Bi-objective Stochastic Simulation Optimization on Integer Lattices via Scalarization

Sebastian Rojas Gonzalez
Ghent University - IMEC
Hasselt University
Belgium
sebastian.rojasgonzalez@imec.be

Ivo Couckuyt
Ghent University - IMEC
Ghent, Belgium
ivo.couckuyt@ugent.be

Joshua Knowles
SLB Cambridge Research
Cambridge, United Kingdom

Abstract

We address the challenging problem of multiobjective optimization via stochastic simulation over a discrete design space. We consider the setting where objective functions are expensive black-box simulations corrupted by heteroscedastic noise, and the decision space is usually too large for exhaustive enumeration. Existing methods often struggle to balance three competing needs: scalable surrogate modeling on discrete domains, principled handling of simulation noise (specifically regarding the uncertainty of the current best solution), and efficient navigation of the multiobjective landscape. Our proposed framework extends the single-objective Complete Expected Improvement acquisition function to the bi-objective case. Our contribution is threefold: (1) we employ Gaussian Markov Random Field surrogates to exploit the integer lattice structure; (2) we use ParEGO-style scalarizations but restrict them to linear to preserve the Gaussianity of the posterior, allowing us to derive a closed-form scalarized acquisition function that explicitly accounts for the covariance between the candidate solution and the noisy incumbent. (3) To maximize this acquisition function, we integrate a discrete Genetic Algorithm with specialized local and jump mutation operators as the inner optimizer. We benchmark our approach against an adaptation of state-of-the-art methods on noisy variants of standard test functions, showing faster early convergence while retaining computational tractability.

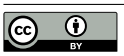
CCS Concepts

• **Computing methodologies** → **Agent / discrete models**; • **Applied computing** → *Multi-criterion optimization and decision-making*; • **Mathematics of computing** → Markov processes.

Keywords

Multiobjective Optimization, Simulation Optimization, Bayesian Optimization, Discrete Optimization, Genetic Algorithms, ParEGO, Complete Expected Improvement

ACM Reference Format:
Sebastian Rojas Gonzalez, Ivo Couckuyt, and Joshua Knowles. 2026. Bi-objective Stochastic Simulation Optimization on Integer Lattices via Scalarization. In *Genetic and Evolutionary Computation Conference (GECCO '26), July 13–17, 2026, San Jose, Costa Rica.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3795095.3805200



GECCO '26, San Jose, Costa Rica

ACM ISBN 979-8-4007-2487-9/2026/07
https://doi.org/10.1145/3795095.3805200

1 Introduction

Optimization via simulation is a critical methodology for solving decision-making problems in many fields, such as engineering, economics, business, and healthcare. It provides a rigorous framework when the relationship between decision variables and objective functions is complex, analytically intractable, and observable only through noisy computer simulations. This paper focuses on a subclass of these problems: *multiobjective optimization via stochastic simulation on integer lattices*.

Such problems arise frequently in real-world systems. Consider the configuration of a manufacturing line where decision variables are the integer counts of machines at each station, and the buffer capacities between them. The objectives—maximizing throughput and minimizing cycle time—are conflicting and noisy due to stochastic processing times and failures. The feasible region $\mathcal{X} \subset \mathbb{Z}^d$ is so large that it makes exhaustive simulation infeasible. In such multiobjective problems, the goal is then to approximate the *Pareto-optimal set* of solutions (i.e., the *non-dominated designs*) rather than identify a single global optimum.

While Bayesian Optimization (BO) [7, 18] is the standard for expensive black-box functions, applying it here is non-trivial. Standard Gaussian Processes (GPs) with continuous kernels (e.g., Matérn) are ill-suited for discrete lattices [7]. Furthermore, standard acquisition functions like Expected Improvement (EI) [23] typically assume a deterministic "current best" value, which is invalid in stochastic simulation where the incumbent is uncertain and subject to re-ranking [22, 35]. A key bottleneck in applying Bayesian optimization to large discrete domains is surrogate scalability. Standard Gaussian processes with common kernels typically induce dense covariance matrices, which become prohibitive on large integer lattices. Noise further complicates acquisition design even with scalable surrogates [38]. In deterministic BO, EI compares a candidate to a fixed incumbent value, whereas under simulation noise the incumbent is random and may be mis-ranked.

Recent work leverages Gaussian Markov Random Fields (GMRFs) [40] as a structure-aligned alternative for lattice-based domains, encoding conditional dependencies via a neighborhood graph. This formulation produces sparse precision matrices, enabling scalable inference through efficient sparse linear algebra. The *Complete Expected Improvement* (CEI) [41] criterion addresses this by taking expectation with respect to the *joint* posterior of the candidate and the uncertain incumbent, explicitly incorporating their covariance. The Gaussian Markov Improvement Algorithm (GMIA) [26] combines GMRF surrogates with the CEI criterion into a complete discrete optimization via simulation framework, computing CEI over the full lattice at each iteration to select the next simulation

point. To address GMIA's computational bottleneck on large lattices, and the *rapid GMIA* (rGMIA) [42] partitions the feasible region into regions to accelerate computations. More recently, Li and Song [28] further extend scalability to high-dimensional lattices with the *projected GMIA* (pGMIA), which hierarchically projects the full-dimensional lattice onto lower-dimensional region and solution layers, enabling effective search even when the lattice size grows exponentially with the number of dimensions.

The aforementioned methods, however, are limited to the single-objective setting. Our work extends this line by integrating the scalability of GMRFs with CEI's principled treatment of noise within a bi-objective framework. A popular strategy for multiobjective BO is scalarization, exemplified by the seminal works of ParEGO [25] and MOEA/D-EGO [44]. ParEGO reduces the vector optimization problem to a sequence of scalar problems using random weights. In this paper, we propose *MO-CEI*, a scalarization framework that bridges GMRF-based discrete optimization and multiobjective evolutionary search. Our specific contributions are:

(1) We restrict the scalarization to random *linear* weighted sums. In contrast to ParEGO-style approaches, which typically rely on Augmented Tchebycheff or other non-linear scalarizations to capture non-convex Pareto fronts [25, 29], we avoid the non-linear max/min operators that break the Gaussian structure of the posterior (as discussed more in detail in Section 4). By preserving Gaussianity, our approach enables a closed-form expression for the Scalarized CEI (sCEI), in which the covariance between the candidate and the incumbent appears explicitly. This term, often neglected in naive extensions, plays a key role in avoiding over-exploration of regions highly correlated with the current best solution [26].
(2) Recognizing that maximizing the acquisition function on a large lattice is a hard optimization problem itself, we explicitly integrate a discrete Genetic Algorithm (GA) as the inner optimizer. Drawing on recent theory of discrete EAs [10], we implement specialized mutation operators: mixing local steps and heavy-tailed jumps to navigate the rough sCEI landscape efficiently.
(3) We adapt the Gaussian Markov Improvement Algorithm (GMIA) [26, 42] to the multiobjective setting using the same random scalarization strategy, providing a fair baseline that isolates the effect of scalarizing the posterior versus scalarizing the observations.

2 Related Work

Our problem sits at the intersection of (i) multiobjective stochastic simulation optimization, (ii) large-scale discrete (lattice) decision spaces, and (iii) evolutionary search for nonconvex, multimodal landscapes. Below we summarize the most relevant streams of the literature and position our contributions with respect to them.

