Revisiting Vanilla Bayesian Optimization in High-Dimensional Permutation Spaces

Anonymous Authors[1]

Abstract

Recent work on high-dimensional Bayesian optimization (BO) suggests that well-configured "vanilla" Gaussian-process BO can be surprisingly competitive in continuous domains. We revisit this claim in a different regime: high-dimensional permutation spaces under limited black-box evaluation budgets. In this setting, the factorial growth of the search space, distance concentration, and rugged local structure make full-space global kernels weakly discriminative, so acquisition optimization becomes unreliable. We study this failure mode empirically and propose VaPBO, a subspace BO framework that combines stochastic active-item selection, Merge-kernel modeling on the induced sub-permutation, and structure-aware completion of inactive items. Our revised experiments separate the roles of variable selection, surrogate guidance, and local search. They show that strong local search is an important driver, but subspace modeling and acquisition guidance further improve performance: on high-dimensional tasks, VaPBO-GLS obtains the best average rank among fourteen baselines and ablations. These results refine, rather than contradict, the continuous-space vanilla BO narrative: vanilla ingredients remain useful, but high-dimensional permutation optimization benefits from subspace structure and careful search attribution.

1. Introduction

Bayesian optimization (BO) is a standard framework for optimizing expensive black-box functions when gradients are unavailable and evaluations are costly (Shahriari et al., 2015; Frazier, 2018; Garnett, 2023). Classical BO with Gaussian processes (GPs) (Rasmussen & Williams, 2006) often struggles as dimensionality increases (Binois & Wycoff, 2022). To address this issue, high-dimensional BO has traditionally used structural assumptions such as additive decompositions (Kandasamy et al., 2015; Rolland et al., 2018; Mutný & Krause, 2018), low-dimensional embeddings (Wang et al., 2016; Letham et al., 2020; Papenmeier et al., 2022), and axis-aligned variable selection (Li et al., 2017; Eriksson & Jankowiak, 2021; Song et al., 2022).

This conventional view has recently been challenged in continuous domains. Several studies show that standard GPs, when equipped with robust initialization, dimension-aware length-scale scaling, and reasonable acquisition optimization, can match or outperform more complex high-dimensional BO methods (Hvarfner et al., 2024; Xu et al., 2025; Papenmeier et al., 2025). This "vanilla BO" narrative, however, is primarily established for continuous search spaces with standard kernels such as RBF or Matern kernels. It does not automatically transfer to permutation spaces, where the search domain grows factorially and where kernel similarity is governed by discrete order structure rather than Euclidean distance.

We therefore revisit vanilla BO in high-dimensional permutation spaces. A permutation space $\mathbb{S}_n$ has size $n!$, and random permutations become nearly equidistant as n grows. This distance concentration weakens global permutation kernels because most unobserved candidates look similarly far from the design set. In addition, local edits in permutations often induce rugged objective changes, making acquisition optimization difficult. These properties create a regime in which full-space global-kernel BO can remain poorly discriminative even when continuous-domain vanilla BO heuristics are applied.

To study this regime, we propose VaPBO, a variable-selection framework for permutation BO. VaPBO repeatedly selects an active subset of items, fits a Merge-kernel GP on the induced sub-permutation, optimizes the acquisition function in this lower-dimensional subspace, and reconstructs a full permutation through a structure-aware Best-K fill-in rule. The method is intentionally simple: random active-set selection is not meant to assume independence among permutation variables, but to provide a stochastic projection that reduces distance collapse under a fixed evaluation budget.

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Submitted to the AI for Science workshop (ICML 2026).

Our revised empirical study focuses not only on final performance but also on attribution. We compare against Random Search, Vanilla BO with continuous relaxation, TuRBO with continuous relaxation, BOPS, Merge-BO, Vanilla Merge-BO, and local-search-only variants. We also separate the contribution of variable selection, local search, and GP-based acquisition guidance. The results show that variable selection plus greedy local search is already strong, but surrogate-guided VaPBO-GLS provides the best overall high-dimensional rank. Thus, our conclusion is conditional rather than absolute: in high-dimensional permutation spaces with limited evaluations, subspace structure is often needed to make BO effective, while strong local search remains a major part of the optimization engine.

Our contributions are summarized as follows:

- **Diagnosis**: We identify distance concentration and rugged local structure as high-dimensional pathologies that make full-space permutation kernels weakly discriminative under limited budgets.
- **Methodology**: We propose VaPBO, a simple subspace BO protocol that combines stochastic active-item selection, subspace Merge-kernel modeling, acquisition optimization, and Best-K completion.
- **Attribution**: We add matched-budget ablations that disentangle local search, variable selection, and surrogate guidance, clarifying where the gains come from and where GP guidance has limited marginal effect.
- **Evaluation**: We evaluate on combinatorial and real-world EDA / AI-for-Science tasks, adding stronger baselines, low-dimensional boundary tests, corrected statistical comparisons, and behavior-level OTSD analyses.

2. Background

This section formalizes Bayesian optimization in permutation spaces, reviews permutation kernels, and positions our study relative to continuous high-dimensional BO and discrete BO baselines.

2.1. Bayesian Optimization in Permutation Spaces

Let $[n] = \{1, \dots, n\}$ denote n items. A permutation can be represented as a ranking vector $\pi \in \mathbb{S}_n$, where $\pi(i)$ is the position of item i. The search space is the symmetric group $\mathbb{S}_n$ with cardinality $|\mathbb{S}_n| = n!$. We consider expensive black-box functions over permutations $f : \mathbb{S}_n \to \mathbb{R}$.

Bayesian Optimization on $\mathbb{S}_n$. BO over permutations follows the standard BO loop, but replaces Euclidean kernels with permutation-aware similarities. Given observations $\mathcal{D}_t = \{(\pi_i, y_i)\}_{i=1}^t$ with $y_i = f(\pi_i) + \epsilon_i$, a GP surrogate $\mathcal{GP}(\mu_0, K(\pi, \pi'))$ is fitted to obtain a posterior mean $\mu_t(\pi)$ and uncertainty $\sigma_t(\pi)$. An acquisition function $\alpha_t(\pi)$ such as expected improvement (EI) (Mockus, 1994; Jones et al., 1998) is then maximized over $\mathbb{S}_n$ to propose the next evaluation point. Continuous relaxation (CR) provides a simple alternative by optimizing in a continuous box and mapping continuous vectors to permutations via sorting, but this representation can discard algebraic order structure.

