MOBO-CAPS: Multi-Objective Bayesian Optimization with Cardinality-Aware Pareto Selection

Hanyang Wang
University of Warwick
Coventry, CV4 7AL, United Kingdom
Hanyang.Wang@warwick.ac.uk

Juergen Branke
University of Warwick
Coventry, CV4 7AL, United Kingdom
Juergen.Branke@wbs.ac.uk

Matthias Poloczek*
Databricks
San Francisco, CA 94105, USA
matthias.poloczek@databricks.com

Abstract

Multi-objective Bayesian optimization (MOBO) traditionally aims to approximate the entire Pareto front, but, due to resource constraints, real-world deployment is often limited to testing a small, representative subset of solutions. Hence current methods first approximate the full front before selecting solutions, potentially wasting costly evaluations. We formalize this problem as Multi-objective Bayesian Optimization with Cardinality-aware Pareto Selection (MOBO-CAPS), where the goal is to identify a set of exactly M solutions with high hypervolume. We propose two practical cardinality-aware acquisition strategies: REHVI, a replacement-aware approximation to expected hypervolume improvement, and CHO-UCB, which constructs an optimistic M-point set by optimizing a collective UCB-based hypervolume objective. We prove a sublinear simple regret bound for the batched version of CHO-UCB, and empirically show that cardinality-aware acquisition design improves over standard MOBO followed by post-hoc subset selection across nine benchmarks.

1 Introduction

Bayesian Optimization (BO) has become a standard method for the optimization of expensive-to-evaluate black-box functions, a task that is ubiquitous in science and engineering. Multi-objective Bayesian Optimization (MOBO) aims to identify the Pareto front, i.e., a set of non-dominated solutions representing different trade-offs between conflicting objectives. A common approach in MOBO is to sequentially select evaluation points that improve coverage of the Pareto front, with hypervolume being a widely used performance metric due to its Pareto compliance and ability to capture both convergence and diversity. This approach has achieved considerable success in practical applications, such as protein design [Stanton et al., 2022], materials engineering [Biswas et al., 2021] and model predictive control [Makrygiorgos et al., 2022].

However, in real-world applications, decision-makers often require only a small, carefully selected subset of Pareto-optimal solutions rather than the entire front. The motivations for considering only a small, finite subset of solutions are twofold. First, the goal may be to find a small but representative set that covers the entire Pareto front. This is common when catering to a heterogeneous user base with diverse preferences, such as providing a range of investment options with different risk profiles or car models that appeal to different market segments. The aim is to present a wide spectrum of possible

*This research does not relate to Matthias' work at Amazon and Databricks.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

trade-offs. Second, MOBO is often only the first step in a longer decision-making process. Engineers and scientists frequently require a manageable candidate set for detailed, hands-on evaluation. This subsequent scrutiny involves domain-specific criteria that are difficult or impossible to formalize as objectives, such as model interpretability, aesthetic appeal of a design, or subtle safety considerations. This detailed review might itself focus on a diverse set of candidates from across the Pareto front, or it may concentrate on a specific sub-region of the Pareto front deemed most promising. In practice, this leads to a two-step procedure: run MOBO to approximate the entire Pareto front, then apply a post-hoc selection procedure to extract a subset of M solutions. One way to select a subset of solutions is to pick the M solutions that jointly maximize the hypervolume, i.e., the space dominated by the solutions, bounded by some given reference point. This problem itself is NP-hard [Chen et al., 2021], and various heuristics have been proposed to solve it, see, e.g., Bringmann and Friedrich [2010], Kuhn et al. [2016]. Despite its convenience, this strategy remains informal, and the cardinality constraint is not incorporated into the optimization process itself, making it potentially inefficient.

In this paper, we formalize this overlooked, yet fundamental setting as a sequential black-box optimization problem: identifying exactly M solutions with maximal hypervolume under a limited evaluation budget. We refer to this problem as *Multi-objective Bayesian Optimization with Cardinality-Aware Pareto Selection* (MOBO-CAPS). This formulation bridges the gap between theoretically founded MOBO algorithms and the practical need to select a small number of actionable solutions. It also opens up new methodological questions, such as how to design acquisition functions that effectively optimize the hypervolume under a fixed-cardinality constraint. Hypervolume serves as our objective for selecting M solutions because it guarantees that the chosen solutions lie on the Pareto front, while promoting coverage of diverse trade-offs among objectives. By targeting the M-point set that maximizes the hypervolume, we preserve alignment with the fundamental principles of MOBO.

To tackle this problem, we propose two algorithms. The first, *Replacement-aware Expected Hypervolume Improvement* (REHVI), extends the classic Expected Hypervolume Improvement (EHVI) acquisition function by accounting for the need to maintain a solution set of size M. The second, *Collective Higher-dimensional Optimization with Upper Confidence Bound* (CHO-UCB), reformulates the problem as a joint optimization over M solutions using a high-dimensional acquisition function. We analyze the computational efficiency and optimization behavior of these methods, and conduct extensive experiments across both synthetic benchmarks and real-world applications. Our results show that incorporating the cardinality constraint throughout the optimization process leads to better performance than traditional MOBO followed by post-hoc selection.

Our contributions are fourfold. First, we introduce MOBO-CAPS, a formal MOBO setting in which the final recommendation is constrained to contain exactly M solutions. Second, we propose two new algorithms, REHVI and CHO-UCB, that directly incorporate cardinality information into the optimization process. Third, we establish a regret bound for batched CHO-UCB, providing a first regret analysis for this setting. Finally, we validate our approach through extensive experiments on synthetic and real-world benchmarks, demonstrating that our methods outperform standard MOBO with post-hoc selection.

The structure of this paper is as follows. Section 2 reviews the literature on MOBO and related problems. Section 3 introduces the MOBO-CAPS problem setting. Section 4 presents the REHVI and CHO-UCB methods. Section 5 reports experimental results. Section 6 concludes with a summary and discusses avenues for future work. The code is available at https://github.com/HanyangHenry-Wang/MOBO-CAPS/tree/master.

2 Literature Review

In multi-objective optimization, various metrics have been proposed to measure the quality of a Pareto front approximation. The most prominent, and the one that is strictly Pareto compliant, is hypervolume [Zitzler and Thiele, 1998]. In MOBO, the EHVI acquisition function directly optimizes hypervolume [Emmerich et al., 2007], though computational costs initially limited adoption. Yang et al. [2019a] improved computational efficiency by decomposing integration domains into axis-aligned hyperrectangles, while Yang et al. [2019b] introduced gradient-based optimization for bi-objective problems. Daulton et al. [2020] leveraged automatic differentiation and parallel hardware to enhance scalability significantly. Alternative hypervolume-based approaches include UCB-based hypervolume improvement estimation called $\mathcal{S}$-Metric-Selection-based Efficient Global Optimization

(SMS-EGO) [Ponweiser et al., 2008] and probability of hypervolume improvement [Couckuyt et al., 2014]. Beyond methods based on hypervolume, ParEGO [Knowles, 2006] converts multi-objective problems into a sequence of single-objective problems by applying a different weighted Tchebycheff scalarization at each iteration. A related approach, proposed by Paria et al. [2020], maintains separate surrogate models for each objective and applies the scalarization directly within the acquisition function.

Four related multi-objective optimization problem settings focus on selecting single points or subsets from Pareto fronts. Bayesian Optimization with Preference Learning [Astudillo and Frazier, 2020, Lin et al., 2022, Ungredda et al., 2022, Ozaki et al., 2024, Wang et al., 2025] selects solutions maximizing decision maker utility based on learned preferences. Finding Knees in Bayesian Multi-objective Optimization [Heidari et al., 2022, 2024], aims to identify a single Pareto optimal point maximizing hypervolume. However, searching for a single solution with maximal hypervolume is much easier, as it avoids the combinatorial problem of identifying a subset. The third similar problem setting is Multi-Objective Coverage Bayesian Optimization [Maus et al., 2025]. It selects solution subsets in many-objective problems, maximizing the so-called coverage score, defined as the sum over all objective functions of the maximum observed across all solutions in the set of given cardinality (with the cardinality being less than the number of objectives). Finally, while not a BO method, Wang et al. [2017] addressed a closely related problem for functions with accessible gradient. Their method performs gradient ascent directly in the decision space to discover a set of Pareto-optimal solutions. A direct comparison of these settings is provided in Appendix A.2.

For a given Pareto front, researchers have looked at the problem of identifying a small number of solutions with maximal hypervolume. The μ-distribution problem identifies μ points from a known Pareto front to maximize hypervolume. For example, Glasmachers [2014] developed a gradient ascent approach that operates directly in the objective space to compute hypervolume-optimal μ-distributions for two- and three-objective benchmark fronts. Theoretical studies have also examined optimal μ-distributions, e.g., Emmerich et al. [2007], Auger et al. [2009] showed that hypervolume-optimal points are uniformly distributed on linear Pareto fronts in two-objective problems. The Hypervolume Subset Selection Problem (HSSP) selects a fixed number of solutions from a given pool to maximize hypervolume. Ishibuchi et al. [2009] proposed genetic algorithms encoding solutions as binary strings with fitness combining hypervolume and a penalty term. Later work [Ishibuchi et al., 2014] reformulated HSSP as multi-objective optimization with different reference points. Several evolutionary algorithms have been proposed for μ-distribution and HSSP; see Shang et al. [2020] for comprehensive coverage. Unlike these settings, MOBO-CAPS is a sequential black-box problem in which the optimal set for a given size M must be composed rather than selected from a pool of already observed sets.

The task of finding a set of solutions in single-objective BO has also been studied. Maus et al. [2023] select diverse solutions according to given criteria by maintaining several trust regions. Mei et al. [2023] identify local optima through BO by incorporating the surrogate model's first-order derivatives. Miller et al. [2024] proposed Expected Diverse Utility, which identifies diverse ϵ-optimal solutions (locally optimal within tolerance ϵ of the global optimum) using a closed-form acquisition function that enables efficient optimization. Kent et al. [2024] extended BO to quality-diversity search by modeling both objectives and descriptors with Gaussian processes (GPs) and proposing an Expected Improvement (EI)-based acquisition to efficiently identify the best solution in each predefined region of the descriptor space.

3 Problem Statement

In this section, we formally introduce the Multi-objective Bayesian Optimization with Cardinality-Aware Pareto Selection (MOBO-CAPS) problem.

The goal is to maximize K expensive objective functions $\mathbf{f} = (f_1, \ldots, f_K)$ over the search space $\mathcal{X} \subseteq \mathbb{R}^d$. In this paper, we use *solution* to refer to an input $x \in \mathcal{X}$ and *point* to refer to an output $y \in \mathbb{R}^K$. That is, a point $\mathbf{f}(x)$ is given by the objective values obtained by the solution x, observed without noise. We use the standard maximization convention: a solution x is Pareto optimal if no $x' \in \mathcal{X}$ satisfies $\mathbf{f}(x') \geq \mathbf{f}(x)$ componentwise with strict inequality in at least one objective.

We measure the quality of a finite set of objective vectors via the *hypervolume* (HV) indicator. Given a reference point $z_{\mathrm{ref}} \in \mathbb{R}^K$ dominated by all points of interest, the hypervolume

of $Y \subset \mathbb{R}^K$ is the Lebesgue measure of the region dominated by Y and bounded by z_{ref}, $\text{HV}(Y; z_{\text{ref}}) := \lambda\Big(\bigcup_{y \in Y} [z_{\text{ref},1}, y_1] \times \cdots \times [z_{\text{ref},K}, y_K]\Big)$, where $\lambda(\cdot)$ denotes the K-dimensional Lebesgue measure. We write $\text{HV}(Y)$ when z_{ref} is clear from context.

We use *CAPS-set* to refer to a solution set of cardinality M returned by an algorithm as a cardinality-constrained approximation of the Pareto set. Our goal is to find a CAPS-set $X^\star := \{x_1^\star, \ldots, x_M^\star\}$ that maximizes the hypervolume:

$$X^\star \in \arg \max_{\substack{X \subseteq \mathcal{X} \\ |X|=M}} \text{HV}\big(\mathbf{f}(X)\big),$$

where $X = \{x_1, \ldots, x_M\}$ denotes a set of M solutions and $\mathbf{f}(X) := \{\mathbf{f}(x) : x \in X\}$.

In what follows, $\mathcal{P}_t$ denotes the current approximate Pareto set in iteration t, and $\mathcal{P}_t^M$ is the currently best CAPS-set in iteration t. Moreover, $\mathbf{f}(x)$ gives the true objective values of x, while $\hat{\mathbf{f}}(x)$ denotes the posterior predictive distribution induced by the K surrogate models.

Notice that as M becomes large, the MOBO-CAPS problem gradually approaches the standard MOBO setting, since returning many Pareto-optimal solutions increasingly resembles approximating the full Pareto front. In this regime, existing MOBO methods are often sufficient, especially when combined with a post-hoc selection procedure. Conversely, the cardinality constraint is most meaningful when only a small number of final solutions is required. A small value of the cardinality parameter M is common in real-world applications. For example, a car company may only release three vehicle variants, such as an economy model, a balanced model, and a high-performance model. Similarly, in pharmaceutical product development, a company may optimize a topical pain-relief gel for absorption, irritation risk, and manufacturing cost, but only commercialize two strengths, such as regular-strength and higher-strength formulations, due to regulatory and production requirements. These practical constraints motivate our focus on small values of M in the experiments. This choice is also motivated by the empirical results in Appendix A.3.4, where the performance gap between MOBO-CAPS and standard MOBO with post-hoc selection generally becomes smaller as M increases, although this trend is not consistent across all test problems.

4 Proposed Approach

A naive approach for MOBO-CAPS is post-hoc selection, where we run a standard MOBO algorithm to approximate the Pareto front and then extract the M solutions that maximize the hypervolume. This approach is conceptually simple and theoretically sound. If MOBO recovers the true Pareto set, post-hoc selection will identify the optimal CAPS-set. However, this approach raises sample efficiency concerns, as it utilizes cardinality information only at the final step rather than incorporating the size constraint already during optimization. Figures 1 and 3 show examples where a MOBO algorithm will evaluate solutions that contribute to the total hypervolume but are not useful for maximizing the hypervolume of a best CAPS-set, because in a cardinality-constrained setting a new solution has to replace another solution.

To address these limitations by leveraging cardinality information during optimization, one could consider a Bayes-optimal one-step approach that selects the next solution maximizing the expected improvement of the CAPS-set hypervolume among all observed solutions. However, this method requires computationally expensive Monte Carlo simulation and demands identification of the optimal CAPS-set for each fantasy value, rendering it impractical for real-world applications.

Therefore, we propose two computationally tractable methods that efficiently incorporate cardinality information to improve MOBO performance: Replacement-aware Expected Hypervolume Improvement (REHVI) and Collective Higher-dimensional Optimization with Upper Confidence Bound (CHO-UCB). REHVI is a variant of EHVI that accounts for a fixed-cardinality solution set by removing one existing point from the current CAPS-set when a new point is added. On the other hand, CHO-UCB reformulates the multi-objective optimization problem as a higher-dimensional single-objective problem, enabling joint selection of multiple solutions via a collective acquisition function.

4.1 Replacement-aware Expected Hypervolume Improvement (REHVI)

Given the sample efficiency of EHVI in MOBO, we adapt this acquisition function to our problem setting as follows. Assuming the current best CAPS-set in iteration t is $\mathcal{P}_t^M$, the replacement-aware hypervolume improvement of a new point y is calculated as

$$\text{RHVI}(y) = \max_{y' \in \mathbf{f}(\mathcal{P}_t^M) \cup \{y\}} \Big\{ \text{HV}\big(\mathbf{f}(\mathcal{P}_t^M) \cup \{y\} \setminus \{y'\}\big) - \text{HV}\big(\mathbf{f}(\mathcal{P}_t^M)\big) \Big\}.$$

The idea behind this formula is that we can only maintain M solutions in the CAPS-set $\mathcal{P}_t^M$. Therefore, when we introduce a new point y, we must remove a point y'. Naturally, we should remove the element that results in the largest gain in hypervolume.

Therefore, the expected improvement when evaluating a solution x is

$$\text{REHVI}(x) = \mathbb{E}[\text{RHVI}(\hat{\mathbf{f}}(x))],$$

where the expectation is taken with respect to the posterior predictive distribution of $\hat{\mathbf{f}}$.

The acquisition function is called Replacement-aware Expected Hypervolume Improvement (REHVI). An illustration highlighting the difference in solution selection between REHVI and EHVI is provided in Figure 1. Since REHVI is defined as a cardinality-aware generalization of EHVI, when the current non-dominated set has fewer than M points, replacement is not yet meaningful, and the criterion reduces to standard EHVI. Once $|\mathcal{P}_t| \geq M$, REHVI evaluates the improvement obtained by replacing one point in the current incumbent CAPS-set.

The implementation and computational complexity of the REHVI acquisition function are discussed in Appendix A.4. The full MOBO-CAPS algorithm using REHVI is given in Appendix A.1.

