}}^{2}} \right\}$$
(19)

$$S_B = S \setminus S_A \tag{20}$$

$$D_h = \{i | i \in S, p_i = h\}$$

$$\tag{21}$$

$$D_d = \{h | h \in S_A, p_h = d\}$$

$$\tag{22}$$

Appendix B Formulation of the practical problem

We define the following:

 P_{ijst} : production amount of product *i* on processing unit *j* at facility *s* in time period *t*, this is a decision variable.

 C_{ijs} : unit production cost of product *i* on processing unit *j* at facility *s*.

 S_{isct} : supply of product *i* from facility *s* to sales region *c* in time period *t*, this is a decision variable.

 t_{sc} : unit transportation cost to move a unit of product from facility s to sales region c.

 Γ_{ict} : amount of product *i* to customer *c* in time period *t* that must be outsourced due to insufficient supply at company's facilities.

 ξ_{ic} : unit price of outsourcing for product *i* demanded in sales region *c*.

 I_{ist} : inventory level for product *i* at the end of time period *t* at facility *s*.

 h_{ist} : inventory cost for holding a unit of product *i* in facility *s* for the duration of time period *t*.

 R_{ijst} : effective rate for product *i* on processing unit *j* at facility *s* in time period *t*.

 RL_{ijst} : run length of product *i* on processing unit *j* at facility *s* in time period *t*.

 H_{jst} : amount of time available for production on process j at facility s during time period t.

 ω_{ict} : uncertain demand for product *i* from customer *c* at time period *t*.

 Z_{ict} : binary variable that takes value 1 if demand in sales region c in time period t is met through production at one of the facilities, and 0 if the demand is met through outsourced production.

 M_b : maximum order size for a sales region in a single time period.

 M_s : minimum order size for a sales region in a single time period.

 χ : coefficient of variation of the demand.

 N_c is the number of sales regions.

 N_t is the number of time periods.

The minimization of the cost and maximization of service level is formulated as

$$\min \sum_{i,j,s,t} C_{ijs} P_{ijst} + \sum_{i,s,c,t} t_{sc} S_{isct} + \sum_{i,c,t} \xi_{ic} \Gamma_{ict} + \sum_{i,s,t} h_{ist} I_{ist}$$
(23)
$$\max \frac{1}{N_c N_t} \sum_{c,t} Z_{ict}$$
(24)

subject to:

$$\begin{split} P_{ijst}, S_{isct}, \Gamma_{ict}, I_{ist} &\geq 0 & \text{non-negativity constraint} \\ P_{ijst} &= R_{ijs}(RL_{ijst}) \\ P_{ijst} &\leq R_{ijs}H_{jst} & \text{production capacity constraints} \\ I_{ist} &= I_{is(t-1)} + \sum_{j} P_{ijst} - \sum_{c} S_{isct} & \forall i, s, t & \text{inventory balance constraint} \end{split}$$

$$M_s(1-Z_{ict}) \le \Gamma_{ict} \le M_b(1-Z_{ict})$$

outsource amount constraints

Acknowledgment

This research was supported by the Research Foundation-Flanders, grant number 12AZF24N. We are also thankful to Juan Ungredda from our industrial partner for providing the simulator of the practical application.