2.1 Bayesian Optimization and Ranking and Selection

Bayesian optimization commonly relies on Gaussian Process (Kriging) surrogates together with acquisition (infill) criteria to guide adaptive sampling of the expensive simulations. Such methods are exceptionally efficient in continuous, low-dimensional spaces [7]. While there has been effort in extending BO to mixed and high-dimensional spaces [12, 33], this literature is mostly the *deterministic* case. For the *stochastic* case, Picheny et al. [35] benchmark a broad family of noisy acquisitions under i.i.d. Gaussian *homoscedastic* noise and show that standard EI can perform poorly because the "best-so-far" value (i.e., the *incumbent* solution) is itself uncertain. Jalali et al. [22] extend this comparison to *heteroscedastic* simulation noise, emphasizing that algorithms must jointly decide *where* to sample (the acquisition rule) and *how much* to sample (replication allocation) under a finite budget; their results highlight that the geometry of the noise variance can materially change relative performance, reinforcing that noise-aware acquisition is essential in simulation optimization. In the multiobjective setting, surveys reach similar conclusions: Hunter et al. [20] note that *multiobjective simulation optimization (MOSO)* methods remain relatively scarce compared to deterministic multiobjective optimization, and Rojas-Gonzalez and Van Nieuwenhuyse [39] observe that Kriging-based MOSO methods that explicitly handle simulation noise are still underdeveloped.

When, after the *search* of solutions, the feasible set is finite and small enough to support extensive replication, Ranking and Selection (R&S) procedures provide statistical guarantees for identifying high-quality designs [3, 16, 24]. Chen and Ryzhov [5] show that CEI-style sampling rules can converge to an optimal budget allocation for single-objective ranking and selection. Multiobjective R&S (MORS) is substantially less mature. Hunter et al. [20] emphasize that theoretical development lags significantly behind the single-objective setting. Some key examples of MORS research include the Multi-objective Optimal Computing Budget Allocation (MOCBA) [27], which allocates replications to reduce misclassification of dominated versus non-dominated designs. SCORE allocation methods [2, 13] provide asymptotically optimal allocation rules by maximizing the decay rate of incorrectly identifying the Pareto set. More recently, model-based approaches have shown potential and robustness to sampling variability [4, 37]: dominance relationships likely change with additional replications, and approximations of the Pareto set can be unstable. Overall, while R&S and MORS are principled when the design set is modest and heavy replication is acceptable, they are typically infeasible when the integer lattices are enormous and the noise is strongly heteroscedastic [37].

Together, these works emphasize open needs central to our setting: computational efficiency at scale, principled treatment of noise, handling of discrete/mixed decision spaces and adaptive sampling that maintains good Pareto coverage.

2.2 Multiobjective Optimization: Bayesian, Evolutionary and Discrete Algorithms

For discrete-lattice optimization, needed for our inner acquisition maximization, we draw on the theoretical literature on discrete evolutionary algorithms [32, 36]. The field has moved beyond establishing basic complexity bounds toward increasingly high-precision analyses of the underlying randomized dynamics. Recent work, in particular, clarifies how the choice of mutation operator governs runtime on rugged fitness landscapes [10]. A key example is

provided by so-called "jump functions": with standard local mutations, crossing wide low-fitness valleys can be exponentially unlikely, whereas heavy-tailed mutation operators (e.g., geometric- or power-law–distributed step sizes) yield provable improvements by enabling occasional long jumps that escape local optima efficiently. These results motivate the mutation operators used in our inner optimization loop (Section 5). At the same time, noise introduces a distinct failure mode: selection based on sample means can induce unstable selection pressure, causing premature elimination of genuinely promising designs. Several notions of dominance under uncertainty have been proposed to mitigate this effect, including probabilistic and rolling formulations, [14, 15, 43] but there is no consensus definition. Moreover, for our setting, principled noise handling also hinges on explicit statistical modeling and appropriate replication strategies [1].

Finally, Multi-objective Bayesian Optimization (MOBO) methods are usually *Pareto-based* approaches [6, 11] that directly optimize set-quality (e.g., hypervolume), or *scalarization-based* approaches [25, 34, 44] that solve a sequence of scalar BO problems to approximate the Pareto front. Under noisy observations, both evaluation and selection become harder because the incumbent Pareto set is itself uncertain, making performance estimation unreliable. Recent Pareto-based work addresses this by integrating hypervolume improvement over posterior uncertainty in the latent outcomes [8], yielding noisy/batch variants that remain effective under noise and are made practical via Monte Carlo estimators and caching strategies. Similarly, ParEGO-like noisy MOBO [38] emphasizes heteroscedastic noise and couples scalarized surrogate search with explicit budget allocation to improve correct Pareto identification, highlighting that common metrics like hypervolume can be misleading when fronts are noisy. Importantly, these lines of work are developed for continuous domains and are not designed to scale to large discrete search spaces, a gap we aim to address with our contributions.

3 Problem Formulation

We consider a minimization problem over a finite integer lattice $\mathcal{X} \subset \mathbb{Z}^d$:

$$\min_{x \in \mathcal{X}} \quad \mathbf{f}(x) = (f_1(x), \ldots, f_m(x))^T . \tag{1}$$

Note that no single design x will minimize all components of $\mathbf{f}(x)$. Thus, we interpret the solution of this multiobjective problem as the set of Pareto-optimal (non-dominated) solutions in $\mathcal{X}$.

We do not observe $\mathbf{f}(x)$ directly. Instead, we observe noisy replications:

$$Y_{j,\ell}(x) = f_j(x) + \varepsilon_{j,\ell}(x), \quad \ell = 1, \ldots, r, \tag{2}$$

where $\varepsilon_{j,\ell}(x) \sim \mathcal{N}(0, \tau_j^2(x))$ represents simulation noise (with variance depending on x, i.e., heteroscedastic noise) at replication ℓ. We assume conditional independence across objectives for tractability.

3.1 GMRF Priors on Lattices

In the following sections we provide only the key equations needed for our development. For the full derivations, additional details, and a more extensive discussion, we refer the reader to [26, 28, 41, 42]. For each objective j, we place a GMRF prior on the latent function values $\mathbf{f}_j = (f_j(x_1), \ldots, f_j(x_N))^T$, assuming a constant prior mean β_j across $\mathcal{X}$:

$$\mathbf{f}_j \sim \mathcal{N}(\beta_j \mathbf{1}, Q_j^{-1}), \tag{3}$$

where Q_j is the precision matrix. We construct Q_j based on the lattice neighborhood structure $\mathcal{N}(x)$. For a point x, neighbors are points x' such that $\|x - x'\|_1 = 1$. The entries of Q_j are non-zero only if points are neighbors, ensuring sparsity. Specifically, we follow the construction by [26], where the precision matrix is parameterized by a vector $\theta = (\theta_0, \theta_1, \ldots, \theta_d)$. The element $Q_{x,x'}$ is defined as:

$$Q_{x,x'} = \begin{cases} \theta_0 + \sum_{k=1}^{d} \theta_k & \text{if } x = x' \\ -\theta_k & \text{if } x = x' \text{are neighbors in dim } k \\ 0 & \text{otherwise .} \end{cases} \tag{4}$$

Here, $\theta_0 > 0$ controls the marginal precision, $\theta_k \geq 0$ controls the spatial dependence in dimension k. This structure ensures Q is diagonally dominant and positive definite. In a d-dimensional lattice, each interior point has $2d$ immediate neighbors, so each row of Q_j has at most $2d + 1$ nonzero entries. Intuitively, this means the conditional expectation of $f_j(x)$ given its neighbors is a weighted average of the neighbors' values, encoding a "smoothness" assumption appropriate for physical systems modeled by simulation.