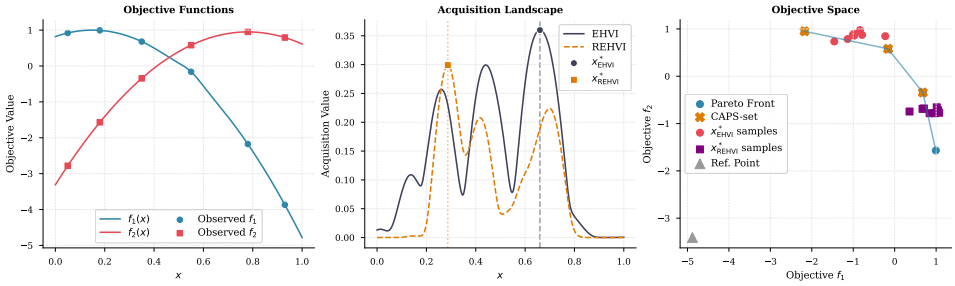

Figure 1: Difference between REHVI and EHVI on a bi-objective problem. (a) The two underlying objective functions and the observed solutions. (b) The acquisition landscapes of REHVI and EHVI, with their respective maximizers. (c) The current CAPS-set and posterior samples of the solutions selected by EHVI and REHVI. The REHVI-selected solution (purple) yields a larger replacement-aware hypervolume gain, despite contributing less to the total hypervolume on its own.

We emphasize that our proposed REHVI is distinct from NEHVI (Noisy Expected Hypervolume Improvement) introduced by [Daulton et al., 2021]. The contributions are orthogonal: while NEHVI extends EHVI to handle uncertainty from noisy objective evaluations, REHVI modifies the EHVI formulation to account for the replacement dynamics required when optimizing under a strict, fixed-cardinality constraint.

4.2 Collective Higher-dimensional Optimization with Upper Confidence Bound (CHO-UCB)

While REHVI optimizes the CAPS-set through iterative single-point replacements, an alternative perspective is to directly treat the entire CAPS-set as a collective decision variable. Following this idea, we propose Collective Higher-dimensional Optimization (CHO): rather than approaching this as a d-dimensional multi-objective optimization problem, we formulate it as a single $M \times d$-dimensional problem that optimizes all M solutions jointly.

A natural starting point for CHO is a set-valued extension of EI: $\text{CHO-EI}(X) = \mathbb{E}\left[\left\{\text{HV}(\hat{\mathbf{f}}(X)) - \text{HV}\big(\mathbf{f}(\mathcal{P}_t^M)\big)\right\}^+\right]$, where $X = \{x_1, \ldots, x_M\}$ is a set of M solutions and $\mathcal{P}_t^M$ is

the current best CAPS-set. In practice, however, CHO-EI is difficult to optimize: as the incumbent hypervolume approaches its optimum, the acquisition landscape becomes flat, the expected improvement vanishes, and gradient-based optimization stalls, even for moderate M. We therefore do not pursue CHO-EI further. Additional discussion and ablations are provided in Appendix A.7.1 and Appendix A.5.

Instead, we propose Collective Higher-dimensional Optimization with Upper Confidence Bound (CHO-UCB), which preserves the collective set-based formulation but computes the hypervolume of an optimistic M-point set under GP upper confidence bounds. This avoids Monte Carlo estimation and provides a smoother acquisition landscape.

Following standard procedure, we calculate the UCB as $\mathrm{UCB}(X) = \mu(X) + \sqrt{\beta_t}\sigma(X)$, where $\mu(\cdot)$ and $\sigma(\cdot)$ denote the posterior mean and standard deviation of the surrogate model at solutions X, and $\beta_t > 0$ is a hyperparameter trading off exploration and exploitation. We then approximate the Pareto front using these UCB values and compute the corresponding hypervolume:

$$\text{CHO-UCB}(X) = \mathrm{HV}(\mathrm{UCB}(X)),$$

where $X = \{x_1, \ldots, x_M\}$ is a set of M solutions.

A key advantage of the CHO-UCB formulation is that it admits a theoretical guarantee: under standard RKHS assumptions, its simple regret converges to zero asymptotically as the number of samples increases. While regret guarantees have been established for standard MOBO, including cumulative regret bounds [Chowdhury and Gopalan, 2021] and sublinear hypervolume regret achieved via scalarizations [Zhang, 2024], these foundational analyses assume an unconstrained Pareto set approximation, and they do not translate to settings with strict budget limits on the solution set size. To the best of our knowledge, our theoretical analysis provides the first simple regret bound specifically derived for MOBO under fixed-cardinality constraints.

A key technical challenge in analyzing CHO-UCB is that the hypervolume indicator is defined over sets of points rather than individual points, which prevents the direct application of standard UCB proof techniques. We resolve this by introducing the matching distance on multisets (Definition B.1) and proving that the hypervolume indicator is Lipschitz continuous with respect to this distance (Lemma B.5). This result, which may be of independent interest, allows us to reduce the set-valued regret to pointwise posterior uncertainty bounds.

Our analysis requires two standard regularity conditions. The first ensures hypervolume calculations are well-defined, while the second guarantees the objective functions are compatible with Gaussian process (GP) modeling.

Assumption 4.1 (Bounded objectives)**.** We fix a reference point $z_{\mathrm{ref}} \in \mathbb{R}^K$ and an upper bound vector $u \in \mathbb{R}^K$ satisfying strict dominance $u \succ z_{\mathrm{ref}}$. We assume that the true objective functions are bounded from above by u:

$$\mathbf{f}(x) \preceq u \quad \text{for all } x \in \mathcal{X}.$$

Assumption 4.2 (RKHS regularity)**.** For each $i \in [K]$, f_i lies in the RKHS $\mathcal{H}_{k_i}$ of a known positive definite kernel $k_i : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$. Moreover, $k_i(x, x) \leq 1$ for all $x \in \mathcal{X}$ and $\|f_i\|_{\mathcal{H}_{k_i}} \leq B$ for some $B > 0$.

Theorem 4.3 (Noise-free Simple Regret of Batched CHO-UCB)**.** *Suppose the objective functions f_i satisfy the RKHS Assumption 4.2 with bound B, and the problem domain is bounded as per Assumption 4.1 with $u = (2B, \ldots, 2B)$. Setting the exploration parameter $\sqrt{\beta_t} = B$ for all t, the simple regret of Batched CHO-UCB satisfies:*

$$r_T^{\mathrm{simple}} \;\leq\; \frac{2BMK(2B - \min_j z_{\mathrm{ref},j})^{K-1}}{T} \sum_{i=1}^{K} C_i(T, d, \theta_i),$$

where $C_i(T, d, \theta_i)$ denotes the cumulative posterior standard deviation bound given in Lemma 3 of Iwazaki [2025], and θ_i collects the hyperparameters of the i-th objective model. For ease of reference, we restate this result as Lemma B.11.

Due to space constraints, the full proof for the batched CHO-UCB formulation is deferred to Appendix B. The analysis follows an optimism-based UCB framework, together with one new geometric ingredient tailored to our set-valued hypervolume objective. First, valid confidence intervals for the objective functions imply that the instantaneous regret can be bounded by the gap

between the hypervolume of the upper and lower confidence bounds of the selected set. To relate this quantity to pointwise posterior uncertainty, Lemma B.5 shows that the hypervolume indicator is Lipschitz continuous with respect to the matching distance on multisets (Definition B.1). This lets us upper bound the regret by the maximum of posterior standard deviations at the queried points. Finally, combining this reduction with the cumulative uncertainty bounds of Iwazaki [2025] yields a sublinear simple regret bound.

The implementation and computational complexity of the CHO-UCB acquisition function are discussed in Appendix A.4.

The CHO-UCB formulation analyzed above is batched: all M points selected by the collective UCB objective are evaluated in each round. In practice, many applications allow only sequential evaluations, so we use a scheduler-based variant: CHO-UCB first constructs the same optimistic M-point candidate set, and an EHVI scheduler selects one point for evaluation. The scheduler is a practical sequentialization layer that enables faster feedback and avoids committing to an entire batch. Thus, the regret theorem applies to the underlying batched rule, while analysis of the scheduler-based variant is left for future work. Empirically (Appendix A.6), the EHVI scheduler outperforms batched evaluation under equal budgets. We additionally compare three schedulers: random, uncertainty (variance) and SMS-EGO. The random scheduler performs worst, followed by the batched variant, while the remaining three schedulers achieve comparable performance. This insensitivity to the specific scheduling heuristic indicates that the cardinality-aware set construction is a key performance driver, whereas the selection criterion is not. In the experiments, "CHO-UCB" refers to this EHVI scheduler-based variant unless otherwise noted.

5 Experimental Evaluation

We study five commonly-used synthetic functions: BraninCurrin (2D, 2 objectives), DTLZ1 (3D, 2 objectives), DTLZ2 (3D, 2 objectives), VLMOP2 (6D, 2 objectives), and VLMOP3 (2D, 3 objectives). To examine the effect of input dimensionality, we additionally evaluate DTLZ2 with 6D and 10D inputs for $M = 3$. Besides, to examine the effect of the number of objectives on performance, we evaluate DTLZ2 with 6D inputs and 2, 3, and 4 objectives. We also consider four real-world tasks spanning diverse application domains, including drug synthesis, chemical manufacturing, and mechanical design: VehicleSafety (5D, 3 objectives; Tanabe and Ishibuchi, 2020), Truss Design (4D, 2 objectives; Cheng and Li, 1999), Penicillin (7D, 3 objectives; Liang and Lai, 2021), and SnAr (4D, 2 objectives; Hone et al., 2017). Detailed problem descriptions are provided in Appendix A.3.1.

Each problem is evaluated for $M \in \{1, 2, 3, 4, 5\}$, yielding 45 test cases in total. We focus on relatively small M, since MOBO-CAPS is primarily motivated by applications in which the decision maker can only inspect or deploy a small set of solutions; as M becomes large, the problem increasingly approaches standard MOBO. In Figure 4 of Appendix A.3.1, we provide several examples of the hypervolume coverage of the CAPS-set relative to the full Pareto front. Across the four examples, the $M = 3$ CAPS-set covers between 69% and 99.7% of the Pareto-front hypervolume, further supporting our focus on small cardinalities. Furthermore, while practical considerations prioritize small M, we also evaluate $M = 10, 15$ on DTLZ2 to demonstrate the scalability of REHVI and CHO-UCB.

We compare against a broad suite of state-of-the-art methods that are applicable to these problems, although they have not been specifically designed for them: Sobol, EHVI, SMS-EGO [Ponweiser et al., 2008], ParEGO [BoTorch Team, 2023], its reference-point variant denoted as ParEGO*, JES [Tu et al., 2022] and HV-KG [Daulton et al., 2023]. JES and HV-KG exploit knowledge of the target recommendation size by setting the number of Pareto samples or posterior-mean Pareto candidates to M. Detailed descriptions of these benchmark algorithms are provided in Appendix A.3.3. Note that REHVI can be viewed as a cardinality-aware variant of EHVI, while CHO-UCB and SMS-EGO are both based on UCB-style hypervolume criteria. Comparing them is therefore important for demonstrating the value of explicitly incorporating the cardinality constraint in MOBO-CAPS.

All algorithms are implemented in `BoTorch` [Balandat et al., 2020]. Each run begins with $\min(3d, 10)$ random initial evaluations. For all methods, K independent GPs are trained for the K objectives

using a Matérn 5/2 kernel with Automatic Relevance Determination (ARD). Acquisition functions are optimized using multi-start L-BFGS with 12 restarts and 256 raw initial samples, except for HV-KG, for which we use 1 restart following standard `BoTorch` practice[2] due to their high computational cost. The Monte Carlo sample size is fixed at 256. To choose starting points for L-BFGS, we adopt the Sample-around-Pareto Initialization (SPI). We implemented SPI in `BoTorch` by setting `sample_around_best=True`, which draws additional candidates in the vicinity of the predicted Pareto set. This strategy is inspired by Random Axis-Aligned Subspace Perturbation (RAASP) [Papenmeier et al., 2025, Rashidi et al., 2024], but adapted to the MOBO setting. We found that SPI substantially improves REHVI performance; an ablation study is provided in Appendix A.5. For fairness, the same initialization is used for CHO-UCB and all baseline methods. We further show that SPI generally does not degrade post-hoc MOBO and CHO-UCB performance, see Appendix A.5 for details. We set the CHO-UCB hyperparameter $\beta_t = 4$.

For each algorithm, the incumbent CAPS-set is updated at every iteration. Note that the computational cost of identifying the CAPS-set increases with M. We employ exact algorithms to accelerate this step. In the bi-objective case, we use a dynamic programming method that returns the exact optimum in polynomial time [Bringmann and Friedrich, 2012]. For problems with more than two objectives, we use an exact Branch-and-Bound algorithm [Gomes et al., 2018] to find a global optimum.

We measure performance via simple regret, defined as the difference between the true maximum hypervolume and the best hypervolume obtained by the algorithm so far. The true maximum hypervolume is approximated via multi-restart L-BFGS optimization on the $M \times d$-dimensional search space. We repeat each experiment 20 times independently. Results are reported as mean $\pm$ one standard error, and for readability, we apply a logarithmic scale to the y-axis where appropriate; values smaller than 10^{-6} are clipped to 10^{-6} to ensure numerical stability.

Due to space constraints, the main paper reports the $M = 3$ results for all nine test problems. Table 1 summarizes the average ranks, computed from the mean simple regret, across all values of M and all test problems. The complete results are provided in Figure 5 and Table 2 in Appendix A.3.

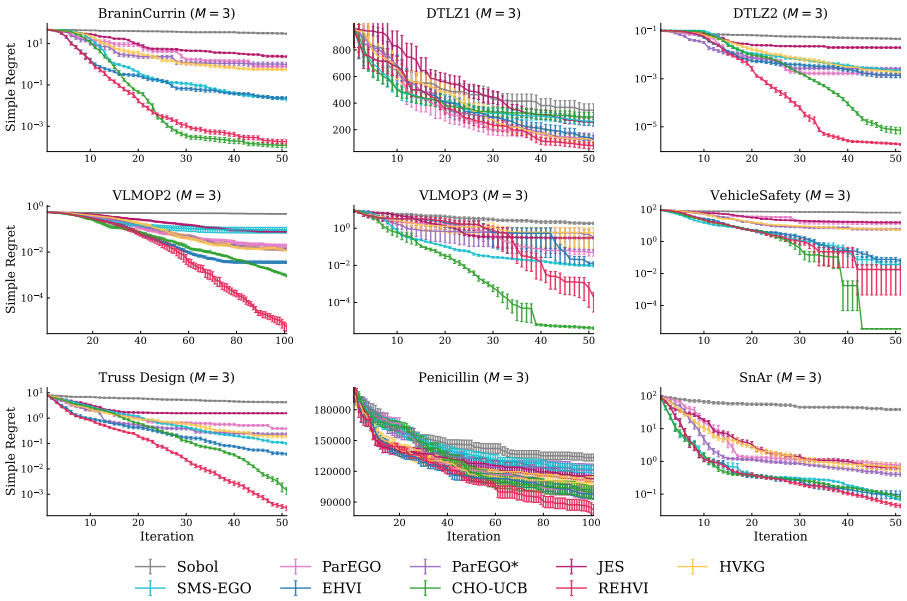

Figure 2: The cardinality-aware methods, REHVI and CHO-UCB, generally achieve the strongest performance among the compared methods.

Figure 2 shows that REHVI and CHO-UCB achieve the strongest overall performance in the benchmark suite, although the advantage varies. This is confirmed by the average ranks in Table 1, where REHVI leads, followed by CHO-UCB and then EHVI with post-hoc selection.

[2]https://archive.botorch.org/tutorials/decoupled_mobo

Table 1: Average rank of each algorithm across all benchmarks. Lower is better. The first row reports the average over all 45 settings, while the remaining rows report averages separately for each cardinality M. Standard errors are shown in parentheses.

	MOBO-CAPS Methods		**Baselines**						
Setting	**REHVI**	**CHO-UCB**	EHVI	SMS-EGO	HV-KG	ParEGO*	ParEGO	JES	Sobol
Overall	**1.58** (0.20)	**2.51** (0.30)	3.40 (0.12)	5.11 (0.31)	5.13 (0.21)	5.49 (0.23)	5.62 (0.25)	7.22 (0.13)	8.93 (0.07)
$M = 1$	**2.00** (0.65)	**2.67** (0.85)	3.78 (0.32)	5.67 (0.71)	4.89 (0.51)	4.89 (0.63)	5.11 (0.65)	7.33 (0.37)	8.67 (0.33)
$M = 2$	**1.11** (0.11)	**2.67** (0.58)	3.33 (0.24)	5.33 (0.76)	5.22 (0.60)	5.33 (0.37)	5.78 (0.55)	7.22 (0.28)	9.00 (0.00)
$M = 3$	**1.33** (0.17)	**2.67** (0.75)	3.56 (0.24)	4.89 (0.79)	5.11 (0.56)	5.89 (0.45)	5.56 (0.58)	7.00 (0.33)	9.00 (0.00)
$M = 4$	**2.11** (0.75)	**2.11** (0.54)	3.11 (0.20)	4.89 (0.75)	5.44 (0.18)	5.67 (0.62)	5.56 (0.58)	7.11 (0.26)	9.00 (0.00)
$M = 5$	**1.33** (0.17)	**2.44** (0.73)	3.22 (0.28)	4.78 (0.49)	5.00 (0.47)	5.67 (0.50)	6.11 (0.51)	7.44 (0.18)	9.00 (0.00)

According to in Figure 5 in Appendix A.3.4, the performance gap between MOBO-CAPS and standard MOBO with post-hoc selection generally becomes smaller as M increases, although this trend is not consistent across all test problems. One possible explanation is that, as M increases, the cardinality constraint becomes less restrictive, potentially reducing the benefit of explicitly optimizing the composition of the solution set.