- **Mallows kernel.** Mallows kernel adapts the RBF framework to the symmetric group via the Kendall-tau distance d_{KT} (Jiao & Vert, 2015; Deshwal et al., 2022). This distance counts discordant pairs between two rankings. It can be written through an implicit pairwise-comparison feature map $\Phi_{Mal}(\pi) \in \{0,1\}^{n(n-1)/2}$:

$$K_{\text{Mallows}}(\pi, \pi') = \exp\big(-\lambda d_{\text{KT}}(\pi, \pi')\big) = \exp\left(-\frac{\|\Phi_{\text{Mal}}(\pi) - \Phi_{\text{Mal}}(\pi')\|_2^2}{2\ell^2}\right).$$

  While mathematically robust, the Mallows kernel has $O(n^2)$ feature dimensionality, which becomes costly in high-dimensional regimes.

- **Merge kernel.** The Merge kernel (Xie & Chen, 2025) uses the fixed comparison path of merge sort to produce a structured representation $\Phi_{\text{Mer}}(\pi) \in \{0,1\}^{\Theta(n \log n)}$. This compact representation is attractive for high-dimensional permutation BO and is the kernel backbone used in this work. Details are provided in Appendix A.2.

2.2. Related Work and Scope

Continuous high-dimensional BO and the vanilla paradigm. High-dimensional BO traditionally relies on structural priors, including additive models (Kandasamy et al., 2015; Rolland et al., 2018; Mutný & Krause, 2018), embeddings (Wang et al., 2016; Letham et al., 2020; Papenmeier et al., 2022), and variable selection (Li et al., 2017; Eriksson & Jankowiak, 2021; Song et al., 2022). Recent continuous-domain studies argue that standard GPs can be much stronger than expected when length scales, priors, and acquisition optimization are configured carefully (Hvarfner et al., 2024; Xu et al., 2025; Papenmeier et al., 2025). Our study does not refute these continuous-space results. Instead, it asks whether the same intuition remains sufficient after changing both the search geometry and the kernel family to high-dimensional permutations.

Discrete and permutation BO. Discrete BO covers a broad range of spaces, including binary, categorical, sequence, graph, and permutation domains (Gonz ’alez-Duque et al., 2024). Methods such as ComBO (Oh et al., 2019) exploit graph Cartesian products, while BOPS (Deshwal et al., 2022) and Merge-BO (Xie & Chen,

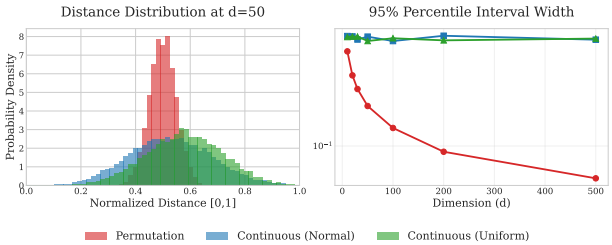

Figure 1. Empirical diagnosis of geometric collapse and concentration of measure in permutation spaces. (**Left**) At a representative dimension of $d = 50$, the normalized distance distribution between random pairs in $\mathbb{S}_n$ (red) is significantly more concentrated around the mean than in Gaussian (blue) or Uniform (green) continuous spaces. This concentration phenomenon renders global kernels uninformative, as most candidates appear nearly equidistant to the observed data. (**Right**) The 95% percentile interval width of normalized distances exhibits a sharp, factorial-rate decay as dimensionality n increases. This "Geometric Collapse" indicates that standard kernels fail to maintain discriminative power in high dimensions, providing the fundamental motivation for the structural dimensionality reduction.

2025) specialize in permutation spaces. TuRBO (Eriksson et al., 2019) is a strong continuous local BO method and can be combined with continuous relaxation for permutation tasks. We include these methods or their fair CR variants in the revised comparisons. Our contribution is complementary: we focus on the geometry of high-dimensional permutations, subspace surrogate modeling, and attribution between local search and BO guidance.

Scope of the claim. Our central claim is conditional. We study high-dimensional permutation spaces under limited black-box evaluation budgets. In this regime, full-space global-kernel BO can be weakly discriminative because distances concentrate and acquisition landscapes are rugged. Low-dimensional permutation tasks and longer-budget regimes may behave differently; we therefore report low-dimensional boundary results and budget sensitivity in the appendix.

3. Principle-Driven Approach for High-Dimensional Permutation Spaces

We first present an empirical diagnosis in Section 3.1, identifying two fundamental pathologies, i.e., geometric concentration of measure and topological ruggedness, which cause standard GP surrogates to fail as dimensionality increases. Motivated by these failures, Section 3.2 introduces VaPBO, a principle-driven framework that restores kernel informativeness. We detail how VaPBO leverages sub-permutation variable selection to restore variance and employs a structure-aware fill-in strategy to navigate the permutation spaces effectively.

3.1. Analyzing the Properties of High-Dimensional Permutation Spaces

Recent studies in continuous domains suggest that Vanilla BO can achieve state-of-the-art performance in high dimensions without subspace techniques. In this section, we investigate whether this "Vanilla Hypothesis" holds for the permutation space $\mathbb{S}_n$. Through the empirical analysis, we identify two fundamental pathologies, i.e., geometric concentration and topological ruggedness, that render global permutation kernels uninformative as n increases.

The Curse of Dimensionality: Geometric Collapse of the Permutation Space. Unlike continuous search spaces where the volume grows polynomially as c^d, the cardinality of the symmetric group $\mathbb{S}_n$ scales factorially ($n! \gg c^d$), creating a fundamentally more sparse and complex optimization landscape. In high-dimensional regimes, this complexity is further compounded by the Concentration of Measure phenomenon, which dictates that the distances between randomly sampled permutations converge tightly around a central mean with vanishing variance as n increases. This phenomenon is empirically diagnosed in Figure 1. At a relatively low dimension of $d = 50$ (Left), the normalized distance distribution for permutations is significantly more peaked than its continuous counterparts (Uniform or Gaussian), indicating that the vast majority of the space resides at a nearly constant distance from any reference point. More critically, as shown in the right sub-figure, the 95% percentile interval width of these distances exhibits a sharp, factorial-rate decay, which is a pathology we define as geometric collapse. This collapse implies that in high dimensions, any unobserved candidate π^* appears nearly equidistant to all previously observed points in the dataset.

Impact on GP Modeling From a modeling perspective, geometric collapse poses a catastrophic challenge for global permutation kernels. When distances concentrate, the resulting Gram matrix tends toward an identity or a constant matrix, effectively stripping the GP of its ability to distinguish between near and far regions. Consequently, the surrogate model fails to propagate information from the training data to unobserved regions, causing the posterior mean to remain flat and the acquisition function to become uninformative.

Topological Ruggedness and Surrogate Fragility. The issues of geometric collapse induce significant topological ruggedness, thereby posing a formidable challenge for GP regression when applying BO to high-dimensional permutation tasks. To evaluate model performance, we utilize Negative Log-Likelihood (NLL) as the primary metric, as it accounts for both predictive accuracy and uncertainty quantification (Deshwal et al., 2022; Papenmeier et al., 2025), which is a crucial balance for effective exploration and ex-

Table 1. Comparison of Negative Log-Likelihood (NLL) results for different methods after 200 iterations. Lower values indicate better model calibration and uncertainty estimation. ID denotes the results for the in-distribution test set, while OOD represents the results for out-of-distribution points with better actual quality.