3.2 Likelihood and Posterior

Given the data $\mathcal{D}_t$ consisting of sampled points and their sample means $\bar{Y}_j(x)$, the likelihood of the sample means is Gaussian [31]. Then the posterior distribution for objective j is also a GMRF with updated parameters [26]:

$$\mathbf{f}_j \mid \mathcal{D}_t \sim \mathcal{N}(\boldsymbol{\mu}_{j,t}, \Sigma_{j,t}) . \tag{5}$$

The posterior mean $\boldsymbol{\mu}_{j,t}$ and precision matrix $\bar{Q}_{j,t}$ can be computed via sparse linear algebra. Specifically, $\Sigma_{j,t} = \bar{Q}_{j,t}^{-1}$ where $\bar{Q}_{j,t} = Q_j + \Lambda_{j,t}$ and $\Lambda_{j,t}$ is a diagonal matrix of observation noise precisions (inverse variance estimates). In summary, $\bar{Q}_{j,t} = Q_j + \Lambda_{j,t}$, and we compute $\boldsymbol{\mu}_{j,t}$ by solving $\bar{Q}_{j,t}\, \boldsymbol{\mu}_{j,t} = Q_j\, \beta_j \mathbf{1} + \Lambda_{j,t}\, \bar{\mathbf{Y}}_j$. Analogous to [28], we never explicitly invert $\bar{Q}_{j,t}$ to get the full dense $\Sigma_{j,t}$. We only compute the specific entries required for the acquisition function (diagonal variances and the covariance terms involving the incumbent) using sparse Cholesky factorization and back-substitution (see [42] for further details).

4 Scalarized CEI

Our method relies on reducing the multiobjective problem to a scalar field that preserves Gaussianity (i.e., linear scalarization). Because the $f_j(x)$ are Gaussian (marginally), any linear combination of them remains Gaussian [30]. While this property enables our closed-form acquisition function, it sacrifices the ability to directly target non-convex Pareto regions that other scalarizations like the Augmented Tchebycheff can reach [29]. However, such alternatives render the derivation of CEI intractable, as the resulting joint distribution between the incumbent and candidate solutions becomes non-Gaussian. While Monte Carlo approximations could be used, they would compromise the analytical tractability and computational efficiency central to our framework. In practice, as illustrated in our experiments, our approach provides good coverage of the (noisy) Pareto front while retaining these key advantages. We therefore leave the important extension to non-linear scalarizations for future work.

At iteration t, we sample weights $\mathbf{w}_t = (w_{t,1}, \ldots, w_{t,m})$ from a uniform distribution over the simplex Δ^{m-1}. We define the scalarized latent function:

$$g_t(x) = \sum_{j=1}^{m} w_{t,j} f_j(x)\,. \tag{6}$$

Then the posterior moments of the scalarized objective $g_t(x)$ are derived directly from the component GMRFs:

$$\mu_{g,t}(x) = \sum_{j=1}^{m} w_{t,j}\mu_{j,t}(x) \tag{7}$$

$$\mathrm{Cov}_t(g_t(x), g_t(x')) = \sum_{j=1}^{m} w_{t,j}^2 \Sigma_{j,t}(x, x')\,, \tag{8}$$

where $\Sigma_{j,t}(x, x')$ is the posterior covariance of objective j between points x and x'. We assume independence between objectives, so cross-covariance terms for $j \neq k$ vanish in Eq. 8.

4.1 Defining the Noisy Incumbent

In deterministic optimization, the incumbent is simply the point with the lowest observed value. In stochastic simulation, we must be more cautious. We define the incumbent $\tilde{x}_t$ as the design in our current dataset $\mathcal{D}_t$ that minimizes the *scalarized sample mean*:

$$\tilde{x}_t = \arg\min_{x \in \mathcal{D}_t} \sum_{j=1}^{m} w_{t,j} \bar{Y}_{j,t}(x)\,. \tag{9}$$

This choice reflects the "current best guess" given the data and the current weight vector. Note that $\tilde{x}_t$ changes as $\mathbf{w}_t$ changes.

4.2 Derivation of Scalarized CEI (sCEI)

We seek a candidate x that maximizes the complete expected improvement over the (uncertain) true value of the incumbent $g_t(\tilde{x}_t)$:

$$\mathrm{sCEI}_t(x) = \mathbb{E}\big[\, \max(g_t(\tilde{x}_t) - g_t(x),\, 0) \mid \mathcal{D}_t \big]\,. \tag{10}$$

Let $D = g_t(\tilde{x}_t) - g_t(x)$. Since g_t is a Gaussian random field, the difference D is normally distributed:

$$D \sim \mathcal{N}(M_D(x),\ V_D(x))\,. \tag{11}$$

The mean of the difference is straightforward:

$$M_D(x) = \mu_{g,t}(\tilde{x}_t) - \mu_{g,t}(x)\,, \tag{12}$$

and the variance of the difference contains the critical interaction term:

$$\begin{aligned} V_D(x) &= \mathrm{Var}\big(g_t(\tilde{x}_t)\big) + \mathrm{Var}\big(g_t(x)\big) - 2\,\mathrm{Cov}\big(g_t(\tilde{x}_t), g_t(x)\big) \\ &= \sigma^2_{g,t}(\tilde{x}_t) + \sigma^2_{g,t}(x) - 2\sum_{j=1}^{m} w_{t,j}^2\, \Sigma_{j,t}(\tilde{x}_t, x)\,. \end{aligned} \tag{13}$$

The term $-2\,\mathrm{Cov}(\cdot)$ penalizes candidates that are highly correlated with the incumbent. On a lattice, points neighboring the incumbent are highly correlated. If the incumbent has high posterior variance (uncertainty), a neighbor will also have high variance. Standard EI might explore the neighbor just because of this uncertainty. CEI, via the covariance term, recognizes that the neighbor's value moves *with* the incumbent's value. If the incumbent turns out to be bad, the neighbor is likely bad too. Thus, CEI greatly reduces the value of purely local resampling unless the mean improvement is significant. In effect, this mechanism prevents the algorithm from wastefully *re-sampling* the incumbent or its immediate neighbors unless there is strong evidence of improvement.

The final closed form is the standard EI formula applied to the difference distribution:

$$\mathrm{sCEI}_t(x) = M_D(x)\,\Phi\Big(\frac{M_D(x)}{\sqrt{V_D(x)}}\Big) + \sqrt{V_D(x)}\,\phi\Big(\frac{M_D(x)}{\sqrt{V_D(x)}}\Big)\,, \tag{14}$$

where ϕ and Φ are the standard normal PDF and CDF.