Overall, REHVI and CHO-UCB outperform standard MOBO with post-hoc selection on average across the benchmark suite. REHVI achieves the strongest overall performance, ranking first in 30 out of 45 settings. This advantage is consistent with the relative strength of the underlying acquisitions: EHVI generally outperforms SMS-EGO, and its replacement-aware counterpart REHVI similarly tends to outperform CHO-UCB. To quantify this relationship, we test whether the pairwise ordering between EHVI and SMS-EGO agrees with that between REHVI and CHO-UCB more often than chance. A two-sided binomial test rejects the null hypothesis of random agreement, yielding $p = 2.4589 \times 10^{-3}$.

We also evaluate DTLZ2 across different input dimensions (3D, 6D, and 10D) and different objective numbers (2,3 and 4 objectives), see Figure 6 in Appendix A.3.5. We observe that the relative performance ranking among algorithms remains broadly consistent across all dimensionalities and all objective numbers. In addition, we test DTLZ2 for $M = 10, 15$ (Figure 7 in Appendix A.3.5). Again we observe that the relative advantage of CHO-UCB and REHVI over baseline acquisition functions decreases as M increases.

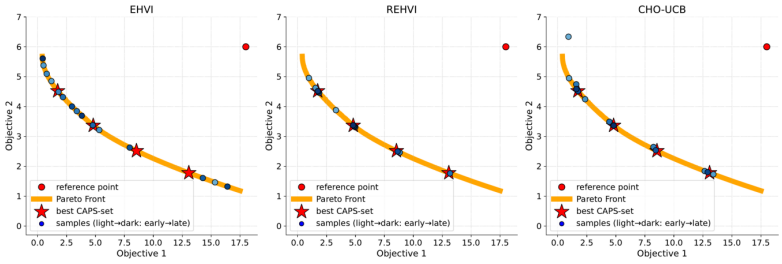

Figure 3: Examples of sampling behavior of acquisition functions on BraninCurrin ($M = 4$)

To better understand the behavior of different acquisition functions, we plot the points sampled by EHVI, CHO-UCB, and REHVI. Figure 3 visualizes the sampling behavior on BraninCurrin ($M = 4$). EHVI spreads evaluations along the Pareto front, while CHO-UCB and REHVI concentrate around the best size-M solutions; REHVI is the most exploitative among the three.

We also compare the wall-clock times of REHVI and CHO-UCB acquisition optimization, see Figure 8 in Appendix A.4. For the values of M in our experiments, CHO-UCB is faster to optimize than REHVI.

Overall, we recommend REHVI for practical applications. Nevertheless, CHO-UCB remains attractive because its batched formulation admits a formal regret bound, and its acquisition function is computationally cheaper than REHVI.

Methodological Caveats. When interpreting these empirical results, two methodological caveats should be noted. First, regarding our statistical significance reporting for the binomial test, the 45 evaluated settings are not strictly independent because they use the same set of test problems for different cardinalities, thus the reported statistical significance may be slightly inflated. Second, the calculation of regret relies on approximating the true maximum hypervolume using multi-start L-BFGS. While this approximation may occasionally yield suboptimal targets, potentially compressing the absolute regret values, this affects all evaluated methods equally. Therefore, while the absolute regret values are subject to this approximation error, the relative ordinal rankings of the algorithms and the resulting comparative analyses remain robust.

Limitations. Our method excels when the surrogate model delivers accurate predictions with well-calibrated uncertainty, as this allows the cardinality-aware acquisition to make reliable selections. On problems where the Pareto front is difficult to identify within the given evaluation budget (e.g., DTLZ1), the posterior-induced approximation may remain coarse, and explicitly reasoning about CAPS yields limited additional benefit over standard hypervolume-based approaches. Consequently, the performance gains of our method are less consistent on such challenging benchmarks.

Our main experiments focus on relatively small cardinalities ($M \leq 5$), low objective dimensions ($K \leq 3$), and moderate input dimensions ($d \leq 10$). Although the DTLZ2 case study provides preliminary evidence for larger M and K, scalability to larger cardinalities, many-objective settings, and high-dimensional input spaces remains to be studied.

Addressing these limitations represents a valuable avenue for future work.

6 Conclusion

We have argued that for many practical applications, the traditional goal of Multi-objective Bayesian Optimization, i.e. approximating the entire Pareto front, is misaligned with the common end-goal of selecting a small, optimal set of solutions. To address this, we introduced a new problem setting, Multi-objective Bayesian Optimization with Cardinality-Aware Pareto Selection (MOBO-CAPS), which directly incorporates the cardinality constraint into the optimization process.

Our proposed algorithms, Replacement-aware Expected Hypervolume Improvement (REHVI) and Collective Higher-dimensional Optimization with Upper Confidence Bound (CHO-UCB), are designed specifically for this setting. Extensive experiments demonstrated that by making the optimization process cardinality-aware, our methods consistently outperform strong baselines that rely on post-hoc selection. We found REHVI to be a powerful empirical performer, while CHO-UCB offers a computationally efficient alternative with favorable theoretical guarantees.

This work opens several promising avenues for future research. Extending REHVI to support batch evaluations and developing a batched version of CHO-UCB with a batch size independent of M would enhance its practical utility. On the methodological front, developing scalable surrogate models that respect the permutation invariance of solution sets could further improve the performance of collective optimization strategies like CHO-EI and CHO-UCB. Finally, moving beyond hypervolume to incorporate other decision-maker preferences, such as fairness or diverse coverage criteria, represents a valuable next step in making Bayesian optimization a more effective tool for real-world decision-making.

Acknowledgments

We sincerely thank the anonymous reviewers for their valuable comments and constructive suggestions, which have greatly helped improve the quality of this paper. The first author gratefully acknowledges support by the Engineering and Physical Sciences Research Council through the Mathematics of Systems II Centre for Doctoral Training at the University of Warwick (reference EP/S022244/1).

References

Alaleh Ahmadianshalchi, Syrine Belakaria, and Janardhan Rao Doppa. Pareto front-diverse batch multi-objective Bayesian optimization. In *AAAI Conference on Artificial Intelligence*, volume 38, pages 10784–10794, 2024.

Raul Astudillo and Peter Frazier. Multi-attribute Bayesian optimization with interactive preference learning. In *International Conference on Artificial Intelligence and Statistics*, pages 4496–4507. PMLR, 2020.

Anne Auger, Johannes Bader, Dimo Brockhoff, and Eckart Zitzler. Theory of the hypervolume indicator: optimal μ-distributions and the choice of the reference point. In *ACM SIGEVO workshop on Foundations of Genetic Algorithms*, pages 87–102, 2009.

Maximilian Balandat, Brian Karrer, Daniel Jiang, Samuel Daulton, Ben Letham, Andrew G Wilson, and Eytan Bakshy. BoTorch: A framework for efficient Monte-Carlo Bayesian optimization. *Advances in Neural Information Processing Systems*, 33:21524–21538, 2020.

Arpan Biswas, Anna N Morozovska, Maxim Ziatdinov, Eugene A Eliseev, and Sergei V Kalinin. Multi-objective Bayesian optimization of ferroelectric materials with interfacial control for memory and energy storage applications. *Journal of Applied Physics*, 130(20), 2021.

BoTorch Team. Multi-Objective Bayesian Optimization Tutorial. `https://archive.botorch.org/tutorials/multi_objective_bo`, 2023. Accessed: Feb. 17, 2025.

Karl Bringmann and Tobias Friedrich. An efficient algorithm for computing hypervolume contributions. *Evolutionary Computation*, 18(3):383–402, 2010.

Karl Bringmann and Tobias Friedrich. Approximating the least hypervolume contributor: NP-hard in general, but fast in practice. *Theoretical Computer Science*, 425:104–116, 2012.

Weiyu Chen, Hisao Ishibuchi, and Ke Shang. Fast greedy subset selection from large candidate solution sets in evolutionary multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, 26(4):750–764, 2021.

Franklin Y Cheng and XS Li. Generalized center method for multiobjective engineering optimization. *Engineering Optimization*, 31(5):641–661, 1999.

Sayak Ray Chowdhury and Aditya Gopalan. No-regret algorithms for multi-task Bayesian optimization. In *International Conference on Artificial Intelligence and Statistics*, pages 1873–1881. PMLR, 2021.

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

Sam Daulton, Maximilian Balandat, and Eytan Bakshy. Hypervolume knowledge gradient: a lookahead approach for multi-objective Bayesian optimization with partial information. In *International Conference on Machine Learning*, pages 7167–7204. PMLR, 2023.

Samuel Daulton, Maximilian Balandat, and Eytan Bakshy. Differentiable expected hypervolume improvement for parallel multi-objective Bayesian optimization. *Advances in Neural Information Processing Systems*, 33:9851–9864, 2020.

Samuel Daulton, Maximilian Balandat, and Eytan Bakshy. Parallel Bayesian optimization of multiple noisy objectives with expected hypervolume improvement. *Advances in neural information processing systems*, 34:2187–2200, 2021.

Kalyanmoy Deb, Lothar Thiele, Marco Laumanns, and Eckart Zitzler. Scalable test problems for evolutionary multiobjective optimization. In *Evolutionary multiobjective optimization: theoretical advances and applications*, pages 105–145. Springer, 2005.

Michael Emmerich and Jan-willem Klinkenberg. The computation of the expected improvement in dominated hypervolume of Pareto front approximations. *Rapport technique, Leiden University*, 34: 7–3, 2008.

Michael Emmerich, André Deutz, and Nicola Beume. Gradient-based/evolutionary relay hybrid for computing Pareto front approximations maximizing the S-metric. In *Hybrid Metaheuristics: International Workshop*, pages 140–156. Springer, 2007.

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

Tobias Glasmachers. Optimized approximation sets for low-dimensional benchmark Pareto fronts. In *International Conference on Parallel Problem Solving from Nature*, pages 569–578. Springer, 2014.

Ricardo J Gomes, Andreia P Guerreiro, Tobias Kuhn, and Luís Paquete. Implicit enumeration strategies for the hypervolume subset selection problem. *Computers & Operations Research*, 100: 244–253, 2018.

Arash Heidari, Jixiang Qing, Sebastian Rojas Gonzalez, Jürgen Branke, Tom Dhaene, and Ivo Couckuyt. Finding knees in Bayesian multi-objective optimization. In *International Conference on Parallel Problem Solving from Nature*, pages 104–117. Springer, 2022.

Arash Heidari, Jixiang Qing, Sebastian Rojas Gonzalez, Juergen Branke, Tom Dhaene, and Ivo Couckuyt. Knee detection in Bayesian multi-objective optimization using thompson sampling. *IEEE Transactions on Evolutionary Computation*, 2024.

Christopher A Hone, Nicholas Holmes, Geoffrey R Akien, Richard A Bourne, and Frans L Muller. Rapid multistep kinetic model generation from transient flow data. *Reaction Chemistry & Engineering*, 2(2):103–108, 2017.

Hisao Ishibuchi, Yuji Sakane, Noritaka Tsukamoto, and Yusuke Nojima. Selecting a small number of representative non-dominated solutions by a hypervolume-based solution selection approach. In *2009 IEEE International Conference on Fuzzy Systems*, pages 1609–1614. IEEE, 2009.

Hisao Ishibuchi, Hiroyuki Masuda, and Yusuke Nojima. Selecting a small number of non-dominated solutions to be presented to the decision maker. In *IEEE International Conference on Systems, Man, and Cybernetics*, pages 3816–3821. IEEE, 2014.

Shogo Iwazaki. Gaussian Process Upper Confidence Bound Achieves Nearly-Optimal Regret in Noise-Free Gaussian Process Bandits. In *Advances in Neural Information Processing Systems*, 2025.

Paul Kent, Adam Gaier, Jean-Baptiste Mouret, and Juergen Branke. Bayesian optimisation for quality diversity search with coupled descriptor functions. *IEEE Transactions on Evolutionary Computation*, 2024.

Joshua Knowles. ParEGO: A hybrid algorithm with on-line landscape approximation for expensive multiobjective optimization problems. *IEEE Transactions on Evolutionary Computation*, 10(1): 50–66, 2006.

Tobias Kuhn, Carlos M Fonseca, Luís Paquete, Stefan Ruzika, Miguel M Duarte, and José Rui Figueira. Hypervolume subset selection in two dimensions: Formulations and algorithms. *Evolutionary Computation*, 24(3):411–425, 2016.

Qiaohao Liang and Lipeng Lai. Scalable Bayesian optimization accelerates process optimization of penicillin production. In *NeurIPS 2021 AI for Science Workshop*, 2021.

Xingtao Liao, Qing Li, Xujing Yang, Weigang Zhang, and Wei Li. Multiobjective optimization for crash safety design of vehicles using stepwise regression model. *Structural and Multidisciplinary Optimization*, 35(6):561–569, 2008.

Zhiyuan Jerry Lin, Raul Astudillo, Peter Frazier, and Eytan Bakshy. Preference exploration for efficient Bayesian optimization with multiple outcomes. In *International Conference on Artificial Intelligence and Statistics*, pages 4235–4258. PMLR, 2022.

Georgios Makrygiorgos, Angelo D Bonzanini, Victor Miller, and Ali Mesbah. Performance-oriented model learning for control via multi-objective Bayesian optimization. *Computers & Chemical Engineering*, 162:107770, 2022.

Natalie Maus, Kaiwen Wu, David Eriksson, and Jacob Gardner. Discovering Many Diverse Solutions with Bayesian Optimization. In *International Conference on Artificial Intelligence and Statistics*, pages 1779–1798. PMLR, 2023.

Natalie Maus, Kyurae Kim, Yimeng Zeng, Haydn Thomas Jones, Fangping Wan, Marcelo Der Torossian Torres, Cesar de la Fuente-Nunez, and Jacob R. Gardner. Covering Multiple Objectives with a Small Set of Solutions Using Bayesian Optimization. In *Annual Conference on Neural Information Processing Systems*, 2025.

Yongsheng Mei, Tian Lan, Mahdi Imani, and Suresh Subramaniam. A Bayesian Optimization Framework for Finding Local Optima in Expensive Multimodal Functions. In *ECAI*, 2023.

John Joshua Miller, Simon Mak, Benny Sun, Sai Ranjeet Narayanan, Suo Yang, Zongxuan Sun, Kenneth S Kim, and Chol-Bum Mike Kweon. Expected diverse utility (EDU): Diverse Bayesian optimization of expensive computer simulators. *arXiv preprint arXiv:2410.01196*, 2024.

Ryota Ozaki, Kazuki Ishikawa, Youhei Kanzaki, Shion Takeno, Ichiro Takeuchi, and Masayuki Karasuyama. Multi-objective Bayesian optimization with active preference learning. In *AAAI Conference on Artificial Intelligence*, volume 38, pages 14490–14498, 2024.

Leonard Papenmeier, Matthias Poloczek, and Luigi Nardi. Understanding High-Dimensional Bayesian Optimization. In *International Conference on Machine Learning*, 2025.

Biswajit Paria, Kirthevasan Kandasamy, and Barnabás Póczos. A flexible framework for multi-objective Bayesian optimization using random scalarizations. In *Uncertainty in Artificial Intelligence*, pages 766–776. PMLR, 2020.

Wolfgang Ponweiser, Tobias Wagner, Dirk Biermann, and Markus Vincze. Multiobjective optimization on a limited budget of evaluations using model-assisted-metric selection. In *International Conference on Parallel Problem Solving from Nature*, pages 784–794. Springer, 2008.

Bahador Rashidi, Kerrick Johnstonbaugh, and Chao Gao. Cylindrical thompson sampling for high-dimensional Bayesian optimization. In *International Conference on Artificial Intelligence and Statistics*, pages 3502–3510. PMLR, 2024.

Ke Shang, Hisao Ishibuchi, Linjun He, and Lie Meng Pang. A survey on the hypervolume indicator in evolutionary multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, 25 (1):1–20, 2020.

Niranjan Srinivas, Andreas Krause, Sham M Kakade, and Matthias Seeger. Gaussian process optimization in the bandit setting: No regret and experimental design. *arXiv preprint arXiv:0912.3995*, 2009.

Samuel Stanton, Wesley Maddox, Nate Gruver, Phillip Maffettone, Emily Delaney, Peyton Greenside, and Andrew Gordon Wilson. Accelerating Bayesian optimization for biological sequence design with denoising autoencoders. In *International Conference on Machine Learning*, pages 20459–20478. PMLR, 2022.

Ryoji Tanabe and Hisao Ishibuchi. An easy-to-use real-world multi-objective optimization problem suite. *Applied Soft Computing*, 89:106078, 2020.

Ben Tu, Axel Gandy, Nikolas Kantas, and Behrang Shafei. Joint entropy search for multi-objective Bayesian optimization. *Advances in Neural Information Processing Systems*, 35:9922–9938, 2022.