Metric	CR	Mallows	Merge
`Chip Placement` (ID)	0.72	1.52	1.51
`Biodiversity` (ID)	0.54	1.51	1.51
`Chip Placement` (OOD)	1015.7	221.78	840.08
`Biodiversity` (OOD)	1251.32	854.67	1035.9

ploitation in BO. To further comprehensively assess the surrogate models' capacity to capture the underlying structure of high-dimensional permutation spaces, we evaluate their performance on two distinct test sets: an In-Distribution (ID) set, mirroring the training data distribution, and an Out-of-Distribution (OOD) set comprising points with superior objective quality. This experimental design, a standard practice in offline model-based optimization (Kim et al., 2025; Trabucco et al., 2022), is crucial for measuring a model's predictive ability on unseen, high-performing candidates.

The results in Table 1 reveal that while Continuous Relaxation (CR) achieves the lowest NLL on ID data, it has the worst NLL on OOD data. The low ID-NLL of CR is a deceptive indicator of performance, as it originates from overfitting to structured noise rather than capturing the underlying combinatorial topology. In contrast, permutation-aware Mallows and Merge kernels exhibit superior OOD robustness. They provide a compact and structure-aware representation, effectively avoiding the structural inconsistencies of continuous relaxation and achieving superior optimization performance. Despite these specialized kernels' improvements, the overall high NLL values across all methods underscore the persistent challenges of fitting accurate surrogates in high-dimensional permutation spaces.

Kernel Selection in High-Dimensional Permutation Spaces. In high-dimensional permutation spaces, the choice of kernel is governed by a critical trade-off between the ability to capture global structural invariance and the search efficiency driven by the compactness of the feature space. While the Mallows (Jiao & Vert, 2015; Deshwal et al., 2022) kernels offer robust distance metrics, their $O(n^2)$ feature dimensionality leads to significant computational and statistical redundancy as the number of objects n grows, severely limiting their sample efficiency in high-dimensional regimes. In this work, the recently proposed Merge kernel (Xie & Chen, 2025) is chosen because it leverages the divide-and-conquer logic of merge sort to produce a structured representation that matches the information-theoretic lower bound of $\Theta(n \log n)$ for lossless permutation encoding (Knuth, 1998). The construction of the Merge kernel provides a scalable approach and significantly outperforms both Mallows kernels and continuous relaxation methods in both final regret and convergence speed.

Discussions on the Loss of Right-Invariance of Merge Kernel. Although the Merge kernel offers compact feature dimensions and computational efficiency, it sacrifices a crucial feature of traditional permutation kernels, i.e., the right-invariance. A kernel K on $\mathbb{S}_n$ is *right-invariant* if $K(\pi, \pi') = K(\pi \circ \sigma, \pi' \circ \sigma)$ for any $\sigma \in \mathbb{S}_n$, meaning that the similarity depends only on the *relative ordering* (i.e., relabeling items does not change distances). This property is desirable for classical permutation optimization where the objective is defined purely over pairwise order relations, where traditional distance-based kernels such as Mallows satisfy it (Jiao & Vert, 2015). However, we argue that right invariance is not a necessary property for practical problems. In many physical and scientific domains, the absolute index of a permutation element carries intrinsic semantic weight that purely relative metrics fail to resolve. Take, for instance, the biodiversity optimization task from AI-for-Science. In this regime, a permutation $\pi \in \mathbb{S}_{2n}$ represents the true temporal sequence of global biological events, such as the first and last appearance datums of various species. In such contexts, right-invariance is not a virtue but a limitation, as it treats all positions as S_n-equivalent. The Merge kernel, by anchoring its feature map to a fixed merge-sort path, implicitly captures these absolute positional sensitivities.

3.2. Restoring Informative GP via Dimensionality Reduction

As illustrated in the right sub-figure of Figure 1, the changing rate of concentration in permutation spaces is significantly steeper than that in continuous spaces, suggesting that dimensionality reduction is potentially more effective for permutations. To mitigate the pathologies identified in Section 3.1, i.e., the concentration of distance measures and topological ruggedness, we propose VaPBO (Variable Selection of Permutation with Bayesian Optimization). We argue that applying recent vanilla techniques directly to the full permutation space yields limited improvements, as they cannot fundamentally counteract the intrinsic concentration of measure. Therefore, to restore the discriminative power of the GP, it is imperative to actively project the high-dimensional $\mathbb{S}_n$ onto a lower-dimensional subspace where the kernel's distance distribution regains sufficient variance.

However, performing such a reduction on permutation structures is non-trivial and necessitates specialized architectural designs. Unlike continuous domains where axes are independent, dimensionality reduction in permutation spaces is non-trivial due to the strict combinatorial constraints (i.e., the bijectivity of the permutation). VaPBO addresses this via a dropout-style variable selection approach (Li et al., 2017) tailored for $\mathbb{S}_n$, integrated with a Structure-Aware Fill-in

strategy to handle inactive dimensions. Detailed algorithm flow of VaPBO is provided in Appendix A.3.

Permutation Subspace Definition. We define the variable selection mechanism on the ranking representation. Let $\pi \in \mathbb{S}_n$ be the permutation vector where $\pi(i)$ denotes the position of item i. In each BO iteration, we randomly select a subset of active items $\mathcal{I} \subset \{1, \dots, n\}$ with cardinality $|\mathcal{I}| = n_a < n$. The optimization is then performed on the subspace configuration $\mathbf{z} \in \{1, \dots, n\}^{n_a}$, which represents the positions of the active items:

$$\mathbf{z} = \{\pi(i) \mid i \in \mathcal{I}\}.$$

Crucially, while standard variable selection in Euclidean space fixes inactive dimensions to a seen value, fixing variables in $\mathbb{S}_n$ is constrained by the mutual exclusion of ranks. The positions assigned to $\mathcal{I}$ dictate the available slots for the inactive set $\mathcal{I}^c$, creating a dependency that must be managed by the fill-in strategy.

Subspace Modeling with Merge Kernel. To enable sample-efficient learning, we construct the surrogate model strictly over the active subspace. For any two permutations π, π', their similarity is computed based solely on the active items $\mathcal{I}$. We employ the Merge Kernel on the sub-permutation features:

$$K_{sub}(\pi, \pi'; \mathcal{I}) = K_{\text{Merge}}(\Phi(\pi)_{\mathcal{I}}, \Phi(\pi')_{\mathcal{I}}),$$

where $\Phi(\cdot)_{\mathcal{I}}$ extracts the merge-sort comparison features relevant only to the items in $\mathcal{I}$. By restricting the kernel to n_a dimensions, we artificially inject variance into the distance distribution, preventing the Gram matrix from collapsing to the identity, which is a necessary condition for effective posterior inference in high-dimensional settings.