5 Inner Optimization: Discrete GA

At each iteration t, we must solve the inner optimization:

$$x_t^* = \arg\max_{x \in X} sCEI_t(x)\,. \tag{15}$$

Since X is discrete and potentially very large, gradients are unavailable and exhaustive search is intractable. The *sCEI* landscape on a lattice is often multimodal and "rugged." We employ a discrete Genetic Algorithm (GA) as the inner optimizer, described in Algorithm 1 and denoted GA_{max}.

5.1 Fitness and Representation

5.1.1 Fitness. Evaluating $F(x) = sCEI_t(x)$ requires solving sparse linear systems for the posterior moments. While relatively fast in our implementation, it is not instantaneous, so the GA must be sample-efficient in its use of fitness evaluations.

5.1.2 Representation. Each candidate solution is an integer vector $x \in \mathbb{Z}^d$ subject to box constraints $l_i \leq x_i \leq u_i$ for each coordinate.

5.2 Variation Operators

Standard real-valued operators (like Gaussian mutation) are not ideal for lattices. We implement operators inspired by recent theory of discrete EAs [10].

5.2.1 Initialization with Warm Start. The GMRF posterior changes incrementally as new data points are added. Therefore, the high-*sCEI* regions at iteration t are likely close to those at $t-1$. We initialize the GA population P_t using the final population P_{t-1} from the previous iteration, replacing the worst 20% of individuals with random immigrants to maintain diversity.

5.2.2 Uniform Crossover. For two parents p_1, p_2, the child c is constructed by selecting each gene c_i from either $p_{1,i}$ or $p_{2,i}$ with probability 0.5. This preserves the discrete lattice values present in the population.

5.2.3 Heavy-Tailed Mutation Strategy. To navigate rugged landscapes and cross valleys of low fitness effectively, we implement *Heavy-Tailed Mutation* [9, 10, 17]. Unlike standard bit-flip mutations which often lead to exponential hitting times, we sample the mutation rate from a power-law distribution. The number of coordinates to perturb, k, is drawn from a power-law distribution D_β:

$$k \in \{1, \ldots, d\}, \quad P(k) = C_\beta k^{-\beta}\,, \tag{16}$$

where C_β is a normalization constant and we typically set $\beta = 1.5$. This operator allows the GA to perform a search on the lattice as follows:

- **Small Steps ($k = 1$):** Occur with high probability, favoring local search in the immediate neighborhood.

Algorithm 1 GA_{max}: Discrete GA for sCEI Maximization

Input: Fitness $F(\cdot) = sCEI_t(\cdot)$, Population size N, Elites E
Initialize: $P \leftarrow P_{prev}$ (warm start). Replace worst 20% with random feasible points.
for Generation $g = 1$ to G_{max} **do**
  Evaluate fitness $F(x)$ for all unevaluated $x \in P$.
  Sort P by fitness (descending).
  Store top E individuals in P_{next}.
  while $|P_{next}| < N$ **do**
    $p_1, p_2 \leftarrow$ TournamentSelect(P)
    $c \leftarrow$ UniformCrossover(p_1, p_2)
    Mutation:
    Sample $k \sim D_\beta$ ($\beta = 1.5$).
    Select k distinct dimensions uniformly.
    Perturb selected dimensions by ±1 (boundary check).
    Add c to P_{next}
  end while
  $P \leftarrow P_{next}$
end for
Output: Best solution x^* and updated population P.

- **Long Jumps ($k \approx d$):** Occur with lower, but polynomial (not exponential) probability. This allows the GA to "jump" over valleys of low *sCEI* to discover new regions.

Once k is determined, we select k distinct dimensions uniformly at random and perturb them by ±1 (subject to boundary constraints). We note that the exponent $\beta = 1.5$ follows the recommendation of [10], who show that $\beta \in (1, 2)$ is required for the heavy-tail property: $\beta > 1$ ensures normalizability of D_β, while $\beta < 2$ guarantees that large jumps ($k \approx d$) occur with polynomial rather than exponential probability. The midpoint $\beta = 1.5$ is the canonical choice in the literature and has been shown to yield near-optimal performance across a broad class of combinatorial landscapes without problem-specific tuning [10]. We also note that k is not a fixed parameter but is sampled from D_β at each mutation event.

6 Complete Algorithm and Baseline

The complete MO-CEI algorithm integrates the GMRF model updates, random scalarizations, and GA optimization. Algorithm 2 outlines the full optimization loop. A key feature is the principled handling of the incumbent's uncertainty. In Step 10, if the chosen point x_{next} is the current incumbent $\tilde{x}_t$, the algorithm naturally performs *resampling* of that design. Resampling reduces the variance $\sigma^2(\tilde{x}_t)$, which in turn increases the sCEI of other points in future iterations (by increasing V_D in Eq. 13). Thus, after exploiting the incumbent, the algorithm is dynamically encouraged to explore more in subsequent iterations.

6.1 Baseline: Scalarized GMRF-CEI

To validate MO-CEI, we adapt the *Gaussian Markov Improvement Algorithm (GMIA)* of [26, 42] to the bi-objective setting using a similar random scalarization strategy. Specifically, at each iteration:

1. Sample weights $\mathbf{w}_t$ (same as MO-CEI).
2. Scalarize the observed sample means into a single response $g_t(x) = \sum_j w_{t,j} \bar{Y}_j(x)$.

Algorithm 2 MO-CEI Optimization Loop

Initialize dataset $\mathcal{D}_0$ with LHS of $5 \times d$ designs.
Simulate each initial design r times to obtain $\bar{Y}_j(x)$.
for Iteration $t = 1$ to T (budget of iterations) **do**
  Sample weight vector $\mathbf{w}_t \sim$ Dirichlet($\mathbf{1}$) (uniform simplex).
  for Objective $j = 1$ to m **do**
    Update GMRF posterior parameters $(\boldsymbol{\mu}_{j,t}, \bar{Q}_{j,t})$.
  end for
  Compute scalarized posterior moments via Eq. 7 and 8.
  Identify noisy incumbent $\tilde{x}_t$ via Eq. 9.
  Inner Loop: $x_{next} \leftarrow$ GAmax[sCEI$_t(x)$] (Algorithm 1).
  Simulate x_{next} for r replications (observe $\bar{Y}(x_{next})$).
  Augment dataset: $\mathcal{D}_t \leftarrow \mathcal{D}_{t-1} \cup \{(x_{next}, \bar{Y}(x_{next}))\}$.
end for
Return: Non-dominated solutions from $\mathcal{D}_T$.

3. Fit a single GMRF to the scalarized values $g_t(x)$ and maximize CEI to choose x_{next}.

The key statistical distinction is when scalarization is applied. MO-CEI maintains m independent GMRF posteriors and scalarizes the *posterior moments* (Eqs. 7–8), preserving per-objective spatial structure: each GMRF learns the smoothness of f_j independently, and objectives with different length scales or active subspaces are modeled separately. The baseline, referred to as *GMRF-CEI*, instead scalarizes the *observations* before modeling, so a single GMRF must capture the spatial structure of the scalarized objective directly.

In addition to the surrogate modeling, the two methods differ in how they maximize the CEI acquisition function over the lattice. MO-CEI uses the custom discrete GA with heavy-tailed mutation described in Section 5 (Algorithm 1), which is specifically designed to navigate rugged acquisition landscapes on integer lattices via a mix of local perturbations and long-range jumps. GMRF-CEI instead maximizes CEI using a standard discrete-variable genetic algorithm from Pymoo[1], adapted to handle the integer-constrained search space. This separation is deliberate: it allows us to assess the combined effect of both contributions (per-objective GMRF modeling *and* the specialized inner optimizer).