Juan Ungredda, Juergen Branke, Mariapia Marchi, and Teresa Montrone. Single interaction multi-objective Bayesian optimization. In *International Conference on Parallel Problem Solving from Nature*, pages 132–145. Springer, 2022.

David A Van Veldhuizen and Gary B Lamont. Multiobjective evolutionary algorithm test suites. In *ACM Symposium on Applied Computing*, pages 351–357, 1999.

Hanyang Wang, Juergen Branke, and Matthias Poloczek. Bayesian Optimization with Preference Exploration using a Monotonic Neural Network Ensemble. In *Advances in Neural Information Processing Systems*, 2025.

Hao Wang, André Deutz, Thomas Bäck, and Michael Emmerich. Hypervolume indicator gradient ascent multi-objective optimization. In *International Conference on Evolutionary Multi-criterion Optimization*, pages 654–669. Springer, 2017.

Kaifeng Yang, Michael Emmerich, André Deutz, and Thomas Bäck. Efficient computation of expected hypervolume improvement using box decomposition algorithms. *Journal of Global Optimization*, 75:3–34, 2019a.

Kaifeng Yang, Michael Emmerich, André Deutz, and Thomas Bäck. Multi-objective Bayesian global optimization using expected hypervolume improvement gradient. *Swarm and Evolutionary Computation*, 44:945–956, 2019b.

Qiuyi Richard Zhang. Optimal scalarizations for sublinear hypervolume regret. *Advances in Neural Information Processing Systems*, 37:39963–39999, 2024.

Eckart Zitzler and Lothar Thiele. Multiobjective optimization using evolutionary algorithms—a comparative case study. In *International Conference on Parallel Problem Solving from Nature*, pages 292–301. Springer, 1998.

A Appendix

A.1 Algorithm Pseudocodes

Algorithm 1 REHVI

Initial observation $\mathcal{D}_0$, initial Pareto set $\mathcal{P}_0$, initial CAPS-set $\mathcal{P}_0^M$
for $t = 0, 1, \ldots, T-1$ **do**
  Train K independent GPs $\hat{\mathbf{f}}_t$ given $\mathcal{D}_t$
  if $|\mathcal{P}_t| < M$ **then**
    $x_{t+1} = \arg\max_{x \in \mathcal{X}} \{\text{EHVI}(x; \mathcal{P}_t, \hat{\mathbf{f}}_t)\}$
  else
    $x_{t+1} = \arg\max_{x \in \mathcal{X}} \{\text{REHVI}(x; \mathcal{P}_t^M, \hat{\mathbf{f}}_t)\}$
  end if
  Query objective function: $y_{t+1} = \mathbf{f}(x_{t+1})$
  Update $\mathcal{D}_{t+1} = \mathcal{D}_t \cup \{x_{t+1}, y_{t+1}\}$
  Update the Pareto set $\mathcal{P}_{t+1}$ and the CAPS-set $\mathcal{P}_{t+1}^M$
end for
return the CAPS-set $\mathcal{P}_T^M$

Algorithm 2 CHO-UCB with EHVI Scheduler

Initial observation $\mathcal{D}_0$, initial Pareto set $\mathcal{P}_0$, initial CAPS-set $\mathcal{P}_M^0$
for $t = 0, 1, \ldots, T-1$ **do**
  Train K independent GPs $\hat{\mathbf{f}}_t$ given $\mathcal{D}_t$
  $X_{t+1} = \arg\max_{X \in \mathcal{X}^M} \{\text{CHO-UCB}(X; \hat{\mathbf{f}}_t)\}$ (construct an optimistic size-M candidate set)
  $x_{t+1} = \arg\max_{x \in X_{t+1}} \{\text{EHVI}(x; \mathcal{P}_t, \hat{\mathbf{f}}_t)\}$ (select one point using the EHVI scheduler)
  Query objective function: $y_{t+1} = \mathbf{f}(x_{t+1})$
  Update $\mathcal{D}_{t+1} = \mathcal{D}_t \cup \{x_{t+1}, y_{t+1}\}$
  Update the Pareto set $\mathcal{P}_{t+1}$ and the CAPS-set $\mathcal{P}_{t+1}^M$
end for
return the CAPS-set $\mathcal{P}_T^M$

A.2 Comparison of Multi-Objective Optimization Problem Settings

Setting	Metric	Cardinality	Objective Function Property
Traditional MOBO	Hypervolume	No limit	Black-box
MOBO-CAPS	Hypervolume	M	Black-box
BOPE [Lin et al., 2022]	Utility Function	1	Black-box
Knee Finding [Heidari et al., 2022]	Hypervolume	1	Black-box
Coverage BO [Maus et al., 2025]	Coverage Score	M ($M < K$)	Black-box
HIGA-MO [Wang et al., 2017]	Hypervolume	M	Accessible Gradient

A.3 Experiment

In this section, we provide supplementary material, including detailed descriptions of the benchmark optimization problems and baseline algorithms, implementation details of the proposed methods,

computational complexity analysis, and a case study examining the effects of input dimension, number of objectives, and cardinality M.

A.3.1 Test Problem

In this section, we provide detailed specifications for the benchmark optimization problems used in our evaluation. Note that reference points were selected following literature or standard `BoTorch` implementations.

BraninCurrin: This problem combines the Branin and Currin single-objective functions into a bi-objective formulation that introduces conflicting landscapes with different scales of nonlinearity. The optimization involves $d = 2$ design variables and $K = 2$ objectives. Complete problem specifications are available in Emmerich et al. [2006]. The reference point is $[-18., -6.]$.

DTLZ1: This benchmark is a standard multi-objective test function designed to evaluate an algorithm's ability to converge toward and maintain a uniform spread along a linear Pareto front. The optimization involves $d = 3$ design variables and $K = 2$ objectives. Complete problem specifications are available in Deb et al. [2005]. The reference point is $[-50., -50.]$.

DTLZ2: This test function features a nonlinear, convex Pareto front and is widely used to assess convergence accuracy and diversity maintenance in multi-objective optimization algorithms. The optimization involves $d = 3$ design variables and $K = 2$ objectives. Complete problem specifications are available in Deb et al. [2005]. The reference point is $[-1.1, -1.1]$.

VLMOP2: This benchmark consists of two smooth, nonlinear objective functions that induce a convex Pareto front, making it suitable for evaluating convergence behavior in bi-objective optimization. The optimization involves $d = 6$ design variables and $K = 2$ objectives. Complete problem specifications are available in Van Veldhuizen and Lamont [1999]. The reference point is $[-1.2, -1.2]$.

VLMOP3: This benchmark presents two smooth, nonlinear objectives that generate a concave Pareto front, providing a moderate challenge for both convergence and diversity. The optimization involves $d = 2$ design variables and $K = 3$ objectives. Complete problem specifications are available in Van Veldhuizen and Lamont [1999]. The reference point is $[-4., -25., -0.25]$.

Truss Design: This optimization problem involves designing a four-bar truss structure with 2 competing objectives: minimizing structural volume while minimizing joint displacement. The design variables correspond to the lengths of the four individual bars. Complete problem specifications are available in Cheng and Li [1999]. The reference point is $[-2886.37, -0.04]$.

SnAr: This optimization problem involves a nucleophilic aromatic substitution reaction (SnAr) between 2,4-difluoronitrobenzene and pyrrolidine in ethanol. The optimization uses 4 design variables representing initial conditions to optimize space-time yield and environmental impact. Complete problem specifications are available in Hone et al. [2017]. The reference point is $[0., 120.]$.

VehicleSafety: This multi-objective optimization problem involves designing vehicle safety systems with competing objectives of optimizing mass, collision acceleration, and toe board intrusion by designing the thickness of five reinforced members around the frontal structure. The optimization uses 5 design variables representing the thickness parameters of structural components to simultaneously optimize 3 conflicting objectives. Complete problem specifications are available in [Liao et al., 2008]. The reference point is $[1864.7, -11.82, -0.29]$.

Penicillin: This benchmark optimizes the Penicillin production process. The optimization involves 7 design variables controlling initial reaction conditions to optimize 3 objectives: yield, carbon dioxide emissions, and fermentation time. Complete problem specifications are available in Liang and Lai [2021]. The reference point is $[-1.85, -86.93, -514.7]$.

A.3.2 CAPS-set Coverage of the Pareto Front

We also report the percentage of Pareto-front hypervolume covered by the CAPS-set in Figure 4. Even for $M = 1$, the CAPS-set covers at least 36.4% of the Pareto-front hypervolume, while for $M = 5$, it covers more than 80% in all cases.

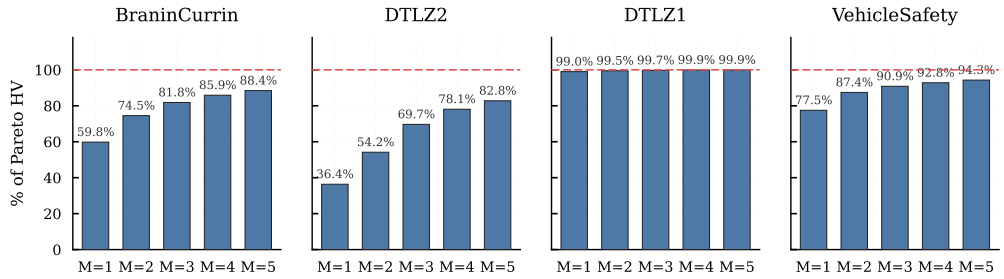

Figure 4: Percentage of Pareto Hypervolume Covered by the CAPS-set

A.3.3 Benchmark Algorithm

We compare the proposed methods against a diverse set of benchmark algorithms that are applicable to the considered problems. Below, we briefly summarize each method.

Sobol: Sobol is a quasi-random search strategy based on low-discrepancy sequences. It does not use a surrogate model and serves as a simple model-free baseline. At each iteration, a new design point is selected from a Sobol sequence within the search domain.

EHVI: EHVI [Emmerich and Klinkenberg, 2008, Daulton et al., 2020] is a standard acquisition function for multi-objective Bayesian optimization. Given the current set of observed non-dominated points, EHVI selects the candidate that maximizes the expected increase in hypervolume after one additional evaluation. EHVI is designed to improve the approximation of the entire Pareto front rather than to directly optimize a fixed-cardinality subset.

SMS-EGO: SMS-EGO [Ponweiser et al., 2008] is a sequential model-based multi-objective optimization method that extends the single-objective EGO framework to the multi-objective setting. It uses upper confidence bound to calculate the hypervolume, in order to identify the next point for evaluation. Similar to EHVI, its goal is to improve the global Pareto-front approximation rather than to optimize a set of exactly M solutions. In our experiments, we set $\beta_t = 4$ in SMS-EGO.

ParEGO: ParEGO [Knowles, 2006] scalarizes the multiple objectives into a single objective using a randomly sampled weight vector and then applies standard single-objective Bayesian optimization to the scalarized problem. By varying the scalarization over iterations, ParEGO is able to explore different parts of the Pareto front. However, it does not explicitly account for the cardinality constraint in the final recommended solution set.

ParEGO*: ParEGO* denotes a reference-point variant of ParEGO that incorporates the hypervolume reference point into the scalarization or recommendation procedure. This variant is included to provide a stronger scalarization-based baseline that is better aligned with hypervolume-oriented evaluation. Nevertheless, like ParEGO, it does not directly optimize a fixed-size subset of Pareto solutions.

JES: Joint Entropy Search (JES) [Tu et al., 2022] is an information-theoretic acquisition function that selects evaluations to maximally reduce uncertainty about the Pareto-optimal set and Pareto front. In the multi-objective setting, JES seeks to efficiently learn the Pareto front by maximizing information gain. In our experiments, we set the number of Pareto set samples to M and the number of Pareto points per sample to M. All remaining implementation details follow the standard `BoTorch` tutorial.

HV-KG: Hypervolume Knowledge Gradient (HV-KG) [Daulton et al., 2023] is a one-step lookahead acquisition function that evaluates the expected value of an observation through its effect on the posterior-mean hypervolume after conditioning on fantasy outcomes. HV-KG is especially relevant to MOBO-CAPS because its terminal recommendation set has a fixed size, denoted N_p in the original formulation. We set $N_p = M$, so that the terminal decision in HV-KG is the best posterior-mean size-M set. This makes HV-KG a strong lookahead baseline for the CAPS objective. Unless otherwise stated, the remaining implementation details follow the standard `BoTorch` implementation.

A.3.4 Full Experimental Results

Figure 5 reports the complete results across all test problems and all values of M. Table 2 presents the ranking of each algorithm based on its final regret mean for each test problem. Overall, the strongest performance is achieved by the MOBO-CAPS methods, namely REHVI and CHO-UCB.

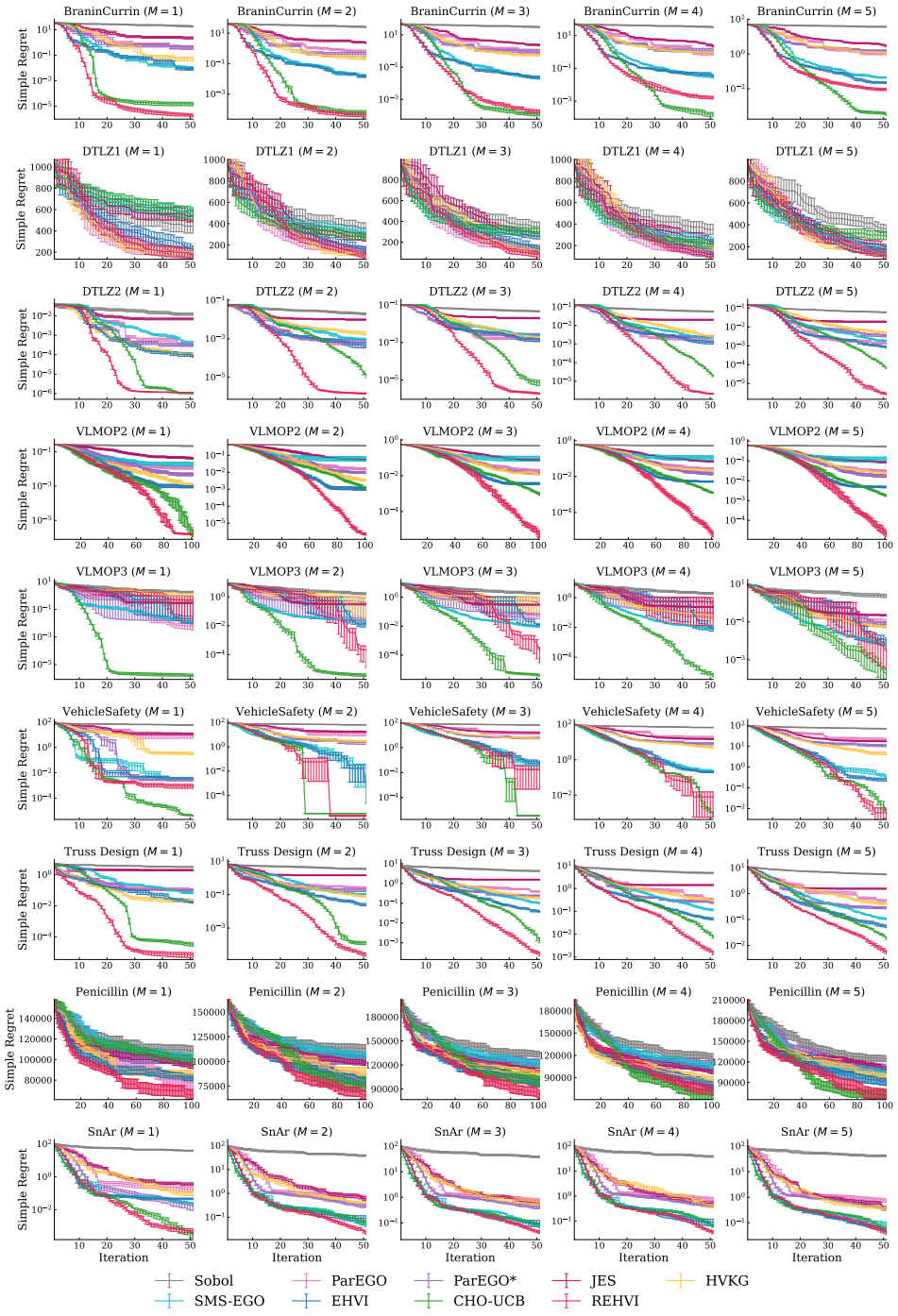

Figure 5: The cardinality-aware methods, REHVI and CHO-UCB, achieve the strongest overall performance across settings.