Structure-Aware Fill-in Strategies. A core challenge in dimensionality reduction-based permutation BO is determining the configuration of the *inactive items* $\mathcal{I}^c$. Since the objective function $f(\pi)$ requires a full permutation for evaluation, we employ a fill-in strategy Ψ that maps a subspace configuration $\mathbf{z}$ to a valid full permutation $\pi = \Psi(\mathbf{z}; \mathcal{H}_t)$, potentially utilizing history $\mathcal{H}_t$. A naive approach is to assign the items in $\mathcal{I}^c$ to the remaining available positions uniformly at random. While this ensures unbiased exploration, it disregards the high degree of conditional dependence between variables. To combat landscape ruggedness and exploit known high-quality substructures, we introduce a Best-k Position fill-in strategy. We maintain a set of the top-k best solutions found so far, denoted as $\mathcal{P}_k^* = \{\pi_1^*, \dots, \pi_k^*\}$. For a candidate subspace configuration $\mathbf{z}$ (defining positions for items $\mathcal{I}$), we select a reference template $\pi_{ref} \in \mathcal{P}_k^*$ that minimizes the positional distance in the active subspace:

$$\pi_{ref} = \arg\min_{\pi \in \mathcal{P}_k^*} \sum_{i \in \mathcal{I}} (\pi(i) - \mathbf{z}_i)^2.$$

The full permutation is constructed by fixing the active items according to $\mathbf{z}$ and filling the remaining positions using the relative ordering of $\mathcal{I}^c$ found in π_{ref}. This acts as a data-driven soft trust region, stabilizing the optimization by ensuring that unoptimized dimensions adhere to structures known to yield low objective values.

Acquisition Optimization. Optimizing $\alpha(\pi)$ over a reduced but still combinatorial subspace remains challenging. We therefore study several acquisition/search operators under a common evaluation protocol.

- **Local Search (LS)** explores the neighborhood of a candidate through swap operations. In VaPBO-LS, each neighbor is evaluated by projecting it onto the active subspace $\mathcal{I}$ and querying the subspace GP acquisition function.
- **Greedy Local Search (GLS)** allocates more neighborhood-search effort around the current incumbent rather than relying mainly on repeated random restarts. This stronger exploitative operator is especially useful in rugged permutation landscapes. Our revised main variant, VaPBO-GLS, combines GLS with the VaPBO subspace surrogate.
- **Evolutionary Algorithm (EA)** maintains a population of sub-permutations and uses structured recombination to explore disjoint acquisition regions.
- **Gradient-Based Optimization (Grad)** uses gradients in the Merge feature space as a proxy search direction. We also evaluate a LogEI version in Appendix B.2.

For attribution, we distinguish **VaP** variants, which use variable selection and fill-in without GP acquisition guidance, from **VaPBO** variants, which rank candidates using the subspace GP acquisition function. This distinction separates gains from variable selection/local search from gains due to surrogate guidance.

4. Experiments

We evaluate whether subspace modeling restores useful optimization behavior in high-dimensional permutation spaces, and we explicitly attribute gains to variable selection, surrogate guidance, and local search.

4.1. Settings and Tasks

Compared methods. We compare VaPBO with full-space BO methods, continuous-relaxation baselines, local-search-only variants, and no-surrogate VaP ablations.

- **Unstructured and continuous-relaxation baselines**: Random Search samples full permutations uniformly;

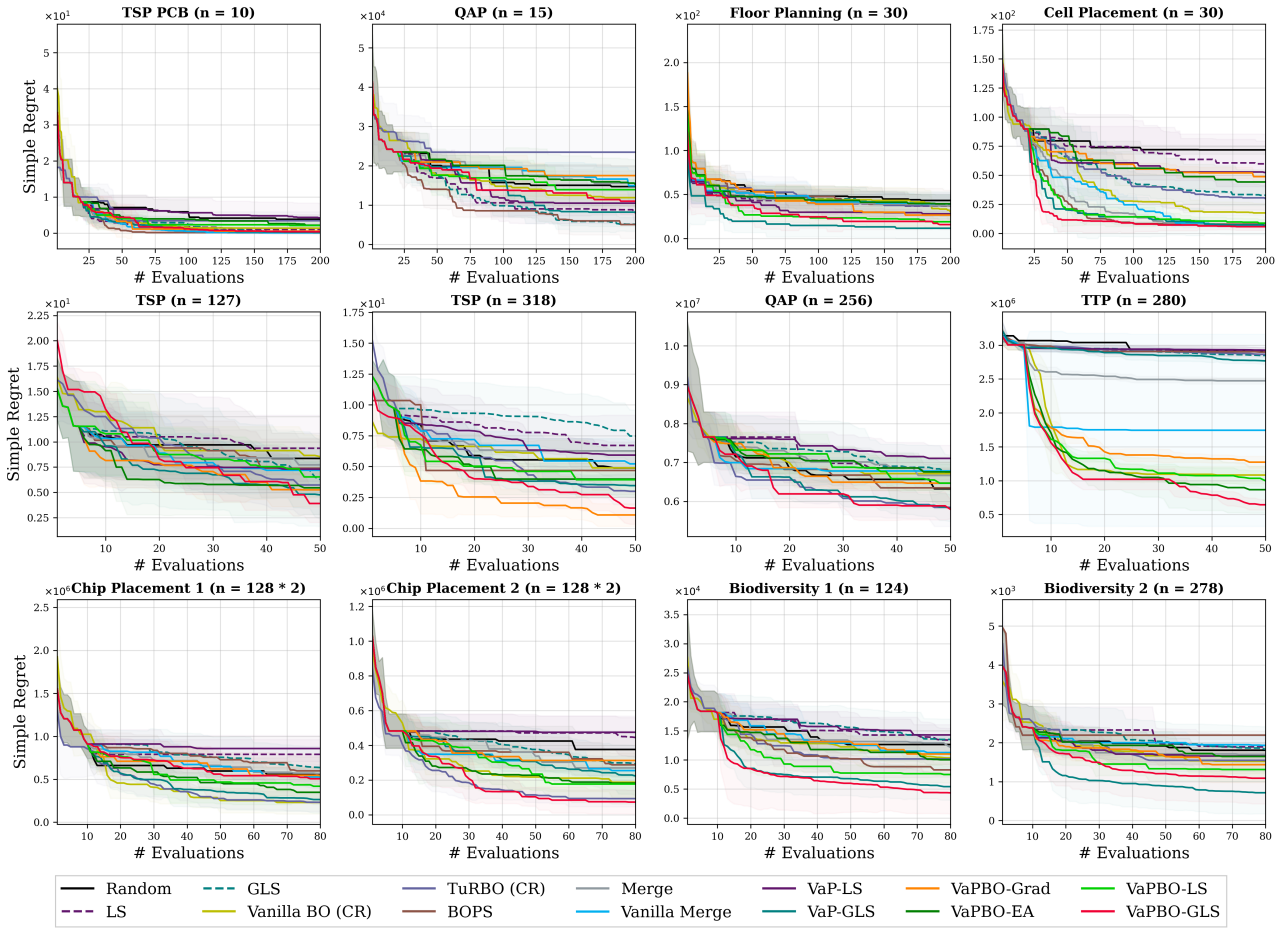

Figure 2. Best-so-far regret curves on high-dimensional tasks. The revised comparison includes Random Search, continuous-relaxation BO baselines, permutation BO baselines, local-search-only baselines, VaP ablations, and VaPBO variants. VaPBO-GLS is strongest overall, but VaP-GLS is also competitive, showing that variable selection and greedy local search account for a large part of the gain.