7 Experiments

We evaluate MO-CEI on two benchmark problems chosen to stress its ability to cope with complex observation noise, multimodality, and high-dimensional discrete search spaces. Standardized benchmarks that combine these characteristics are largely absent from the literature, so our goal here is not exhaustive benchmarking but to provide empirical evidence that MO-CEI is competitive with state-of-the-art baselines under challenging conditions. As future work, we plan to develop and release a broader suite of benchmark problems with these properties (see also the comments in [28]), addressing a current gap in available evaluation testbeds.

We perturb the true objective values with *heteroscedastic* Gaussian noise and thus observe noisy replications

$$Y_{j,\ell}(x) = f_j(x) + \varepsilon_{j,\ell}(x), \qquad \varepsilon_{j,\ell}(x) \sim \mathcal{N}\big(0, \tau_j^2(x)\big),$$

[1] https://pymoo.org/customization/mixed.html

for objectives $j \in \{1, \ldots, m\}$ and replications $\ell = 1, \ldots, r$. Following prior work [22, 38], we set the noise standard deviation $\tau_j(x)$ to vary linearly with the objective value $f_j(x)$. The minimum and maximum values of $\tau_j(x)$ are scaled to the objective's range over the region of interest,

$$RF_j = \max_{x \in P} f_j(x) - \min_{x \in P} f_j(x), \qquad j \in \{1, \ldots, m\},$$

so that noise magnitudes are comparable across objectives. In our experiments, we set the minimum noise level at each objective's individual optimum (i.e., at $\arg\min_{x \in \mathcal{X}} f_j(x)$). Because Pareto-optimal solutions necessarily trade off objective values, this construction induces a corresponding trade-off in *noise* along the data set: points near one extreme of the Pareto front are observed accurately for one objective but noisily for others, and vice versa. Consequently, no design enjoys low-noise observations across all objectives simultaneously. Evidently, other heteroscedastic noise models could achieve a similar effect.

7.1 Test Functions

We construct two bi-objective test problems on the integer lattice $\mathcal{X} = \{0, 1, \ldots, 20\}^d$. Each problem pairs the Zakharov function (objective f_1) with a second objective (f_2), yielding a nontrivial discrete Pareto front. Both single-objective functions are taken from the literature [28]; each integer coordinate x_i is mapped to a continuous domain via an affine transformation before evaluating the function. Figure 1 displays example Pareto fronts for the proposed problems.

7.1.1 Zakharov function (objective f_1 in both problems). The integer vector $\mathbf{x} \in \{0, \ldots, 20\}^d$ is mapped to $\bar{\mathbf{x}} \in [-2, 2]^d$ via $\bar{x}_i = -2 + \frac{4}{20}\, x_i$. The Zakharov function is then

$$f_1(\mathbf{x}) = \sum_{i=1}^{d} \bar{x}_i^2 \;+\; \Big(\sum_{i=1}^{d} 0.5\, i\, \bar{x}_i\Big)^2 \;+\; \Big(\sum_{i=1}^{d} 0.5\, i\, \bar{x}_i\Big)^4.$$

This is a convex function with a unique global minimum at $\bar{\mathbf{x}} = \mathbf{0}$ (integer coordinate $\mathbf{x} = (10, \ldots, 10)$). All d dimensions are active, and the function range grows rapidly with d.

7.1.2 Branin function (objective f_2 in problem ZB). Only the first two coordinates are active. The integer vector $(x_1, x_2) \in \{0, \ldots, 20\}^2$ is mapped to $(\bar{x}_1, \bar{x}_2) \in [0, 1]^2$ via $\bar{x}_i = x_i/20$, followed by the standard rescaling $\hat{x}_1 = 10\bar{x}_1 - 5$ and $\hat{x}_2 = 10\bar{x}_2$. The Branin function is

$$f_2^{\mathrm{ZB}}(\mathbf{x}) = \Big(\hat{x}_2 - \frac{5.1}{4\pi^2}\hat{x}_1^2 + \frac{5}{\pi}\hat{x}_1 - 6\Big)^2 + 10\Big(1 - \frac{1}{8\pi}\Big)\cos(\hat{x}_1) + 10.$$

This is a multimodal, non-convex function. On the lattice, the global minimum is at integer coordinate $(x_1, x_2) = (16, 5)$. Because only two of d dimensions affect f_2, this problem tests whether the algorithm can exploit the low effective dimensionality of one objective while simultaneously optimizing the fully d-dimensional f_1.

7.1.3 Styblinski–Tang function (objective f_2 in problem ZS). The integer vector $\mathbf{x} \in \{0, \ldots, 20\}^d$ is mapped to $\bar{\mathbf{x}} \in [-6, 6]^d$ via $\bar{x}_i = -6 + \frac{12}{20}\, x_i$. The scaled Styblinski–Tang function is

$$f_2^{\mathrm{ZS}}(\mathbf{x}) = \frac{1}{2d} \sum_{i=1}^{d} \big(\bar{x}_i^4 - 16\bar{x}_i^2 + 5\bar{x}_i\big).$$

This is a highly multimodal function with 3^d local minima. On the lattice, the global minimum is at $\mathbf{x} = (5, \ldots, 5)$, mapping to $\bar{x}_i = -3.0$. Unlike the Branin function, all d dimensions are active, making this a harder problem in which neither objective has a low-dimensional structure the algorithm can exploit.

7.2 Parameter Settings

Each candidate design is evaluated with $r = 5$ independent replications, and each algorithm is allocated $T = 200$ iterations after an initial Latin Hypercube Sample (LHS) of $5d$ points. For the MO-CEI inner optimizer, the GA uses a population of $N = 100$ individuals evolved for $G_{\max} = 50$ generations per iteration, with binary tournament selection, uniform crossover at rate 0.8, and the heavy-tailed mutation operator (Eq. 16) with $\beta = 1.5$. Elitism preserves the top $E = 5$ individuals, and the population is warm-started from the previous iteration with 20% of individuals replaced by random feasible points to maintain diversity. The baseline maximizes CEI using `pymoo`'s discrete-variable GA with the same population size and generation budget. The GMRF hyperparameters $\boldsymbol{\theta} = (\theta_0, \theta_1, \ldots, \theta_d)$ are estimated via maximum likelihood every 10 iterations (for MO-CEI).

7.3 Metric

We report the normalized Hypervolume Regret $R_t \in [0, 1]$, defined as

$$R_t = \frac{\mathrm{HV}(P_{\mathrm{true}}) - \mathrm{HV}(P_t)}{\mathrm{HV}(P_{\mathrm{true}}) - \mathrm{HV}(P_0)}, \tag{17}$$

where P_{true} is the true (deterministic) Pareto front, P_0 is the non-dominated set obtained from the initial LHS before any algorithm iteration, and P_t is the non-dominated set of solutions found by the algorithm after t iterations. By construction, $R_0 = 1$, and $R_t \to 0$ as the algorithm's solution set approaches the true Pareto front. The denominator normalizes by the gap available after initialization, so R_t measures the fraction of the initial HV gap that remains at iteration t. Because the decision space is the finite integer lattice $\mathcal{X} = \{0, \ldots, 20\}^d$, the true Pareto front can be obtained by exhaustive enumeration for moderate d. We evaluate the *true* (noiseless) objective values $f_1(\mathbf{x}), f_2(\mathbf{x})$ at every feasible point and extract the non-dominated subset. Within each macro-replication, all competing methods share the same $\mathcal{D}_0$ (and thus the same P_0), so the normalization denominator in Eq. 17 is identical across methods.