Table 2: Ranked performances (lower is better) of all algorithms for all benchmarks

	Sobol	ParEGO	ParEGO*	JES	HV-KG	SMS-EGO	EHVI	CHO-UCB	REHVI
BraninCurrin ($M = 1$)	9	6	7	8	5	4	3	2	**1**
BraninCurrin ($M = 2$)	9	7	6	8	5	3	4	2	**1**
BraninCurrin ($M = 3$)	9	6	7	8	5	3	4	**1**	2
BraninCurrin ($M = 4$)	9	5	7	8	6	3	4	**1**	2
BraninCurrin ($M = 5$)	9	5	7	8	6	4	3	**1**	2
DTLZ1 ($M = 1$)	6	4	2	7	3	9	5	8	**1**
DTLZ1 ($M = 2$)	9	5	3	6	2	8	4	7	**1**
DTLZ1 ($M = 3$)	9	5	3	6	2	7	4	8	**1**
DTLZ1 ($M = 4$)	9	4	2	7	5	8	3	6	**1**
DTLZ1 ($M = 5$)	9	4	3	7	2	6	5	8	**1**
DTLZ2 ($M = 1$)	9	6	5	8	4	7	3	**1**	2
DTLZ2 ($M = 2$)	9	3	5	8	7	6	4	2	**1**
DTLZ2 ($M = 3$)	9	3	7	8	5	6	4	2	**1**
DTLZ2 ($M = 4$)	9	4	7	8	6	5	3	2	**1**
DTLZ2 ($M = 5$)	9	4	6	8	7	5	3	2	**1**
VLMOP2 ($M = 1$)	9	6	5	8	4	7	3	2	**1**
VLMOP2 ($M = 2$)	9	6	5	7	4	8	2	3	**1**
VLMOP2 ($M = 3$)	9	6	5	7	4	8	3	2	**1**
VLMOP2 ($M = 4$)	9	6	4	7	5	8	3	2	**1**
VLMOP2 ($M = 5$)	9	6	4	7	5	8	3	2	**1**
VLMOP3 ($M = 1$)	9	2	5	6	8	3	4	**1**	7
VLMOP3 ($M = 2$)	9	5	6	7	8	4	3	**1**	2
VLMOP3 ($M = 3$)	9	5	7	6	8	3	4	**1**	2
VLMOP3 ($M = 4$)	9	4	7	6	5	3	2	**1**	8
VLMOP3 ($M = 5$)	9	7	6	8	5	3	4	**1**	2
VehicleSafety ($M = 1$)	9	7	3	8	6	4	5	**1**	2
VehicleSafety ($M = 2$)	9	7	5	8	6	4	3	2	**1**
VehicleSafety ($M = 3$)	9	7	6	8	5	3	4	**1**	2
VehicleSafety ($M = 4$)	9	8	6	7	5	4	3	**1**	2
VehicleSafety ($M = 5$)	9	8	6	7	5	4	3	**1**	2
TrussDesign ($M = 1$)	9	6	7	8	4	5	3	2	**1**
TrussDesign ($M = 2$)	9	7	6	8	4	5	3	2	**1**
TrussDesign ($M = 3$)	9	7	6	8	5	4	3	2	**1**
TrussDesign ($M = 4$)	9	7	5	8	6	4	3	2	**1**
TrussDesign ($M = 5$)	9	7	5	8	6	4	3	2	**1**
Penicillin ($M = 1$)	9	2	7	5	4	8	3	6	**1**
Penicillin ($M = 2$)	9	4	7	6	5	8	3	2	**1**
Penicillin ($M = 3$)	9	3	7	6	5	8	2	4	**1**
Penicillin ($M = 4$)	9	4	8	6	5	7	3	**1**	2
Penicillin ($M = 5$)	9	6	8	7	4	5	3	2	**1**
SnAr ($M = 1$)	9	7	3	8	6	4	5	**1**	2
SnAr ($M = 2$)	9	8	5	7	6	2	4	3	**1**
SnAr ($M = 3$)	9	8	5	6	7	2	4	3	**1**
SnAr ($M = 4$)	9	8	5	7	6	2	4	3	**1**
SnAr ($M = 5$)	9	8	6	7	5	4	2	3	**1**

A.3.5 Case Study: Effects of Dimension, Objective Count, and Cardinality

In this section, we conduct a case study on DTLZ2 to examine how the performance of MOBO-CAPS algorithms varies with input dimension, number of objectives, and target cardinality. In Figure 6, we evaluate DTLZ2 with input dimensions $d = 3, 6, 10$ for the bi-objective case, and with $K = 2, 3, 4$ objectives for a fixed input dimension $d = 6$. The relative ranking among algorithms remains broadly consistent across these settings.

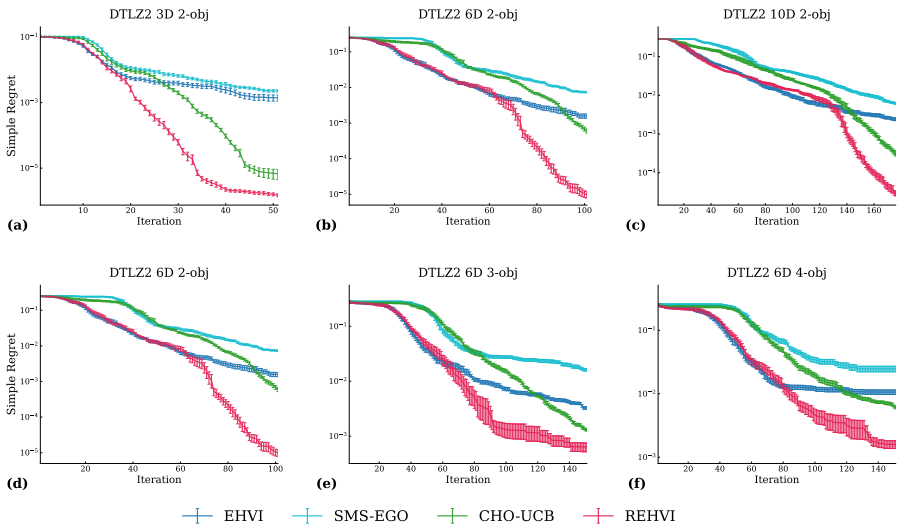

Figure 6: Results on DTLZ2 across different input dimensionalities and different numbers of objectives. The relative ranking among algorithms remains broadly consistent across settings.

We also evaluate two larger cardinalities, $M = 10$ and $M = 15$. The results show that the relative advantage of REHVI and CHO-UCB over baseline acquisition functions decreases as M increases.

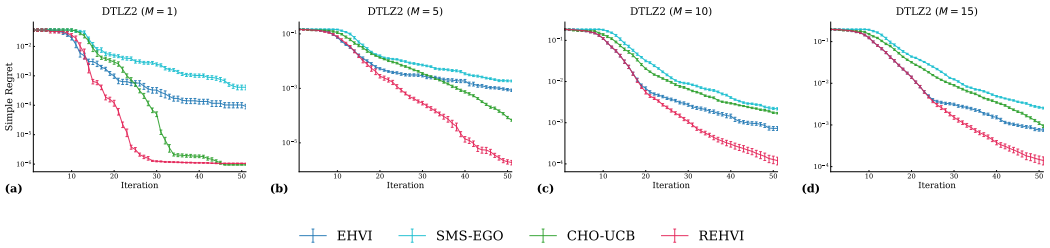

Figure 7: Results on DTLZ2 for larger target cardinalities M. The relative advantage of REHVI and CHO-UCB over baseline acquisition functions decreases as M increases.

A.4 Implementation and Complexity Analysis

We implement REHVI in `BoTorch` [Balandat et al., 2020]. Specifically, for each candidate x, we draw Monte Carlo samples from the GP posterior and evaluate the corresponding sampled objective values. For each sample, we compute the maximum hypervolume gain achievable by a one-point swap with the current CAPS-set. In the implementation, we add the new candidate point to the current size-M set and then remove the point whose deletion maximizes the hypervolume of the remaining M points. We use the best such replacement to compute the replacement-aware hypervolume gain. The REHVI value of x is then estimated by averaging this maximal gain over posterior samples. This implementation is fully differentiable and supports automatic differentiation.

We implement CHO-UCB in `BoTorch` [Balandat et al., 2020]. For each candidate x, we compute the GP posterior mean and standard deviation for each objective, and form an optimistic estimate $\mu(x) + \sqrt{\beta_t}\sigma(x)$, where β_t controls the exploration-exploitation trade-off. We then compute the hypervolume using the corresponding non-dominated partitioning and reference point. In this way, CHO-UCB favors candidates that are predicted to yield large hypervolume under an upper confidence bound criterion. This implementation is fully differentiable and amenable to gradient-based optimization.

The time complexity of our proposed methods can be analyzed by starting with the q-Expected Hypervolume Improvement (qEHVI) algorithm, which has a complexity of $O(KNP(2^q - 1))$

[Daulton et al., 2020]. Here, K is the number of objective functions, N is the number of Monte Carlo samples, P is the number of disjoint hyper-rectangles that result from the decomposition of the non-dominated objective space, and q is the batch size. Furthermore, it is important to note that P is a function of K and the Pareto front size $|\mathcal{P}_t|$. In the two-objective case, $P = |\mathcal{P}_t| + 1$, but P is super-polynomial in $|\mathcal{P}_t|$ for higher K values. We therefore present the complexity analysis for the bi-objective case, where the expression takes a clear form:

- **REHVI:** Our method builds upon EHVI ($q = 1$) with a fixed-size candidate set ($|\mathcal{P}| = M$). The acquisition function must iterate through each of the M points in this set to identify an optimal replacement, which introduces an additional factor of M. This results in a time complexity of $O(KNM^2)$.
- **CHO-UCB**: This method has a batch size $q = M$ and current Pareto front size $|\mathcal{P}| = 0$ but offers a significant computational advantage by eliminating the need for Monte Carlo simulation ($N = 1$). Its time complexity is therefore $O(K(2^M - 1))$. Although the computational cost grows exponentially with M, it remains strictly lower than that of qEHVI ($q = M$), ensuring the method scales effectively for practical values of M.

Figure 8 shows the average wall clock time of acquisition function optimization of SMS-EGO, EHVI, CHO-UCB and REHVI (surrogate model training time not included). Note that for SMS-EGO and EHVI, we calculate $M = 1, 2, 3, 4, 5$ at the same time, so the time for different M is the same. All computations were performed on an Intel Core i7-1355U CPU. For CHO-UCB, and REHVI, we observe that acquisition function optimization time generally increases with M. Among the cardinality-aware acquisitions considered, REHVI exhibits the longest wall clock time, while CHO-UCB requires the least computational time as it does not require Monte Carlo simulation.

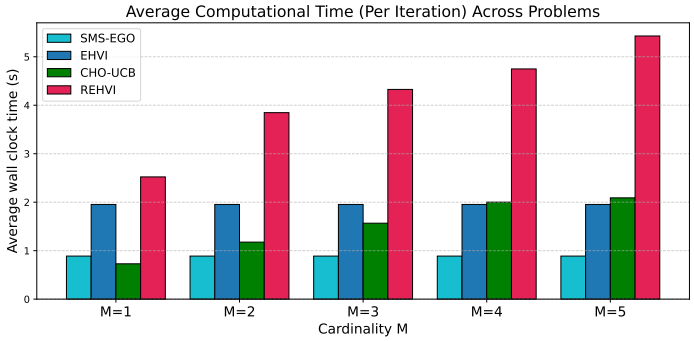

Figure 8: Wall clock time grows with M. CHO-UCB is computationally cheaper than REHVI.

As a side note, acquisition function optimization in this setting is relatively inexpensive. The primary computational burden lies in updating the CAPS-set, specifically, identifying the subset of M solutions from N observations that maximizes hypervolume, given that this subset selection is an NP-hard problem.

A.5 Ablation Study on SPI

We observe that acquisition function optimization, particularly the choice of initialization strategy, has a significant effect on the performance of both REHVI and CHO-EI. In this section, we perform a detailed analysis of the impact of SPI on REHVI, CHO-UCB, and CHO-EI. Notably, the optimization of the acquisition function proves to be more challenging for CHO-EI than for the other two methods, regardless of whether SPI is applied, which likely contributes to its inferior overall performance.

Figure 9 presents an ablation study illustrating that SPI substantially improves the performance of REHVI and CHO-EI, whereas its effect on CHO-UCB is comparatively modest.

To further investigate how SPI influences acquisition function optimization, we track the evolution of acquisition values across iterations with and without SPI on the DTLZ2 test function with $M = 3$. We also analyze the number of L-BFGS iterations per restart as a proxy for optimization stability: a small iteration count indicates premature termination, whereas a larger one reflects a smoother and

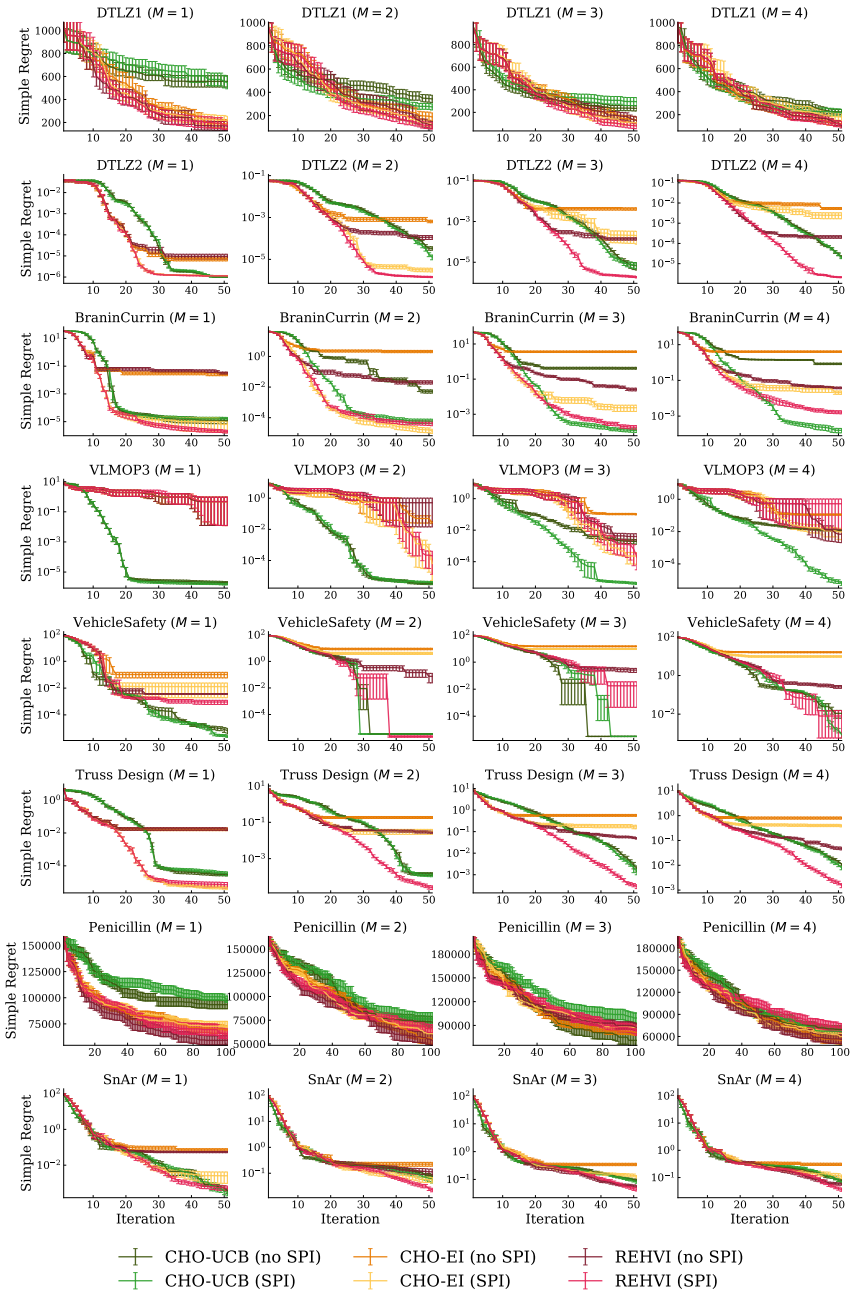

Figure 9: Ablation study of SPI

more stable optimization process. Figure 10 compares the two scenarios, i.e. without SPI and with SPI, demonstrating that SPI yields more stable and reliable acquisition optimization dynamics.

In Figure 10 (a), we observe that, for CHO-EI and REHVI, the iteration count rapidly drops to near zero in the later stages. This decline occurs because the initial acquisition function value becomes zero, causing the optimization to terminate almost immediately returning zero acquisition function value (see Figure 11). In Figure 10 (b), with SPI applied, the iteration numbers become more stable for both REHVI and CHO-EI. However, CHO-EI still experiences a drop in iterations, approaching zero in some cases. This behavior aligns with the percentage of zero acquisition values shown in Figure 12. While REHVI no longer encounters zero initial values with SPI, CHO-EI still exhibits some, albeit at a reduced frequency.