Vanilla BO (CR) applies standard BO in a continuous relaxation and maps candidates back to permutations; TuRBO (CR) applies TuRBO (Eriksson et al., 2019) under the same CR protocol.

- **Permutation BO baselines**: BOPS (Deshwal et al., 2022), Merge-BO (Xie & Chen, 2025), and Vanilla Merge-BO, which augments Merge-BO with recent vanilla GP practices (Xu et al., 2025; Papenmeier et al., 2025).
- **Local-search baselines**: LS and GLS use the same neighborhood-search families as our inner optimizers but do not use VaPBO's subspace GP guidance.
- **VaP ablations**: VaP-LS and VaP-GLS use stochastic active sets and Best-K fill-in but remove GP acquisition ranking. They isolate the effect of variable selection plus local search.
- **VaPBO variants**: VaPBO-LS, VaPBO-GLS, VaPBO-EA, and VaPBO-Grad combine subspace variable selection with GP acquisition guidance and different acquisition optimizers. Unless otherwise stated, n_a is set to 50% of the total dimension and $K = 3$ for fill-in.

Evaluation protocol. All methods use matched initialization and matched black-box evaluation budgets within each task. Following the high-dimensional permutation evaluation regime of Merge-BO, combinatorial optimization tasks use 50 evaluations. The EDA and biodiversity tasks use 80 evaluations because they are higher-cost application tasks, while Appendix B.3 reports a unified-budget check at 50 evaluations. Each result is averaged over five independent random seeds. We report mean $\pm$ standard deviation and use Wilcoxon rank-sum tests with two-stage Benjamini-Hochberg correction (TSBH) to control the false discovery rate at 0.05 across task-wise comparisons.

Tasks. We evaluate on permutation BO benchmarks spanning controlled combinatorial problems and large-scale real applications (Deshwal et al., 2022; Xie & Chen, 2025). The

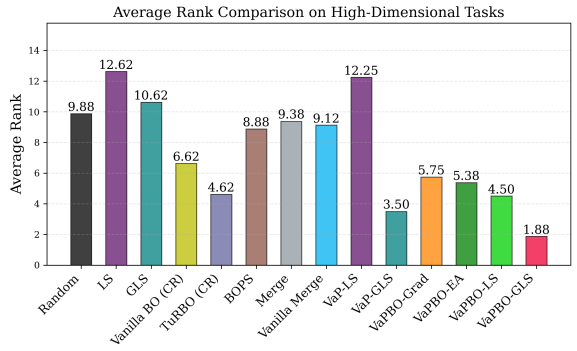

Figure 3. Average rank comparison on high-dimensional tasks. Lower is better. VaPBO-GLS obtains the best average rank (1.88/14). The gap between GLS (10.62/14), VaP-GLS (3.50/14), and VaPBO-GLS (1.88/14) separates the contributions of local search, variable selection, and surrogate-guided acquisition.

combinatorial suite includes traveling salesperson (TSP), quadratic assignment (QAP), and traveling thief (TTP) tasks. The application suite includes chip placement from electronic design automation (Mirhoseini et al., 2021) and biodiversity history reconstruction from quantitative paleobiology (Fan et al., 2020; Tang et al., 2024; Lu et al., 2026). These tasks cover dimensions up to 318 and include cases where permutation variables have strong coupling constraints.

4.2. Main Results

Is full-space vanilla BO sufficient? The high-dimensional results in Figures 2–3 and Table 2 suggest a conditional answer: full-space vanilla BO is not sufficient in the high-dimensional permutation regime considered here. Vanilla BO (CR), Merge-BO, and Vanilla Merge-BO improve on some tasks, but they do not close the gap to the best subspace variants. This supports the claim that continuous-space vanilla BO ingredients remain useful but need additional subspace structure in permutation spaces.

Where do the gains come from? The attribution pattern is more nuanced than a surrogate-only explanation. Local search alone is weak in the high-dimensional comparison: GLS and LS rank 10.62/14 and 12.62/14, respectively. Adding VaP-style variable selection and fill-in substantially improves the same search shell, with VaP-GLS improving over GLS from 10.62/14 to 3.50/14. Finally, GP-guided acquisition adds a smaller but consistent improvement on top of this strong shell, with VaPBO-GLS improving over VaP-GLS from 3.50/14 to 1.88/14. Thus, the strongest performance comes from combining subspace variable selection, surrogate-guided candidate ranking, and greedy local exploitation; the surrogate is helpful, but not the sole source of improvement.

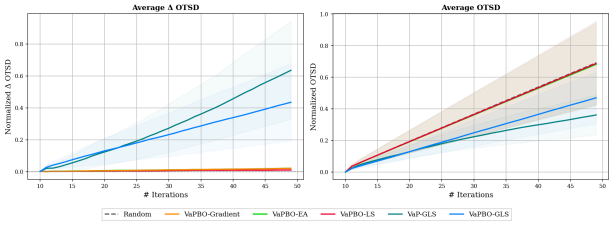

Figure 4. OTSD analysis with VaP-GLS included. VaP-GLS removes surrogate guidance while preserving the VaP subspace and GLS search shell. The comparison separates trajectory geometry from final optimization performance.

Calibration versus online gain. Table 3 shows that reduced-space modeling improves ID/OOD NLL relative to full-space baselines. However, better predictive calibration does not necessarily translate one-to-one into large final regret gains in rugged discrete landscapes, especially when GLS already captures many immediate local improvements. We therefore interpret the surrogate contribution as incremental guidance for ranking and uncertainty-aware proposal selection rather than as the only optimization driver.

Search behavior through OTSD. Figure 4 compares OTSD trajectories for VaPBO variants and the VaP-GLS ablation. OTSD measures the geometry of evaluated trajectories, not the usefulness of acquisition ranking. VaP-GLS without surrogate is the most exploitation-oriented variant, while VaPBO-GLS shows a more balanced trajectory and achieves the best average rank. This distinction explains why similar or exploitative geometric traces do not imply equivalence to random search.