As all problems are bi-objective, the hypervolume is computed exactly [19]. We normalize each objective to the interval $[0, 1]$ using the Pareto front's *ideal* and *nadir* points:

$$\tilde{f}_j(\mathbf{x}) = \frac{f_j(\mathbf{x}) - f_j^{\mathrm{ideal}}}{f_j^{\mathrm{nadir}} - f_j^{\mathrm{ideal}}}, \qquad j \in \{1, 2\}, \tag{18}$$

where $f_j^{\mathrm{ideal}} = \min_{\mathbf{x} \in P_{\mathrm{true}}} f_j(\mathbf{x})$ and $f_j^{\mathrm{nadir}} = \max_{\mathbf{x} \in P_{\mathrm{true}}} f_j(\mathbf{x})$. The reference point is set to $\mathbf{r} = (1 + \delta,\ 1 + \delta)$ with $\delta = 0.1$ [21]. This normalization ensures that the metric is scale-invariant across objectives and problems. We run 30 macro-replications with independent random seeds and report mean R_t with standard error bands.

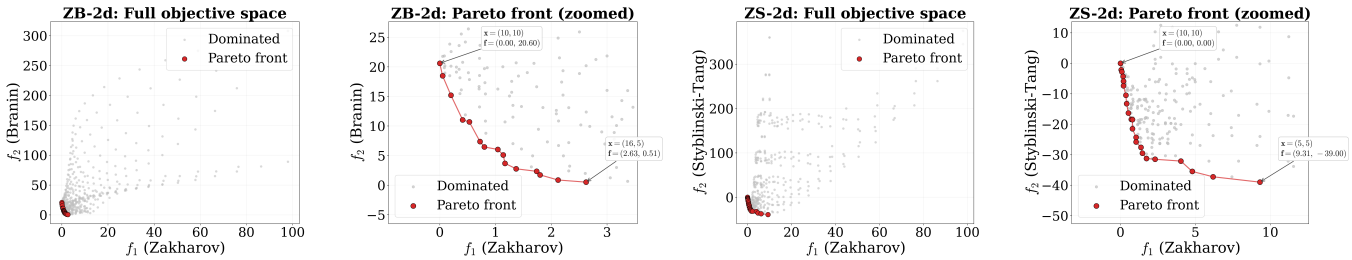


Figure 1: Objective space and Pareto fronts of the two bi-objective test problems on the $\{0, \ldots, 20\}^2$ integer lattice. Left pair: ZB with 15 Pareto-optimal points. Right pair: ZS with 22 Pareto-optimal points.

8 Results and Analysis

Figure 2 shows the convergence of R_t over $T = 200$ iterations for both test problems at $d = 2, 5, 10$. We analyze these results through the two key differences between MO-CEI and the baselines.

The advantage of scalarizing the posterior rather than the observations is most visible on ZB, where Branin depends on only 2 of d dimensions while Zakharov depends on all d. MO-CEI's per-objective GMRFs learn these distinct length scales independently, whereas the baseline's single GMRF must represent both structures with one set of hyperparameters. At $d = 2$ both methods perform similarly because the objectives share the same two active dimensions, but at $d = 5$ and $d = 10$ the gap widens as the active-subspace mismatch grows. On ZS, where both objectives depend on all d dimensions, the modeling advantage is smaller but still present: the smooth convex landscape of Zakharov and the highly multimodal landscape of Styblinski-Tang benefit from separate precision parameters. Additionally, since the baseline refits its single GMRF to a different scalar field at every iteration, hyperparameters from iteration $t-1$ likely transfer poorly. MO-CEI's per-objective GMRFs are weight-independent, avoiding this issue.

The warm-started GA exploits the incremental nature of posterior updates (high sCEI regions shift gradually between iterations), while the heavy-tailed operator enables occasional long-range jumps to escape local optima (recall that these problems have a large number of local optima). At $d = 10$, the baseline's standard `pymoo` GA frequently gets trapped in suboptimal acquisition peaks, whereas MO-CEI's tailored operator continues to discover improving new areas. The two contributions are complementary: better per-objective modeling produces a more informative sCEI surface with sharper peaks near the Pareto-optimal region, and the specialized GA is more effective at finding them.

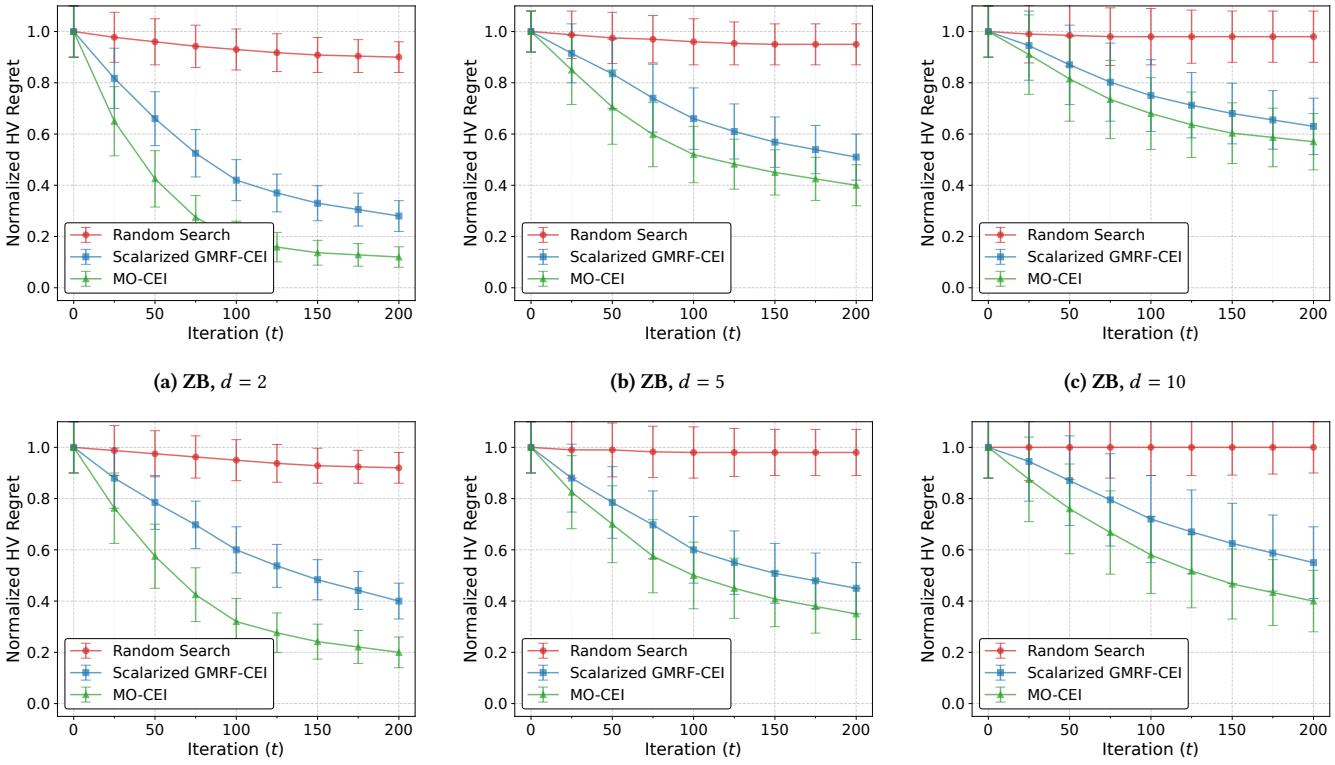
Finally, MO-CEI incurs a higher per-evaluation cost in the inner loop, since each sCEI query requires posterior moments from m independent GMRFs rather than one. The baseline, on the other hand, must refit its single GMRF (including hyperparameter re-estimation) at *every* iteration. However, in typical simulation optimization settings where each function evaluation may take minutes or hours [38], this overhead is negligible as a single iteration of either method completes in seconds on the problem sizes tested here.