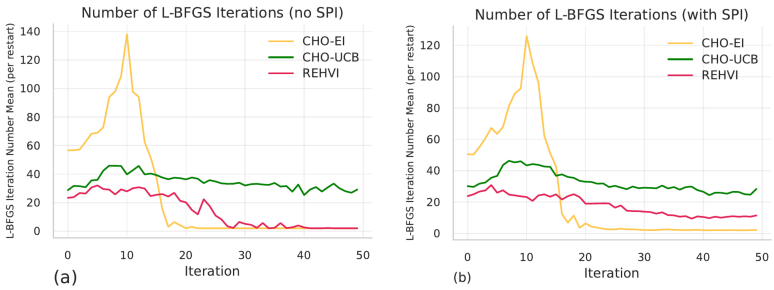

Figure 10: Number of L-BFGS Iterations

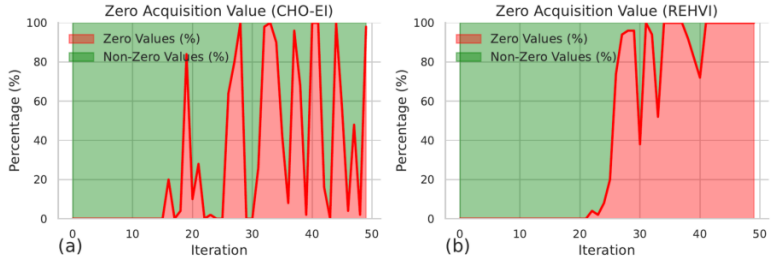

Figure 11: Percentage of Zero Acquisition Values (no SPI): Red indicates the proportion of zero values; green represents non-zero values. CHO-EI and REHVI exhibit a high percentage of zero values, indicating much invalid acquisition optimization.

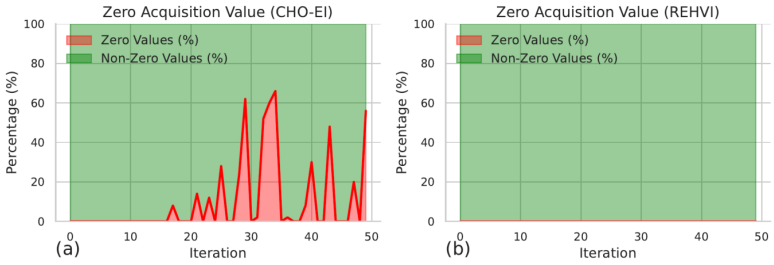

Figure 12: Percentage of Zero Acquisition Values (with SPI): Red indicates the proportion of zero values; green represents non-zero values.

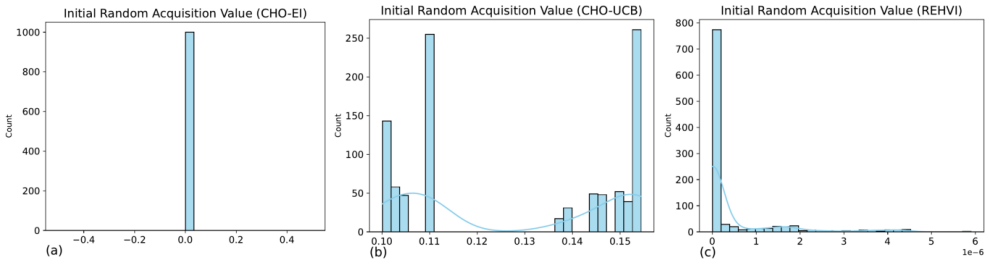

Figure 13: Initial Random Acquisition Value (with SPI)

We find that for both initialization strategies, CHO-EI performs worse than CHO-UCB and REHVI primarily due to challenges in acquisition function optimization. We observe that CHO-EI exhibits a flatter landscape compared to the other two methods. Consequently, it is more difficult to obtain valid initial acquisition function values to start L-BFGS optimization. In Figure 13 (a), we sample 1000 points using CHO-EI at the 40th iteration on SPI. However, all sampled points yield zero acquisition function values. This issue is less severe for CHO-UCB and REHVI. As shown in Figures 13(b) and (c), the 1000 samples exhibit diverse acquisition function values, making the optimization landscape more amenable to gradient-based methods.

In addition, we conduct an ablation study of SPI for MOBO with post-hoc selection (see Figure 14). Overall, SPI does not degrade performance. The only exception is the Penicillin problem, where SPI reduces performance; however, a similar degradation is also observed for REHVI and CHO-UCB (see Figure 9).

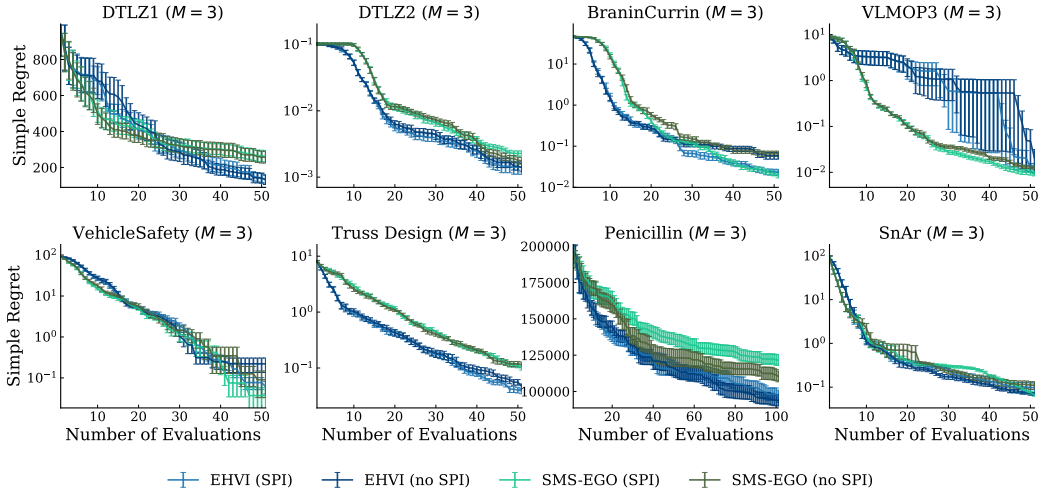

Figure 14: SPI generally does not degrade post-hoc MOBO performance.

A.6 Effect of the Scheduler in CHO-UCB

We investigate the role of the evaluation scheduler in CHO-UCB for the $M = 3$ setting. We compare five variants of CHO-UCB: (i) the batched variant CHO-UCB (BATCH), which evaluates all M points in the proposed set at each iteration; (ii) CHO-UCB (RANDOM), which selects one point uniformly at random from the set; and (iii) CHO-UCB (EHVI), (iv) CHO-UCB (UNCERTAINTY), and (v) CHO-UCB (SMS-EGO), which select a single point using EHVI, variance, and SMS-EGO as their secondary selection criteria, respectively. To match a total budget of 50 and 100 evaluations, we evaluate in batches of size M until the budget is exhausted. In the final iteration, if the remaining evaluation budget is less than M, we randomly select the required number of points from the last batch so that the total number of observations matches the prescribed budget.

Figure 15 shows that CHO-UCB (RANDOM) performs generally worst among the variants, followed by CHO-UCB (BATCH). The remaining three schedulers (EHVI, SMS-EGO, and Uncertainty) perform similarly, indicating that performance is not sensitive to the choice of scheduler. This insensitivity confirms that our algorithm's performance stems fundamentally from the cardinality-aware set construction, rather than from reliance on any specific heuristic.

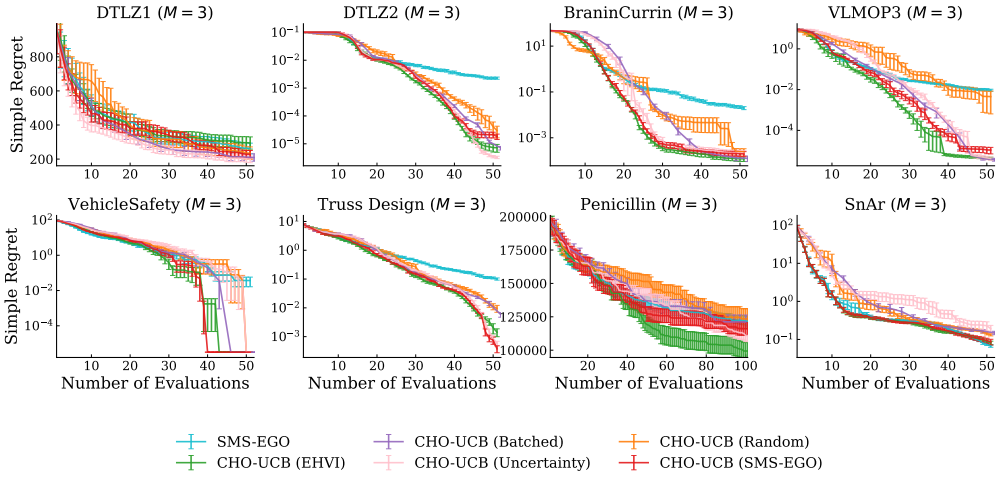

Figure 15: Scheduler ablation for CHO-UCB with $M = 3$. The three schedulers (EHVI, SMS-EGO, and Uncertainty) perform similarly, indicating that performance is not sensitive to the choice of scheduler.

A.7 Study of CHO-EI

In this section, we examine the behavior of CHO-EI. We first identify that its main challenge arises from the optimization of the acquisition function itself. Subsequently, we evaluate the MOBO performance of CHO-EI under several acquisition function optimization strategies, SPI, logEI transformation, and increased numbers of restarts and initial samples. While some techniques provide noticeable improvements, the overall performance of CHO-EI remains inferior to that of REHVI.

A.7.1 Challenges in Optimizing the CHO-EI Acquisition Function

The difficulty in CHO-EI's acquisition optimization stems from its combinatorial nature. Although we formulate this as an $M \times d$-dimensional single-objective problem, it differs fundamentally from traditional single-objective BO. While standard BO employs an $M \times d$-dimensional GP as the surrogate model, our approach is based on K d-dimensional GPs that are joined to $M \times d$ dimensions through the hypervolume operation.

This combinatorial structure significantly impacts the optimization landscape. For example, in the case of $M = 2$, 20 observations, this means $20^2 = 400$ known data points in the $M \times d$ space due to all possible pairwise combinations. This dense combinatorial coverage generally reduces model uncertainty compared to standard $M \times d$-dimensional BO, making it more challenging to identify regions with positive expected improvement values.

To illustrate this challenge, we compare two approaches using a 2-objective optimization problem ($f_1(x) = 0.8 - (x - 0.3)^2 + 0.1 \cdot \sin(10x)$ and $f_2(x) = 0.9 - (x - 0.7)^2 - 0.1 \cdot \cos(8x)$) with 20 observations. In Figure 16, we build separate 1D GP models for each objective and estimate the hypervolume mean and uncertainty through the hypervolume operation (this represents our proposed approach). In Figure 17, we directly model the hypervolume using a standard 2D GP with 20 observations for comparison.

The results clearly demonstrate the challenge in optimizing the acquisition function. Figure 16 shows that our hypervolume-based approach produces low uncertainty and a nearly flat expected improvement landscape with only small regions of positive values with only 20 training data. In contrast, Figure 17 shows that the direct 2D GP approach exhibits higher uncertainty and a smoother, more

navigable expected improvement surface. This comparison highlights why acquisition optimization becomes significantly more difficult in the hypervolume-based setting.

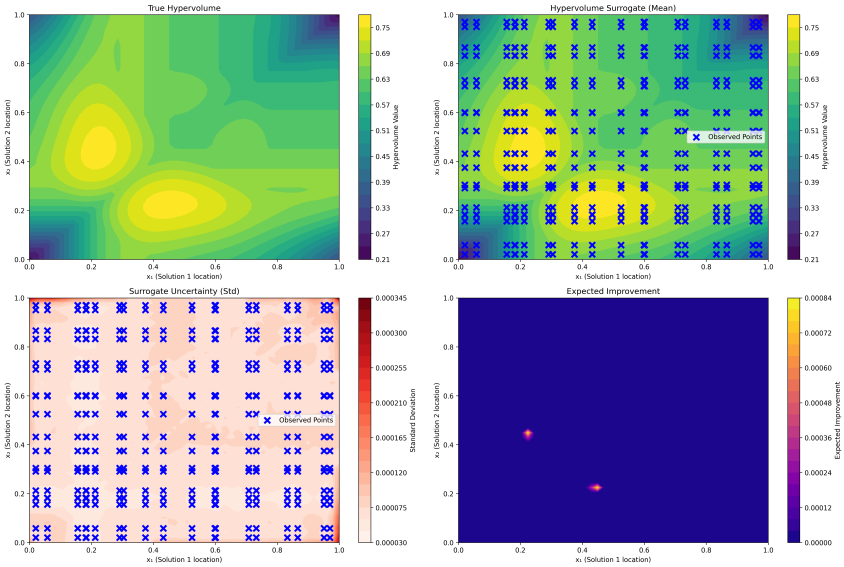

Figure 16: Combinatorial 2D Surrogate Model Using 1D Gaussian Processes and Hypervolume Operation (20 training data)

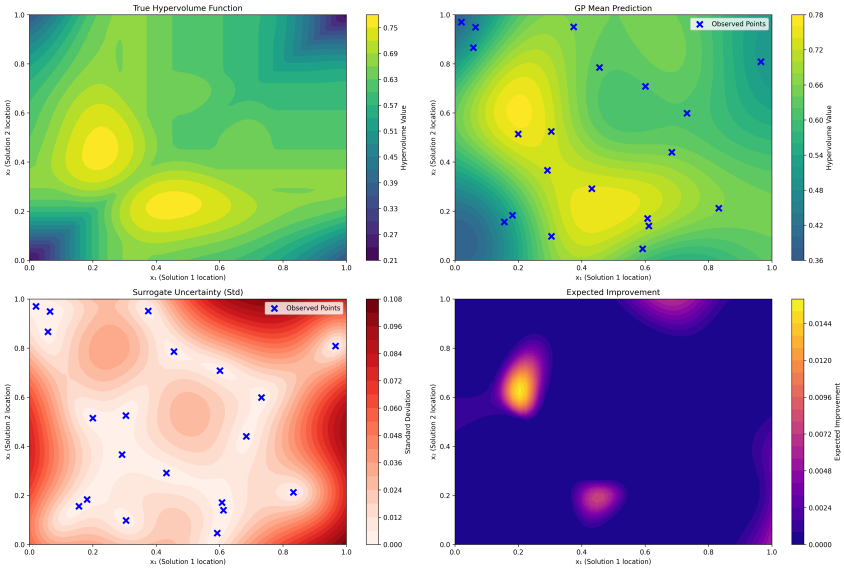

Figure 17: 2D Surrogate Model Directly Modeling Hypervolume Values (20 training data)

A.7.2 SPI

The experimental results with and without SPI are presented in the Figure 9. Although SPI generally improves CHO-EI performance, it still remains inferior to REHVI and CHO-UCB.

A.7.3 CHO-logEI

As discussed earlier, the poor performance of CHO-EI stems from its tendency to exhibit flat regions with zero acquisition values, leading to vanishing gradients. A common strategy to mitigate this issue is logEI transformation [Ahmadianshalchi et al., 2024], which preserves gradient information even

when the EI value is close to zero. Hence, we conducted an additional experiment to evaluate the effect of applying a logarithmic transformation in the CHO-EI Monte Carlo simulation [Ahmadianshalchi et al., 2024], to see whether this trick can solve the issue. Specifically, we use the logmeanexp transformation in `BoTorch` to compute a log-space Monte Carlo estimate of the expected hypervolume improvement.

We evaluate the case $M = 3$, and the results are shown in Figure 18. The logarithmic EI transformation does not significantly improve the overall MOBO performance.

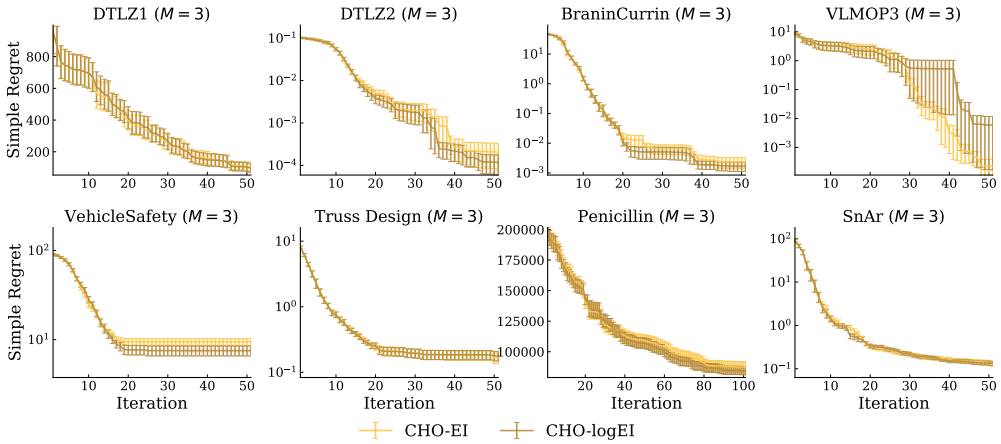

Figure 18: logEI does not improve performance in the MOBO-CAPS problem.

A.8 Different Hyperparameters

In this section, we conduct tests to examine whether increasing the number of restarts or the number of initial samples can improve CHO-EI's performance. Our results indicate that increasing the number of restarts provides minimal improvement, while increasing the number of initial random samples offers modest benefits in certain cases, such as DTLZ2, BraninCurrin, and the Truss Design problem. However, despite these improvements, CHO-EI continues to underperform relative to REHVI.