Low-dimensional boundary. Our claim is not that dimensionality reduction is unconditionally necessary. In low-dimensional tasks, BOPS and VaPBO-GLS are comparable, and BOPS can be strong. Appendix B.1 reports low-dimensional rankings and ComBO comparisons. This boundary result supports the scope of our claim: subspace reduction is most useful in high-dimensional permutation regimes under tight budgets.

5. Limitations and Scope

VaPBO uses random active-item selection. This simple design helps expose the distance-collapse bottleneck, but it does not exploit objective-informed dependencies among permutation variables. Dependence-aware active selection is an important direction for future work. The results also show that strong greedy local search explains a large fraction of the performance gain; the GP surrogate provides additional guidance, but its marginal contribution can be compressed when the local-search shell is already strong. In addition, improved OOD NLL should be interpreted as

Table 2. High-dimensional regret results. Each entry is mean ± standard deviation over five independent runs. The symbols −, +, and ≈ indicate whether the method is significantly worse, better, or comparable to VaPBO-GLS under a Wilcoxon rank-sum test with TSBH correction at FDR 0.05.

Method	TSP127	TSP318	QAP256	TTP280	CP1	CP2	Bio1	Bio2	Avg. rank
Random	8.36±2.50−	4.88±1.30−	632.43±51.01 ≈	291.09±0.00−	55.80±7.99 ≈	37.58±10.72−	1.27±0.15 ≈	0.18±0.02 ≈	9.88
LS	9.36±3.27−	6.71±2.45−	677.75±35.28−	290.95±5.95−	79.02±17.29 ≈	44.49±11.49−	1.34±0.20−	0.19±0.03−	12.62
GLS	6.27±3.89 ≈	7.47±2.11−	672.70±46.58−	284.80±6.29−	63.65±24.31 ≈	29.12±9.74−	1.35±0.35 ≈	0.18±0.06 ≈	10.62
Vanilla BO (CR)	8.53±2.05−	4.89±1.79−	671.49±43.25−	108.22±12.58−	23.04±13.44 ≈	18.74±8.89 ≈	1.04±0.29 ≈	0.15±0.03 ≈	6.62
TuRBO (CR)	5.73±2.93 ≈	3.00±1.11 ≈	584.88±44.84 ≈	286.44±4.20 ≈	23.22±10.65 ≈	9.29±4.71 ≈	1.02±0.21 ≈	0.15±0.03 ≈	4.62
BOPS	7.39±2.90 ≈	4.69±0.00−	634.93±51.81 ≈	289.65±6.21−	59.79±16.29 ≈	28.75±6.78−	0.83±0.12 ≈	0.22±0.01 ≈	8.88
Merge	7.72±2.67 ≈	6.27±1.00−	666.78±38.21−	247.33±4.77−	56.74±10.50 ≈	26.95±4.58−	1.22±0.17−	0.17±0.03 ≈	9.38
Vanilla Merge	7.19±2.25 ≈	5.22±1.88−	678.21±34.49−	174.35±141.78 ≈	52.74±7.72 ≈	25.43±3.79−	1.13±0.27 ≈	0.19±0.02 ≈	9.12
VaP-LS	7.30±2.16−	5.93±1.07−	710.54±33.62−	292.58±4.51−	85.80±15.30 ≈	47.49±9.35−	1.43±0.18−	0.17±0.02 ≈	12.25
VaP-GLS	4.75±3.11 ≈	3.46±0.96 ≈	582.57±34.42 ≈	276.71±5.21−	26.42±15.70 ≈	22.34±11.21 ≈	0.54±0.23 ≈	0.07±0.05 ≈	3.50
VaPBO-Grad	5.25±1.69 ≈	1.08±0.92 ≈	646.96±17.00−	127.25±8.74−	54.22±10.57 ≈	31.26±6.02−	1.10±0.18−	0.14±0.05 ≈	5.75
VaPBO-EA	5.46±2.64 ≈	3.99±1.26−	675.94±20.47−	86.63±34.54 ≈	34.83±10.48 ≈	18.20±11.98 ≈	1.00±0.08 ≈	0.16±0.01 ≈	5.38
VaPBO-LS	6.53±1.94 ≈	3.93±1.38−	647.01±55.04 ≈	99.77±12.30−	41.89±21.02 ≈	17.69±8.07 ≈	0.75±0.14 ≈	0.13±0.03 ≈	4.50
VaPBO-GLS	3.90±1.87	1.64±1.31	579.68±24.42	64.40±7.57	50.64±28.78	7.30±6.86	0.43±0.36	0.11±0.07	1.88

Table 3. Reduced-space NLL results. Lower values indicate better model calibration. Compared with the full-space results in Table 1, VaPBO substantially improves OOD calibration on the application tasks.

Metric	**Merge**	**Vanilla Merge**	**VaPBO**
`Chip Placement` (ID)	1.51	52.45	**1.30**
`Biodiversity` (ID)	1.51	77.21	**1.30**
`Chip Placement` (OOD)	840.08	40.08	**5.06**
`Biodiversity` (OOD)	1035.9	52.87	**11.03**

better calibration rather than a guarantee of equally large online optimization gains. Finally, the Merge kernel is not right-invariant. This trade-off is acceptable for our application tasks, where absolute item identities and positions can carry semantic information, but kernels with stronger invariance may be preferable for objectives whose symmetries demand them. Our statistical analysis uses five independent runs with corrected tests; this improves reliability but does not eliminate all uncertainty under expensive evaluation budgets.

6. Conclusion

We revisited vanilla BO in high-dimensional permutation spaces and showed that conclusions from continuous domains do not transfer automatically. Full-space global kernels can become weakly discriminative because permutation distances concentrate and acquisition landscapes are rugged. VaPBO addresses this regime through stochastic subspace selection, Merge-kernel modeling, and structure-aware completion. The revised experiments clarify the attribution: variable selection and strong local search are major drivers, while surrogate-guided acquisition provides additional improvements and the best overall high-dimensional rank. These findings position VaPBO as a practical subspace BO framework and, more broadly, suggest that high-dimensional permutation BO requires joint attention to geometry, surrogate calibration, and search operators.

Impact Statement

This paper presents work whose goal is to advance the field of Bayesian optimization in high-dimensional permutation spaces. There are many potential societal consequences of our work, none of which we feel must be specifically highlighted here.

References

Binois, M. and Wycoff, N. A survey on high-dimensional Gaussian process modeling with application to Bayesian optimization. *ACM Transactions on Evolutionary Learning and Optimization*, 2(2):1–26, 2022.

Deshwal, A., Belakaria, S., Doppa, J. R., and Kim, D. H. Bayesian optimization over permutation spaces. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 6515–6523, 2022.