9 Conclusion and Future Work

We have presented a framework designed to bridge the gap between rigorous statistical modeling and heuristic search for bi-objective stochastic simulation optimization on integer lattices. By utilizing GMRF surrogates, we leveraged sparse precision matrices to model large discrete spaces more efficiently than standard Gaussian Processes. By restricting our approach to random linear scalarizations, we preserved the Gaussian nature of the posterior, enabling the derivation of a closed-form Scalarized Complete Expected Improvement (sCEI). This criterion explicitly accounts for the covariance between a candidate solution and the noisy incumbent, preventing the over-exploration of correlated regions. However, linear scalarizations will likely struggle to uncover non-convex regions of Pareto fronts [29]. Naturally, future work will extend MO-CEI to support augmented Tchebycheff scalarizations or Expected Hypervolume Improvement. As these approaches sacrifice the Gaussian property of the posterior, we aim to investigate efficient Monte Carlo estimators to approximate the acquisition values without incurring prohibitive computational costs.

To maximize this acquisition function over large lattices, we integrated a discrete Genetic Algorithm as the inner optimizer. Our results demonstrate that incorporating heavy-tailed mutation operators allows the algorithm to escape local optima in the acquisition landscape effectively. The interplay between the GMRF model updates and the evolutionary search dynamics is certainly an important avenue for future investigation. We intend to explore adaptive mutation rates and population sizing to further reduce the computational overhead of the inner optimization loop.

A natural question is how good MO-CEI (and GMRF-CEI) scale beyond the dimensions tested here. In the lattice size grows as 21^d, storing a full precision matrix $\bar{Q}_{j,t}$ per objective becomes the dominant bottleneck well before the inner GA does: at $d = 10$ the lattice already contains more than 10^{13} points, and although $\bar{Q}_{j,t}$ is sparse with at most $2d + 1$ non-zeros per row, the number of rows itself becomes prohibitive. Two complementary directions are worth pursuing. First, the projected GMRF construction of [28] could be adapted to our per-objective setting, hierarchically decomposing each $\bar{Q}_{j,t}$ onto region- and solution-layer graphs and substantially reducing computing effort. Second, because the GA only queries sCEI at a small subset of lattice points per iteration, the posterior moments required by the acquisition function could in principle be computed by solving sparse linear systems *only* at visited candidates rather than obtaining the full posterior.



(a) ZB, $d = 2$ **(b) ZB, $d = 5$** **(c) ZB, $d = 10$**

(d) ZS, $d = 2$ **(e) ZS, $d = 5$** **(f) ZS, $d = 10$**

Figure 2: Convergence of normalized HV regret R_t on both test problems for dimensions $d = 2, 5, 10$. Top row: Zakharov + Branin (ZB). Bottom row: Zakharov + Styblinski-Tang (ZS). Mean ± SE over 30 macro-replications.

Acknowledgments

Sebastian Rojas Gonzalez acknowledges support by FWO (grant number 1216021N). Ivo Couckuyt acknowledges support by the Belgian (Flemish) Government under the Onderzoeksprogramma Artificiële Intelligentie Vlaanderen. Joshua Knowles acknowledges SLB's support and latitude in granting time to work on this manuscript. All the authors acknowledge the use of generative AI tools to improve the clarity and readability of the manuscript and to accelerate portions of the software implementation (e.g., prototyping and code refactoring). The authors take full responsibility for the correctness of the methods, results, references, and conclusions.

References

[1] Bruce Ankenman, Barry L Nelson, and Jeremy Staum. 2010. Stochastic Kriging for Simulation Metamodeling. *Operations Research* 58, 2 (2010), 371–382.

[2] Eric A. Applegate, Guy Feldman, Susan R. Hunter, and Raghu Pasupathy. 2020. Multi-objective ranking and selection: Optimal sampling laws and tractable approximations via SCORE. *Journal of Simulation* 14, 1 (2020), 21–40.

[3] Justin Boesel, Barry L Nelson, and Seong-Hee Kim. 2003. Using ranking and selection to "Clean Up" after simulation optimization. *Operations Research* 51, 5 (2003), 814–825.

[4] Juergen Branke and Wen Zhang. 2019. Identifying efficient solutions via simulation: myopic multi-objective budget allocation for the bi-objective case. *OR Spectrum* 41, 3 (2019), 831–865.

[5] Ye Chen and Ilya O Ryzhov. 2019. Complete expected improvement converges to an optimal budget allocation. *Advances in Applied Probability* 51, 1 (2019), 209–235.

[6] Ivo Couckuyt, Dirk Deschrijver, and Tom Dhaene. 2014. Fast calculation of multiobjective probability of improvement and expected improvement criteria for Pareto optimization. *Journal of Global Optimization* 60, 3 (2014), 575–594.

[7] Ivo Couckuyt, Sebastian Rojas Gonzalez, and Juergen Branke. 2022. Bayesian optimization: tutorial. In *Proceedings of the genetic and evolutionary computation conference companion*. 843–863.

[8] Samuel Daulton, Maximilian Balandat, and Eytan Bakshy. 2021. Parallel bayesian optimization of multiple noisy objectives with expected hypervolume improvement. *Advances in neural information processing systems* 34 (2021), 2187–2200.

[9] Benjamin Doerr, Huu Phuoc Le, Régis Makhmara, and Ta Duy Nguyen. 2017. Fast genetic algorithms. In *Proceedings of the genetic and evolutionary computation conference*. 777–784.

[10] Benjamin Doerr and Frank Neumann. 2021. A survey on recent progress in the theory of evolutionary algorithms for discrete optimization. *ACM Transactions on Evolutionary Learning and Optimization* 1, 4 (2021), 1–43.

[11] Michael TM Emmerich, Kyriakos C Giannakoglou, and Boris Naujoks. 2006. Single-and multiobjective evolutionary optimization assisted by Gaussian random field metamodels. *IEEE Transactions on Evolutionary Computation* 10, 4 (2006), 421–439.