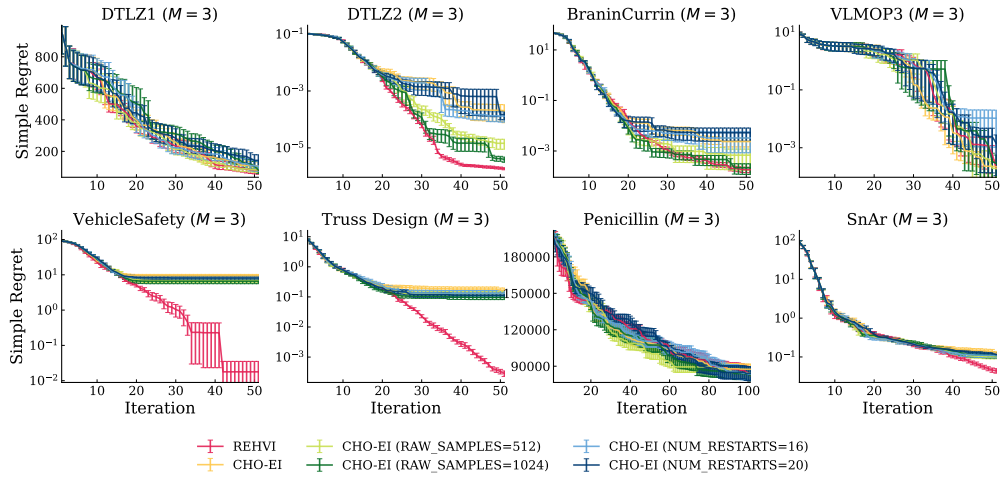

Figure 19: A larger number of initial random samples leads to better MOBO performance; however, REHVI still achieves superior results overall.

B Proof

B.1 Distance Metric on Multisets

Populations as multisets. We model a population of size M as a collection of M objective vectors in $\mathbb{R}^K$, allowing for duplicates. To formalize this while disregarding the order of elements, we define an equivalence relation $\sim$ on the space of ordered tuples $(\mathbb{R}^K)^M$. Two tuples $\mathbf{a} = (a^{(1)}, \ldots, a^{(M)})$ and $\mathbf{b} = (b^{(1)}, \ldots, b^{(M)})$ are considered equivalent, denoted $\mathbf{a} \sim \mathbf{b}$, if one is a permutation of the other. That is, if there exists a permutation $\sigma \in \Pi_M$ such that $b^{(m)} = a^{(\sigma(m))}$ for all $m \in [M]$. The space of multisets is denoted as $\mathcal{M}_M(\mathbb{R}^K)$.

In this framework, a multiset $X \in \mathcal{M}_M(\mathbb{R}^K)$ represents an equivalence class of tuples that differ only by ordering. Any metric on $\mathcal{M}_M(\mathbb{R}^K)$ must therefore be invariant to the representative sequence chosen.

Definition B.1 (Matching Distance)**.** Let $A, B \in \mathcal{M}_M(\mathbb{R}^K)$ be two multisets of size M. We define the matching distance $d_{\mathcal{M}}(A, B)$ as the minimum maximal deviation between their elements under any possible bijective pairing. Formally, choosing arbitrary representative sequences $(a^{(1)}, \ldots, a^{(M)})$ for A and $(b^{(1)}, \ldots, b^{(M)})$ for B, we define:

$$d_{\mathcal{M}}(A, B) := \min_{\pi \in \Pi_M} \left(\max_{m \in [M]} \left\| a^{(m)} - b^{(\pi(m))} \right\|_\infty \right), \tag{1}$$

where Π_M denotes the set of all permutations of $\{1, \ldots, M\}$.

Proposition B.2 (Metric property of matching distance on multisets)**.** *The function $d_{\mathcal{M}}$ is well-defined on $\mathcal{M}_M(\mathbb{R}^K)$ and is a metric. In particular, for all $A, B, C \in \mathcal{M}_M(\mathbb{R}^K)$:*

1. ***Nonnegativity:*** $d_{\mathcal{M}}(A, B) \geq 0$.
2. ***Identity:*** $d_{\mathcal{M}}(A, B) = 0 \iff A = B$ *as multisets* .
3. ***Symmetry:*** $d_{\mathcal{M}}(A, B) = d_{\mathcal{M}}(B, A)$.
4. ***Triangle inequality:*** $d_{\mathcal{M}}(A, C) \leq d_{\mathcal{M}}(A, B) + d_{\mathcal{M}}(B, C)$.

Proof. **Well-definedness.** The value of $d_{\mathcal{M}}(X, Y)$ is independent of the specific ordered sequences used to represent the multisets X and Y. This is immediate from the definition: since the distance is calculated by minimizing over *all possible permutations* $\pi \in \Pi_M$, any initial reordering of the underlying vectors in X or Y does not influence the result.

Nonnegativity. For any π, $\max_m \|a^{(m)} - b^{(\pi(m))}\|_\infty \geq 0$, hence taking the minimum or maximum preserves nonnegativity.

Identity. If $A = B$ as multisets, then there exists a permutation π such that $a^{(m)} = b^{(\pi(m))}$ for all m. For that π, the maximum norm is 0, hence $d_{\mathcal{M}}(A, B) = 0$. Conversely, if $d_{\mathcal{M}}(A, B) = 0$, then for some π, $\max_m \|a^{(m)} - b^{(\pi(m))}\|_\infty = 0$, so $\|a^{(m)} - b^{(\pi(m))}\|_\infty = 0$ for all m, i.e., $a^{(m)} = b^{(\pi(m))}$ for all m. Thus $A = B$ as multisets.

Symmetry. This follows immediately from the symmetry of the underlying norm and the bijection. The expression $\min_\pi \max_m \|a^{(m)} - b^{(\pi(m))}\|_\infty$ essentially minimizes the maximum distance over all possible one-to-one pairings between the elements of X and Y since the pairing (x, y) has the same distance as (y, x) (i.e., $\|x - y\| = \|y - x\|$), and the set of all possible pairings is identical.

Triangle Inequality: Let A, B, C be three multisets. Let π_{AB} and π_{BC} be the optimal permutations achieving the distances $d_{\mathcal{M}}(A, B)$ and $d_{\mathcal{M}}(B, C)$ respectively. We construct a candidate permutation $\pi_{AC} = \pi_{BC} \circ \pi_{AB}$ to match A to C. For any index m, the triangle inequality of the vector norm implies:

$$\|a^{(m)} - c^{(\pi_{AC}(m))}\|_\infty \leq \|a^{(m)} - b^{(\pi_{AB}(m))}\|_\infty + \|b^{(\pi_{AB}(m))} - c^{(\pi_{BC}(\pi_{AB}(m)))}\|_\infty.$$

Taking the maximum over all $m \in [M]$ on both sides:

$$\begin{aligned}\max_m \|a^{(m)} - c^{(\pi_{AC}(m))}\|_\infty &\le \max_m \|a^{(m)} - b^{(\pi_{AB}(m))}\|_\infty + \max_m \|b^{(\pi_{AB}(m))} - c^{(\pi_{BC}(\pi_{AB}(m)))}\|_\infty \\ &= d_{\mathcal{M}}(A,B) + d_{\mathcal{M}}(B,C).\end{aligned}$$

Finally, since $d_{\mathcal{M}}(A,C)$ minimizes over *all* permutations, we have:

$$d_{\mathcal{M}}(A,C) \le \max_m \|a^{(m)} - c^{(\pi_{AC}(m))}\|_\infty \le d_{\mathcal{M}}(A,B) + d_{\mathcal{M}}(B,C).$$

□

B.2 Lipschitz Continuity of Hypervolume

Definition B.3 (Clamped Hypervolume)**.** Given a reference point $z_{\mathrm{ref}} \in \mathbb{R}^K$, the hypervolume of a multiset $A \in \mathcal{M}_M(\mathbb{R}^K)$ is defined as the measure of the union of hyper-rectangles dominated by the elements of A and bounded below by z_{ref}. To handle points that do not strictly dominate the reference point, we apply a coordinate-wise projection:

$$\mathrm{HV}(A) := \lambda^K \left(\bigcup_{a \in A} \prod_{j=1}^{K} \left[z_{\mathrm{ref},j}, \max(a_j, z_{\mathrm{ref},j}) \right] \right). \tag{2}$$

Under this definition, any coordinate $a_j < z_{\mathrm{ref},j}$ contributes zero length to the volume integration in dimension j (effectively projecting the point onto the reference boundary).

Assumption B.4 (Bounded objectives)**.** We fix a reference point $z_{\mathrm{ref}} \in \mathbb{R}^K$ and an upper bound vector $u \in \mathbb{R}^K$ satisfying strict dominance $u \succ z_{\mathrm{ref}}$. We assume that the true objective functions are bounded from above by u:

$$f(x) \preceq u \quad \text{for all } x \in \mathcal{X}.$$

We define the maximum domain width $D_{\max} := \|u - z_{\mathrm{ref}}\|_\infty$. The Lipschitz constant for the hypervolume indicator is derived using this width as $L_{\mathrm{HV}} = MKD_{\max}^{K-1}$.

Lemma B.5 (Hypervolume Lipschitzness w.r.t. matching distance)**.** *Under Assumption B.4, let $A = \{a^{(1)}, \dots, a^{(M)}\}$ and $B = \{b^{(1)}, \dots, b^{(M)}\}$ be two multisets. Define*

$$\mathit{HV}(A) := \lambda^K \left(\bigcup_{a \in A} \prod_{j=1}^{K} [z_{\mathrm{ref},j}, \max(a_j, z_{\mathrm{ref},j})] \right) \qquad d_{\mathcal{M}}(A,B) := \min_{\pi \in \Pi_M} \max_{m \in [M]} \|a^{(m)} - b^{(\pi(m))}\|_\infty.$$

Then

$$|\mathit{HV}(A) - \mathit{HV}(B)| \le MKD_{\max}^{K-1}\, d_{\mathcal{M}}(A,B).$$

Proof. Let $\pi^\star \in \Pi_M$ attain the minimum in $d_{\mathcal{M}}(A,B)$ and reindex B so that $\tilde{b}^{(m)} := b^{(\pi^\star(m))}$. Then for all m,

$$\|a^{(m)} - \tilde{b}^{(m)}\|_\infty \le d_{\mathcal{M}}(A,B),$$

and $\mathrm{HV}(B) = \mathrm{HV}(\{\tilde{b}^{(1)}, \dots, \tilde{b}^{(M)}\})$ by permutation invariance.

Case 1: All points dominate the reference point. Assume first that $a^{(m)} \succeq z_{\mathrm{ref}}$ and $\tilde{b}^{(m)} \succeq z_{\mathrm{ref}}$. Define intermediate multisets $Z^{(0)}, \dots, Z^{(M)}$ by $Z^{(0)} = A$ and $Z^{(m)}$ is obtained from $Z^{(m-1)}$ by replacing $a^{(m)}$ with $\tilde{b}^{(m)}$. By the triangle inequality on $\mathbb{R}$,

$$|\mathrm{HV}(A) - \mathrm{HV}(B)| \le \sum_{m=1}^{M} |\mathrm{HV}(Z^{(m-1)}) - \mathrm{HV}(Z^{(m)})|.$$

Let $\{z^{(1)}, \dots, z^{(M-1)}\}$ denote the $M-1$ points common to $Z^{(m-1)}$ and $Z^{(m)}$. Let $U := \bigcup_{\ell=1}^{M-1} \prod_{j=1}^{K} [z_{\mathrm{ref},j}, z_j^{(\ell)}]$ be the union of the other $M-1$ boxes, and write $R := \prod_{j=1}^{K} [z_{\mathrm{ref},j}, a_j^{(m)}]$, $\tilde{R} := \prod_{j=1}^{K} [z_{\mathrm{ref},j}, \tilde{b}_j^{(m)}]$. Since $\mathrm{HV}(Z^{(m-1)}) = \lambda^K(U \cup R)$ and $\mathrm{HV}(Z^{(m)}) = \lambda^K(U \cup \tilde{R})$, we have

$$|\mathrm{HV}(Z^{(m-1)}) - \mathrm{HV}(Z^{(m)})| \le \lambda^K\big((U \cup R) \Delta (U \cup \tilde{R})\big).$$

Moreover, $(U \cup R)\Delta(U \cup \tilde{R}) \subseteq R\Delta\tilde{R}$. Consider a point $z \in R\Delta\tilde{R}$. This point belongs to exactly one of the two boxes. Without loss of generality, assume $z \in R$ but $z \notin \tilde{R}$. Since $z \in R$, we know that for all dimensions k, $z_k \leq a_k^{(m)}$. On the other hand, since $z \notin \tilde{R}$, there must be at least one dimension j where the box constraint is violated, meaning $z_j > \tilde{b}_j^{(m)}$. Combining these inequalities for that specific dimension j, the coordinate z_j must lie in the interval $(\tilde{b}_j^{(m)}, a_j^{(m)}]$. Consequently, z is contained within the slab S_j defined by the mismatch in the j-th coordinate:

$$S_j := \Big\{ z \in \mathbb{R}^K : \; z_j \in [\min(a_j^{(m)}, \tilde{b}_j^{(m)}), \max(a_j^{(m)}, \tilde{b}_j^{(m)})], \; z_\ell \in [z_{\mathrm{ref},\ell}, u_\ell] \; \forall \ell \neq j \Big\}.$$

The same logic applies if $z \in \tilde{R} \setminus R$. Therefore,

$$R\Delta\tilde{R} \subseteq \bigcup_{j=1}^{K} S_j.$$

By subadditivity and Assumption B.4,

$$\lambda^K(R\Delta\tilde{R}) \leq \sum_{j=1}^{K} \lambda^K(S_j) \leq \sum_{j=1}^{K} |a_j^{(m)} - \tilde{b}_j^{(m)}| \, D_{\max}^{K-1} \leq K D_{\max}^{K-1} \|a^{(m)} - \tilde{b}^{(m)}\|_\infty.$$

Combining the above and summing over $m = 1, \ldots, M$ yields the result for Case 1.

Case 2: General configuration. Consider the general case where elements of A or B may not dominate the reference point z_{ref} in all coordinates (i.e., $a^{(m)} \not\succeq z_{\mathrm{ref}}$). We define the projection operator $\mathrm{Proj}_{z_{\mathrm{ref}}} : \mathbb{R}^K \to \mathbb{R}^K$ component-wise by:

$$\big(\mathrm{Proj}_{z_{\mathrm{ref}}}(x)\big)_j = \max(x_j, z_{\mathrm{ref},j}), \quad \forall j \in \{1, \ldots, K\}.$$

By definition, the hypervolume indicator measures the union of rectangular regions lower-bounded by z_{ref}. Consequently, the contribution of any point x is identical to the contribution of its projection. Thus, defining the projected multisets $A' = \{\mathrm{Proj}_{z_{\mathrm{ref}}}(a) : a \in A\}$ and $B' = \{\mathrm{Proj}_{z_{\mathrm{ref}}}(b) : b \in B\}$, we have the identity $\mathrm{HV}(A) = \mathrm{HV}(A')$ and $\mathrm{HV}(B) = \mathrm{HV}(B')$.

Since the projected points satisfy $a' \succeq z_{\mathrm{ref}}$ by construction, the condition of Case 1 is met. Therefore, the Lipschitz bound holds for the projected sets:

$$|\mathrm{HV}(A) - \mathrm{HV}(B)| = |\mathrm{HV}(A') - \mathrm{HV}(B')| \leq MKD_{\max}^{K-1} \, d_{\mathcal{M}}(A', B').$$

The operator $\mathrm{Proj}_{z_{\mathrm{ref}}}$ is non-expansive with respect to the ℓ_∞-norm, since $|\max(u, c) - \max(v, c)| \leq |u - v|$ for any scalars u, v, c. This implies that the matching distance between the projected populations is no larger than the original distance:

$$d_{\mathcal{M}}(A', B') \leq d_{\mathcal{M}}(A, B).$$

Substituting this back into the inequality yields the final result:

$$|\mathrm{HV}(A) - \mathrm{HV}(B)| \leq MKD_{\max}^{K-1} \, d_{\mathcal{M}}(A, B).$$

□

B.3 Noise-free Simple Regret of CHO-UCB

We consider the noise-free setting where, at each iteration $t = 1, \ldots, T$, CHO-UCB selects a population X_t of M points, modeled as a multiset. The proof uses techniques from [Iwazaki, 2025, Srinivas et al., 2009].

Setting and notation. Let $\mathcal{X} \subset \mathbb{R}^d$ be compact. We have K objectives $f_i : \mathcal{X} \to \mathbb{R}$, $i \in [K]$, and write $\mathbf{f}(x) := (f_1(x), \ldots, f_K(x)) \in \mathbb{R}^K$. In iteration t, the algorithm selects a multiset

$$X_t = \{x_{t,1}, \ldots, x_{t,M}\} \in \mathcal{M}_M(\mathcal{X}),$$

and we observe noiselessly $f(X_t)$.

We compute hypervolume with respect to a fixed reference point $z_{\mathrm{ref}} \in \mathbb{R}^K$ (maximization convention), and write $\mathrm{HV}(\cdot)$ as shorthand for $\mathrm{HV}_{z_{\mathrm{ref}}}(\cdot)$. We define the optimal CAPS-set of size M as

$$X^\star \in \operatorname*{argmax}_{X \in \mathcal{M}_M(\mathcal{X})} \mathrm{HV}\big(\mathbf{f}(X)\big).$$

Define the per-iteration regret and simple regret as

$$r_t := \mathrm{HV}\big(\mathbf{f}(X^\star)\big) - \mathrm{HV}\big(\mathbf{f}(X_t)\big), \qquad r_T^{\mathrm{simple}} := \min_{1 \le t \le T} r_t.$$

Assumption B.6 (RKHS regularity)**.** For each $i \in [K]$, f_i lies in the RKHS $\mathcal{H}_{k_i}$ of a known positive definite kernel $k_i : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$. Moreover, $k_i(x, x) \le 1$ for all $x \in \mathcal{X}$ and $\|f_i\|_{\mathcal{H}_{k_i}} \le B$ for some $B > 0$.