Eriksson, D. and Jankowiak, M. High-dimensional Bayesian optimization with sparse axis-aligned subspaces. In *Proceedings of the 37th Conference on Uncertainty in Artificial Intelligence*, pp. 493–503, Virtual, 2021.

Eriksson, D., Pearce, M., Gardner, J. R., Turner, R. D., and Poloczek, M. Scalable global optimization via local Bayesian optimization. In *Advances in Neural Information Processing Systems 32*, pp. 5497–5508, Vancouver, Canada, 2019.

Fan, J.-x., Shen, S.-z., Erwin, D. H., Sadler, P. M., MacLeod, N., Cheng, Q.-m., Hou, X.-d., Yang, J., Wang, X.-d., Wang, Y., et al. A high-resolution summary of cambrian to early triassic marine invertebrate biodiversity. *Science*, 367(6475):272–277, 2020.

Frazier, P. I. A tutorial on Bayesian optimization. *CoRR abs/1807.02811*, 2018.

Garnett, R. *Bayesian Optimization*. Cambridge University Press, 2023.

Gonz
'alez-Duque, M., Michael, R., Bartels, S., Zainchkovskyy, Y., Hauberg, S., and Boomsma, W. A survey and benchmark of high-dimensional Bayesian optimization of discrete sequences. In *Advances in Neural Information Processing Systems 37*, pp. 140478–140508, 2024.

Hvarfner, C., Hellsten, E. O., and Nardi, L. Vanilla Bayesian optimization performs great in high dimensions. In *Proceedings of the 41st International Conference on Machine Learning*, Vienna, Austria, 2024.

Jiao, Y. and Vert, J.-P. The kendall and mallows kernels for permutations. In *International Conference on Machine Learning*, pp. 1935–1944, 2015.

Jones, D. R., Schonlau, M., and Welch, W. J. Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, 13(4):455–492, 1998.

Kandasamy, K., Schneider, J. G., and Póczos, B. High dimensional Bayesian optimisation and bandits via additive models. In *Proceedings of the 32nd International Conference on Machine Learning*, pp. 295–304, Lille, France, 2015.

Kim, M., Gu, J., Yuan, Y., Yun, T., Liu, Z., Bengio, Y., and Chen, C. Offline model-based optimization: Comprehensive review. *arXiv:2503.17286*, 2025.

Knuth, D. E. *The Art of Computer Programming: Sorting and Searching, Volume 3*. Addison-Wesley Professional, 1998.

Letham, B., Calandra, R., Rai, A., and Bakshy, E. Re-examining linear embeddings for high-dimensional Bayesian optimization. In *Advances in Neural Information Processing Systems 33*, pp. 1546–1558, Vancouver, Canada, 2020.

Li, C., Gupta, S., Rana, S., Nguyen, V., Venkatesh, S., and Shilton, A. High dimensional Bayesian optimization using dropout. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pp. 2096–2102, Melbourne, Australia, 2017.

Lu, Z., Xue, K., Deng, Y., et al. Complex marine ecological response during the eocene-oligocene revealed by global foraminiferal record. *Nature Communications*, 17:3954, 2026. doi: 10.1038/s41467-026-70541-w.

Mirhoseini, A., Goldie, A., Yazgan, M., Jiang, J. W., Songhori, E., Wang, S., Lee, Y.-J., Johnson, E., Pathak, O., Nazi, A., et al. A graph placement methodology for fast chip design. *Nature*, 594(7862):207–212, 2021.

Mockus, J. Application of Bayesian approach to numerical methods of global and stochastic optimization. *Journal of Global Optimization*, 4(4):347–365, 1994.

Mutný, M. and Krause, A. Efficient high dimensional Bayesian optimization with additivity and quadrature Fourier features. In *Advances in Neural Information Processing Systems 31*, pp. 9005–9016, Montreal, Canada, 2018.

Oh, C., Tomczak, J., Gavves, E., and Welling, M. Combinatorial Bayesian optimization using the graph cartesian product. In *Advances in Neural Information Processing Systems 32*, 2019.

Oh, C., Bondesan, R., Gavves, E., and Welling, M. Batch Bayesian optimization on permutations using the acquisition weighted kernel. In *Advances in Neural Information Processing Systems*, volume 35, pp. 6843–6858, 2022.

Papenmeier, L., Nardi, L., and Poloczek, M. Increasing the scope as you learn: Adaptive Bayesian optimization in nested subspaces. In *Advances in Neural Information Processing Systems 35*, New Orleans, LA, 2022.

Papenmeier, L., Poloczek, M., and Nardi, L. Understanding high-dimensional Bayesian optimization. In *Proceedings of the 42nd International Conference on Machine Learning*, Vancouver, Canada, 2025.

Rasmussen, C. E. and Williams, C. K. I. *Gaussian Processes for Machine Learning*. The MIT Press, Cambridge, MA, 2006.

Rolland, P., Scarlett, J., Bogunovic, I., and Cevher, V. High-dimensional Bayesian optimization via additive models with overlapping groups. In *Proceedings of the 21st International Conference on Artificial Intelligence and Statistics*, pp. 298–307, Playa Blanca, Spain, 2018.

Shahriari, B., Swersky, K., Wang, Z., Adams, R. P., and De Freitas, N. Taking the human out of the loop: A review of Bayesian optimization. *Proceedings of the IEEE*, 104(1):148–175, 2015.

Song, L., Xue, K., Huang, X., and Qian, C. Monte Carlo tree search based variable selection for high dimensional Bayesian optimization. In *Advances in Neural Information Processing Systems 35*, New Orleans, LA, 2022.

Tang, Q., Zheng, W., Zhang, S., Fan, J., Riedman, L. A., Hou, X., Muscente, A., Bykova, N., Sadler, P. M., Wang, X., et al. Quantifying the global biodiversity of proterozoic eukaryotes. *Science*, 386(6728):eadm9137, 2024.

Trabucco, B., Geng, X., Kumar, A., and Levine, S. Design-Bench: Benchmarks for data-driven offline model-based optimization. In *Proceedings of the 39th International Conference on Machine Learning*, pp. 21658–21676, Baltimore, MD, 2022.

Wang, Z., Hutter, F., Zoghi, M., Matheson, D., and de Feitas, N. Bayesian optimization in a billion dimensions via random embeddings. *Journal of Artificial Intelligence Research*, 55(1):361–387, 2016.

Xie, Z. and Chen, L. From sorting algorithms to scalable kernels: Bayesian optimization in high-dimensional permutation spaces. *arXiv:2507.13263*, 2025.

Xu, Z., Wang, H., Phillips, J. M., and Zhe, S. Standard Gaussian process is all you need for high-dimensional Bayesian optimization. In *Proceedings of the 13th International Conference on Learning Representations*, Singapore, 2025.