[12] David Eriksson, Michael Pearce, Jacob Gardner, Ryan D Turner, and Matthias Poloczek. 2019. Scalable Global Optimization via Local Bayesian Optimization. In *Advances in Neural Information Processing Systems*. 5496–5507. http://papers.nips.cc/paper/8788-scalable-global-optimization-via-local-bayesian-optimization.pdf

[13] Guy Feldman and Susan R. Hunter. 2018. SCORE Allocations for Bi-objective Ranking and Selection. *ACM Transactions on Modeling and Computer Simulation* 28, 1, Article 7 (Jan. 2018), 28 pages.

[14] J. E. Fieldsend and R. M. Everson. 2005. Multi-objective optimisation in the presence of uncertainty. In *2005 IEEE Congress on Evolutionary Computation*, Vol. 1. 243–250.

[15] Jonathan E Fieldsend and Richard M Everson. 2015. The rolling tide evolutionary algorithm: A multiobjective optimizer for noisy optimization problems. *IEEE Transactions on Evolutionary Computation* 19, 1 (2015), 103–117.

[16] Peter I Frazier. 2014. A fully sequential elimination procedure for indifference-zone ranking and selection with tight bounds on probability of correct selection. *Operations Research* 62, 4 (2014), 926–942.
[17] Tobias Friedrich, Francesco Quinzan, and Markus Wagner. 2018. Escaping large deceptive basins of attraction with heavy-tailed mutation operators. In *Proceedings of the Genetic and Evolutionary Computation Conference*. 293–300.
[18] Roman Garnett. 2023. *Bayesian optimization*. Cambridge University Press.
[19] Andreia P Guerreiro, Carlos M Fonseca, and Luís Paquete. 2021. The hypervolume indicator: Computational problems and algorithms. *ACM Computing Surveys (CSUR)* 54, 6 (2021), 1–42.
[20] Susan R. Hunter, Eric A. Applegate, Viplove Arora, Bryan Chong, Kyle Cooper, Oscar Rincón-Guevara, and Carolina Vivas-Valencia. 2019. An Introduction to Multiobjective Simulation Optimization. *ACM Trans. Model. Comput. Simul.* 29, 1, Article 7 (Jan. 2019), 36 pages. doi:10.1145/3299872
[21] Hisao Ishibuchi, Ryo Imada, Yu Setoguchi, and Yusuke Nojima. 2018. How to specify a reference point in hypervolume calculation for fair performance comparison. *Evolutionary computation* 26, 3 (2018), 411–440.
[22] Hamed Jalali, Inneke Van Nieuwenhuyse, and Victor Picheny. 2017. Comparison of Kriging-based Algorithms for Simulation Optimization with Heterogeneous Noise. *European Journal of Operational Research* 261, 1 (2017), 279 – 301.
[23] Donald R Jones, Matthias Schonlau, and William J Welch. 1998. Efficient global optimization of expensive black-box functions. *Journal of Global optimization* 13 (1998), 455–492.
[24] S.-H. Kim and Barry L. Nelson. 2006. Selecting the best system. In *Handbooks in Operations Research and Management Science*, Vol. 13. Elsevier, North Holland, 501–534.
[25] Joshua Knowles. 2006. ParEGO: A hybrid algorithm with on-line landscape approximation for expensive multiobjective optimization problems. *IEEE Transactions on Evolutionary Computation* 10, 1 (2006), 50–66.
[26] Peter L. Salemi, Eunhye Song, Barry L Nelson, and Jeremy Staum. 2019. Gaussian Markov random fields for discrete optimization via simulation: Framework and algorithms. *Operations Research* 67, 1 (2019), 250–266.
[27] Loo Hay Lee, Ek Peng Chew, Suyan Teng, and David Goldsman. 2010. Finding the non-dominated Pareto set for multi-objective simulation models. *IIE Transactions* 42, 9 (2010), 656–674.
[28] Xinru Li and Eunhye Song. 2024. Projected Gaussian Markov improvement algorithm for high-dimensional discrete optimization via simulation. *ACM Transactions on Modeling and Computer Simulation* 34, 3 (2024), 1–29.
[29] Kaisa Miettinen. 1999. *Nonlinear multiobjective optimization*. Vol. 12. Springer Science & Business Media.
[30] Saralees Nadarajah and Samuel Kotz. 2008. Exact distribution of the max/min of two Gaussian random variables. *IEEE Transactions on Very Large Scale Integration Systems* 16, 2 (2008), 210–212.
[31] Barry L Nelson. 2010. Optimization via simulation over discrete decision variables. In *Risk and optimization in an uncertain world*. Informs, 193–207.
[32] Frank Neumann and Carsten Witt. 2010. *Bioinspired Computation in Combinatorial Optimization: Algorithms and Their Computational Complexity*. Springer.
[33] Leonard Papenmeier, Luigi Nardi, and Matthias Poloczek. 2023. Bounce: Reliable high-dimensional Bayesian optimization for combinatorial and mixed spaces. *Advances in Neural Information Processing Systems* 36 (2023), 1764–1793.
[34] Biswajit Paria, Kirthevasan Kandasamy, and Barnabás Póczos. 2020. A flexible framework for multi-objective Bayesian optimization using random scalarizations. In *Uncertainty in Artificial Intelligence*. PMLR, 766–776.
[35] Victor Picheny, Tobias Wagner, and David Ginsbourger. 2013. A benchmark of kriging-based infill criteria for noisy optimization. *Structural and Multidisciplinary Optimization* 48, 3 (2013), 607–626.
[36] Camelia-Mihaela Pintea. 2014. *Advances in bio-inspired computing for combinatorial optimization problems*. Springer.
[37] Sebastian Rojas Gonzalez, Juergen Branke, and Inneke Van Nieuwenhuyse. 2025. Bi-objective ranking and selection using stochastic kriging. *European Journal of Operational Research* 322, 2 (2025), 599–614.
[38] Sebastian Rojas Gonzalez, Hamed Jalali, and Inneke Van Nieuwenhuyse. 2020. A multiobjective stochastic simulation optimization algorithm. *European Journal of Operational Research* 284, 1 (2020), 212–226.
[39] Sebastian Rojas-Gonzalez and Inneke Van Nieuwenhuyse. 2020. A survey on kriging-based infill algorithms for multiobjective simulation optimization. *Computers & Operations Research* 116 (2020), 104869.
[40] Havard Rue and Leonhard Held. 2005. *Gaussian Markov random fields: theory and applications*. Chapman and Hall/CRC.
[41] Peter Salemi, Barry L Nelson, and Jeremy Staum. 2014. Discrete optimization via simulation using Gaussian Markov random fields. In *Proceedings of the Winter Simulation Conference 2014*. IEEE, 3809–3820.
[42] Mark Semelhago, Barry L Nelson, Eunhye Song, and Andreas Wächter. 2021. Rapid discrete optimization via simulation with Gaussian Markov random fields. *INFORMS Journal on Computing* 33, 3 (2021), 915–930.
[43] Heike Trautmann, Jörn Mehnen, and Boris Naujoks. 2009. Pareto-dominance in Noisy Environments. In *Proceedings of the Eleventh Conference on Congress on Evolutionary Computation* (Trondheim, Norway) *(CEC'09)*. IEEE Press, Piscataway, NJ, USA, 3119–3126.
[44] Qingfu Zhang, Wudong Liu, Edward Tsang, and Botond Virginas. 2009. Expensive multiobjective optimization by MOEA/D with Gaussian process model. *IEEE Transactions on Evolutionary Computation* 14, 3 (2009), 456–474.