Lemma B.7 (Objective range under RKHS regularity)**.** *Assume Assumption B.6 and kernel normalization $k_i(x, x) \le 1$ for all $x \in \mathcal{X}$ and $i = 1, \dots, K$. In the noise-free setting with $\sqrt{\beta_t} = B$, for every $t \ge 1$, every $x \in \mathcal{X}$, and every objective $i \in \{1, \dots, K\}$,*

$$f_i(x) \in [-B, B], \qquad \mu_{t-1,i}(x) \in [-B, B], \qquad \sigma_{t-1,i}(x) \le 1,$$

and consequently

$$\mathrm{UCB}_{t,i}(x),\ \mathrm{LCB}_{t,i}(x) \in [-2B, 2B]. \tag{3}$$

Proof. By the reproducing property, for any $g \in \mathcal{H}_{k_i}$, $|g(x)| \le \|g\|_{\mathcal{H}_{k_i}} \sqrt{k_i(x, x)} \le \|g\|_{\mathcal{H}_{k_i}}$. Thus $|f_i(x)| \le \|f_i\|_{\mathcal{H}_{k_i}} \le B$. In the noise-free setting, the posterior mean $\mu_{t-1,i}$ is the minimum-norm interpolant (orthogonal projection) of f_i onto $\mathrm{span}\{k_i(x_s, \cdot)\}_{s \le t-1}$, hence $\|\mu_{t-1,i}\|_{\mathcal{H}_{k_i}} \le \|f_i\|_{\mathcal{H}_{k_i}} \le B$, which implies $|\mu_{t-1,i}(x)| \le B$. Moreover, $\sigma^2_{t-1,i}(x) = k_i(x, x) - k^\top K^{-1} k \le k_i(x, x) \le 1$, so $\sigma_{t-1,i}(x) \le 1$. With $\sqrt{\beta_t} = B$, we have $\mathrm{UCB}_{t,i}(x) = \mu_{t-1,i}(x) + B\sigma_{t-1,i}(x) \le B + B = 2B$ and $\mathrm{LCB}_{t,i}(x) = \mu_{t-1,i}(x) - B\sigma_{t-1,i}(x) \ge -B - B = -2B$, giving (3). □

GP posterior and UCB. We place independent zero-mean GP priors on $(f_i)_{i=1}^K$ with kernel k_i. Let $\mu_{t-1,i}$ and $\sigma_{t-1,i}$ denote the posterior mean and standard deviation of f_i after $(t-1)$ iterations. In this noise-free RKHS setting, we use a constant exploration parameter:

$$\sqrt{\beta_t} := B \quad \text{for all } t \ge 1. \tag{4}$$

Algorithm (CHO-UCB). In iteration $t = 1, \dots, T$:

1. Using the posteriors $(\mu_{t-1,i}, \sigma_{t-1,i})_{i=1}^K$, define

$$UCB_t(x) := \mu_{t-1}(x) + B\sigma_{t-1}(x), \qquad LCB_t(x) := \mu_{t-1}(x) - B\sigma_{t-1}(x).$$

2. Select a multiset of M points by maximizing the hypervolume of the UCB:

$$X_t \in \operatorname*{argmax}_{X \in \mathcal{M}_M(\mathcal{X})} \mathrm{HV}\big(UCB_t(X)\big).$$

3. Evaluate $\mathbf{f}(x)$ for all $x \in X_t$ and update all K GP posteriors.

Lemma B.8 (Confidence bound)**.** *Under Assumption B.6, for all $i \in [K]$, $t \ge 1$ and $x \in \mathcal{X}$,*

$$|f_i(x) - \mu_{t-1,i}(x)| \ \le\ B\,\sigma_{t-1,i}(x). \tag{5}$$

Proof. This is the standard RKHS inequality for noise-free GP regression; see Lemma 7 in Iwazaki [2025]. □

Lemma B.9 (Instantaneous regret bound)**.** *Under Assumption B.6, the batched CHO-UCB algorithm satisfies:*

$$r_t \ \le\ 2BL_{HV} \sum_{i=1}^{K} \max_{m \in [M]} \sigma_{t-1,i}(x_{t,m}), \tag{6}$$

where $L_{HV} = MK(2B - \min_j z_{\mathrm{ref},j})^{K-1}$ is the Lipschitz constant adapted to the RKHS bounds.

Proof. By Lemma B.8, we have for all $x \in \mathcal{X}$ that $LCB_t(x) \preceq \mathbf{f}(x) \preceq UCB_t(x)$, and thus for any population X, $LCB_t(X) \preceq \mathbf{f}(X) \preceq UCB_t(X)$ componentwise. Since $\mathrm{HV}(\cdot)$ is monotone w.r.t. componentwise dominance, this implies

$$\mathrm{HV}(\mathbf{f}(X^\star)) \le \mathrm{HV}(UCB_t(X^\star)) \quad \text{and} \quad \mathrm{HV}(LCB_t(X_t)) \le \mathrm{HV}(\mathbf{f}(X_t)).$$

Moreover, because $X_t \in \arg\max_{X \in \mathcal{X}^M} \mathrm{HV}(UCB_t(X))$, we have $\mathrm{HV}(UCB_t(X^\star)) \le \mathrm{HV}(UCB_t(X_t))$. Combining these inequalities yields

$$r_t = \mathrm{HV}(\mathbf{f}(X^\star)) - \mathrm{HV}(\mathbf{f}(X_t)) \le \mathrm{HV}(UCB_t(X_t)) - \mathrm{HV}(LCB_t(X_t)).$$

From Lemma B.7, we know that for all t and x, every component of $UCB_t(x)$ and $LCB_t(x)$ lies in $[-2B, 2B]$. Therefore, these sets are contained within the hyper-rectangle bounded above by the vector $\mathbf{u}_{\text{alg}} = (2B, \dots, 2B) \in \mathbb{R}^K$. Assuming $2B > z_{\mathrm{ref},j}$ for all j, the effective domain width for the Lipschitz constant is:

$$D_{\max} := \max_{j \in [K]} (2B - z_{\mathrm{ref},j}).$$

We apply Lemma B.5 using the constant $L_{\mathrm{HV}} = MKD_{\max}^{K-1}$:

$$r_t \le L_{\mathrm{HV}}\, d_{\mathcal{M}}\big(UCB_t(X_t), LCB_t(X_t)\big).$$

Since $UCB_t(X_t)$ and $LCB_t(X_t)$ are computed point-wise on the *same* multiset X_t, the optimal pairing is the identity. Thus, the matching distance is bounded by the maximal element-wise difference:

$$d_{\mathcal{M}}\big(UCB_t(X_t), LCB_t(X_t)\big) \le \max_{x \in X_t} \|UCB_t(x) - LCB_t(x)\|_\infty.$$

By definition, $\|UCB_t(x) - LCB_t(x)\|_\infty = 2B \max_i \sigma_{t-1,i}(x)$. Therefore:

$$r_t \le 2BL_{\mathrm{HV}} \max_{m \in [M]} \max_{i \in [K]} \sigma_{t-1,i}(x_{t,m}).$$

Using $\max_i a_i \le \sum_{i=1}^K a_i$, we obtain the final bound. □

Lemma B.10 (Batched maximum posterior standard deviation bound)**.** *Fix an objective $i \in [K]$. Let $X_t = \{x_{t,1}, \dots, x_{t,M}\}$ be any sequence of batches, and let $\sigma_{t-1,i}$ denote the posterior standard deviation after conditioning on all observations in the previous batches $X_1, \dots, X_{t-1}$. Then*

$$\sum_{t=1}^{T} \max_{m \in [M]} \sigma_{t-1,i}(x_{t,m}) \le C_i(T, d, \theta_i),$$

where $C_i(T, d, \theta_i)$ is the cumulative posterior standard-deviation bound in Lemma B.11.

Proof. For each t, choose

$$x^\star_{t,i} \in \arg\max_{m \in [M]} \sigma_{t-1,i}(x_{t,m}).$$

Then

$$\sum_{t=1}^{T} \max_{m \in [M]} \sigma_{t-1,i}(x_{t,m}) = \sum_{t=1}^{T} \sigma_{t-1,i}(x^\star_{t,i}).$$

Now define $\widetilde{\sigma}_{t-1,i}$ as the posterior standard deviation obtained by conditioning only on the reduced sequential history

$$\{x^\star_{1,i}, \dots, x^\star_{t-1,i}\}.$$

The actual posterior standard deviation $\sigma_{t-1,i}$ is conditioned on the full previous batched history

$$\bigcup_{\tau=1}^{t-1} X_\tau,$$

which contains the reduced history as a subset. Since posterior variance in a noise-free Gaussian process cannot increase when conditioning on additional observations, we have

$$\sigma_{t-1,i}(x^\star_{t,i}) \le \widetilde{\sigma}_{t-1,i}(x^\star_{t,i}).$$

Therefore,

$$\sum_{t=1}^{T} \sigma_{t-1,i}(x_{t,i}^{\star}) \leq \sum_{t=1}^{T} \widetilde{\sigma}_{t-1,i}(x_{t,i}^{\star}).$$

Applying Lemma 3 of Iwazaki [2025] to the sequentially updated sequence $\{x_{t,i}^{\star}\}_{t=1}^{T}$ gives

$$\sum_{t=1}^{T} \widetilde{\sigma}_{t-1,i}(x_{t,i}^{\star}) \leq C_i(T, d, \theta_i).$$

Combining the inequalities yields the result. □

Lemma B.11 (cumulative sum of posterior standard deviations for SE and Matérn kernel)**.** *Fix any $L > 0$, input domain $\mathcal{X} \subset [-L, L]^d$, and kernel function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ that satisfies $k(x, x) \leq 1$ for all $x \in \mathcal{X}$. Furthermore, let $C_{\mathrm{SE}}, C_{\mathrm{Mat}}, \underline{\lambda}_{\mathrm{SE}}, \underline{\lambda}_{\mathrm{Mat}} > 0, \underline{T}_{\mathrm{SE}}, \underline{T}_{\mathrm{Mat}} \geq 2$ be constants such that:*

$$\forall \lambda \in (0, \underline{\lambda}_{\mathrm{SE}}], \forall t \geq \underline{T}_{\mathrm{SE}}, \quad \gamma_t(\lambda^2) \leq C_{\mathrm{SE}} \left(\ln \frac{t}{\lambda^2} \right)^{d+1},$$

$$\forall \lambda \in (0, \underline{\lambda}_{\mathrm{Mat}}], \forall t \geq \underline{T}_{\mathrm{Mat}}, \quad \gamma_t(\lambda^2) \leq C_{\mathrm{Mat}} \left(\frac{t}{\lambda^2} \right)^{\frac{d}{2\nu+d}} \left(\ln \frac{t}{\lambda^2} \right)^{\frac{2\nu}{2\nu+d}}$$

for $k = k_{\mathrm{SE}}$ and $k = k_{\mathrm{Matérn}}$, respectively. Then, the following statements hold for any $T \in \mathbb{N}_+$ and any input sequence $x_1, \ldots, x_T \in \mathcal{X}$.

For** $k = k_{\mathrm{SE}}$**:

$$\sum_{t=1}^{T} \sigma(x_t; \mathbf{X}_{t-1}) \leq \overline{T}_{\mathrm{SE}} + (d+1) \left(\frac{\overline{C}_{\mathrm{SE}}}{2} \right)^{-\frac{3d+3}{2}} \Gamma\left(\frac{3d+3}{2} \right),$$

where $\overline{C}_{\mathrm{SE}} = (6C_{\mathrm{SE}})^{-\frac{1}{d+1}}$ and $\overline{T}_{\mathrm{SE}} = \max\{\underline{T}_{\mathrm{SE}}, \underline{T}_{\mathrm{SE}}^{(A)}, \lfloor (d+1)^{d+1} / \overline{C}_{\mathrm{SE}}^{d+1} \rfloor + 1\}$ with $\underline{T}_{\mathrm{SE}}^{(A)} = \min\{T \in \mathbb{N}_+ \mid \forall t \geq T,\ t \exp(-\overline{C}_{\mathrm{SE}} t^{1/(d+1)}) \leq \underline{\lambda}_{\mathrm{SE}}^2\}$.

For** $k = k_{\mathrm{Matérn}}$ **with** $\nu > 1/2$**:

$$\sum_{t=1}^{T} \sigma(x_t; \mathbf{X}_{t-1}) \leq \begin{cases} \overline{T}_{\mathrm{Mat}} + \overline{C}_{\mathrm{Mat}}^{1/2} \dfrac{d}{d-\nu} T^{\frac{d-\nu}{d}} (\ln T)^{\frac{2\nu}{d}} & \text{if } d > \nu, \\ \overline{T}_{\mathrm{Mat}} + \overline{C}_{\mathrm{Mat}}^{1/2} (\ln T)^2 & \text{if } d = \nu, \\ \overline{T}_{\mathrm{Mat}} + \overline{C}_{\mathrm{Mat}}^{1/2} \dfrac{\Gamma\left(\frac{2\nu}{d} + 1\right)}{\left(\frac{2\nu}{d} - 1\right)^{\frac{2\nu}{d}+1}} & \text{if } d < \nu, \end{cases}$$

where $\overline{C}_{\mathrm{Mat}} = \max\left\{1, \left(2 + \frac{2\nu}{d}\right)^{\frac{2\nu}{d}} (6C_{\mathrm{Mat}})^{1+\frac{2\nu}{d}}\right\}$ and $\overline{T}_{\mathrm{Mat}} = \max\{4, \underline{T}_{\mathrm{Mat}}, \underline{T}_{\mathrm{Mat}}^{(A)}\}$ with $\underline{T}_{\mathrm{Mat}}^{(A)} = \min\{T \in \mathbb{N}_+ \mid \forall t \geq T,\ \overline{C}_{\mathrm{Mat}} t^{-\frac{2\nu}{d}} (\ln t)^{\frac{2\nu}{d}} \leq \underline{\lambda}_{\mathrm{Mat}}^2\}$.

Theorem B.12 (Noise-free Simple Regret of Batched CHO-UCB)**.** *Suppose the objective functions f_i satisfy the RKHS Assumption 4.2 with bound B, and the problem domain is bounded as per Assumption 4.1 with $u = (2B, \ldots, 2B)$. Setting the exploration parameter $\sqrt{\beta_t} = B$ for all t, the simple regret of Batched CHO-UCB satisfies:*

$$r_T^{\mathrm{simple}} \leq \frac{2BMK(2B - \min_j z_{\mathrm{ref},j})^{K-1}}{T} \sum_{i=1}^{K} C_i(T, d, \theta_i),$$

where $C_i(T, d, \theta_i)$ represents the cumulative standard deviation bound given by Lemma 3 of Iwazaki [2025], and θ_i are the hyperparameters of the i-th objective.

Proof. By Lemma B.9, the instantaneous regret is bounded by the maximum of standard deviations in the batch. Summing over $t = 1, \ldots, T$:

$$\sum_{t=1}^{T} r_t \leq 2BL_{\mathrm{HV}} \sum_{t=1}^{T} \sum_{i=1}^{K} \max_{m \in [M]} \sigma_{t-1,i}(x_{t,m}),$$

where $L_{\text{HV}} = MK(2B - \min_j z_{\text{ref},j})^{K-1}$.

By Lemma B.10, we have:

$$\sum_{t=1}^{T} r_t \leq 2BL_{\text{HV}} \sum_{i=1}^{K} C_i(T, d, \theta_i).$$

Finally, since $r_T^{\text{simple}} \leq \frac{1}{T} \sum_{t=1}^{T} r_t$, we conclude:

$$r_T^{\text{simple}} \leq \frac{2BL_{\text{HV}}}{T} \sum_{i=1}^{K} C_i(T, d, \theta_i).$$

□

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: All contributions and scope are well reflected in the paper's abstract and introduction.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Yes, we acknowledge this limitation and discuss it explicitly in the main paper.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: All assumptions are stated explicitly in the paper.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: All necessary experimental details have been provided, and the code is publicly available to ensure reproducibility.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: Yes, the code is publicly available and implements standard Python operations.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: All necessary details have been provided and the code is available.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Error bars are shown to indicate the significance.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: All necessary details about computer resources have been provided in the Appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The paper strictly follows the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The paper briefly discusses societal impacts in the Impact Statement. Potential positive impacts include more efficient multi-objective optimization in scientific and engineering applications. Potential negative impacts are application-dependent, as the method could be used to optimize objectives with harmful downstream consequences.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: This paper does not utilize external data. All prior work and Python packages used have been properly credited through citations.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: This paper does not introduce new datasets. The code is provided and open-source.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve human subjects, user studies, or study participants, so there are no participant risks to disclose and no IRB approval was required.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: LLM does not impact the core methodology, scientific rigorousness, or originality of the research.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.