Zaefferer, M., Stork, J., and Bartz-Beielstein, T. Distance measures for permutations in combinatorial efficient global optimization. In *International Conference on Parallel Problem Solving from Nature*, pp. 373–383, 2014.

A. Details of methods.

A.1. Distances on $\mathbb{S}_n$.

A key ingredient for modeling permutations is a notion of similarity/distance. Following standard practice in permutation BO (Deshwal et al., 2022; Oh et al., 2022; Xie & Chen, 2025), we highlight two widely used distances:

(i) Kendall distance counts the number of discordant pairs

$$d_{\mathrm{KT}}(\pi,\pi') = n_d(\pi,\pi') = \sum_{1\le i<j\le n} \mathbb{I}\Big[(\pi(i)-\pi(j)) \cdot (\pi'(i)-\pi'(j)) < 0\Big].$$

(ii) Position-based distances measure absolute position discrepancies, e.g., the Spearman footrule

$$d_{\mathrm{pos}}(\pi,\pi') = \sum_{i=1}^{n} \big|\pi(i)-\pi'(i)\big|.$$

These distances induce different invariances and inductive biases; see Jiao & Vert (2015); Zaefferer et al. (2014) for a systematic discussion.

A.2. Details of the Merge Kernel (Xie & Chen, 2025)

To mitigate the pathologies of geometric collapse and distance concentration in high-dimensional permutation spaces, VaPBO utilizes the Merge Kernel as its primary surrogate component. This section formalizes the construction of the merge-sort-inspired feature mapping $\Phi_{Mer}(\cdot)$ and discusses the structural modifications required to ensure its validity for GP inference, as shown in Algorithm 1.

Theoretical Motivation: Sorting as Feature Generation The fundamental insight behind the Merge Kernel is that any comparison-based sorting algorithm defines a unique mapping from a permutation $\pi \in \mathbb{S}_n$ to a sequence of binary outcomes. By recording these outcomes (i.e., whether a swap or a specific relative ordering was observed), we can generate a feature vector that losslessly encodes the structural information of the permutation.

However, a critical requirement for a kernel function is that the feature mapping must be deterministic and reside in a fixed-dimensional Hilbert space. While standard sorting algorithms are often adaptive (with data-dependent execution paths), the Merge Kernel framework introduces a fixed comparison path. This ensures that for any input, the algorithm traverses a predetermined sequence of comparisons, making the resulting feature vector $\Phi_{Mer}(\pi)$ compatible with standard RBF kernels.

Algorithm and Implementation The mapping $\Phi_{Mer}(\pi)$ is constructed via a recursive divide-and-conquer procedure. Unlike vanilla merge sort, which terminates merging once a sub-list is exhausted, our implementation employs **Fixed-Length Padding**. This structural constraint ensures that the feature dimensionality remains constant at $\Theta(n \log n)$, which corresponds to the information-theoretic lower bound for lossless permutation encoding.

Algorithm 1 Merge Feature Mapping with Fixed Comparison Path

Function MERGEFEATURE(π, n):
Input: Permutation $\pi \in \mathbb{S}_n$, dimension n
Output: Binary feature vector V
if $n = 1$ **then**
  Return: [] // Base case: single element
end if
if $n = 2$ **then**
  Return: $[\mathbb{I}(\pi[0] > \pi[1])]$ // Base case: single comparison
end if
$mid \leftarrow \lfloor n/2 \rfloor$
$V_L \leftarrow$ MergeFeature$(\pi[0 : mid], mid)$ // Recursive split
$V_R \leftarrow$ MergeFeature$(\pi[mid : n], n - mid)$
// Track merge path using sorted sub-sequences
$\hat{\pi}_l \leftarrow$ Sort$(\pi[0 : mid])$, $\hat{\pi}_r \leftarrow$ Sort$(\pi[mid : n])$
$V_{\text{Merge}} \leftarrow []$, $i \leftarrow 0$, $j \leftarrow 0$
while $i <$ length$(\hat{\pi}_l)$ **and** $j <$ length$(\hat{\pi}_r)$ **do**
  if $\hat{\pi}_l[i] > \hat{\pi}_r[j]$ **then**
    V_{Merge}.append(1); $j \leftarrow j + 1$
  else
    V_{Merge}.append(0); $i \leftarrow i + 1$
  end if
end while
Apply fixed padding to V_{Merge} to ensure length $n - 1$
Return: $V_L \oplus V_R \oplus V_{\text{Merge}}$ // Concatenate features

A.3. Detailed flow of VaPBO

This section provides a detailed procedural walkthrough of the Variable Selection for Permutation Bayesian Optimization (VaPBO) framework as formalized in Algorithm 2.

Stage 1: Dynamic Variable Selection. The algorithm begins by identifying a subset of active items $\mathcal{I} \subset \{1, \ldots, n\}$ with cardinality n_a, where $n_a < n$. This selection is updated periodically every N_{update} iterations to ensure that the optimizer does not become trapped in a stagnant subspace. By focusing only on n_a elements, we effectively project the factorial search space onto a lower-dimensional manifold where the distance distribution retains sufficient variance for effective modeling. Note that our selection of dropout stems from its simplicity, as it avoids introducing complex algorithmic components. It functions as the simplest variable selection algorithm, and in practice, any variable selection

algorithm can replace it.

Stage 2: Subspace Projection and Modeling. Once the active set is defined, all previously observed permutations in the dataset $\mathcal{D}_{t-1}$ are projected into their subspace representations $Z_{1:t-1}$. The GP surrogate is then fitted strictly over this active subspace using the Merge Kernel. This construction ensures that the kernel similarity is computed based solely on the structural comparison features relevant to the items in $\mathcal{I}$, thereby preventing the Gram matrix from collapsing into an uninformative identity matrix.

Stage 3: Acquisition Optimization. The acquisition function α_t is maximized exclusively within the combinatorial subspace $z \in \{1, \dots, n\}^{n_a}$. Although the underlying search space remains discrete, the reduced dimensionality allows for more intensive search strategies. While local search serves as the default optimizer, VaPBO can be extended to utilize evolutionary algorithms or gradient-assisted solvers in the feature space to escape local optima and navigate the rugged acquisition landscape.

Stage 4: Structure-Aware Reconstruction. To evaluate the proposed candidate z_t^* in the objective function f, it must be mapped back to a valid full permutation $\pi_t \in \mathbb{S}_n$. VaPBO employs the Best-k Position fill-in strategy, which identifies a reference template from the top-k historical solutions. By fixing the active items according to z_t^* and filling the remaining slots using the relative ordering found in the template, the algorithm creates a data-driven soft trust region that stabilizes the optimization against landscape ruggedness.

Algorithm 2 VaPBO: Variable Selection of Permutation with Bayesian Optimization

Input: Black-box function $f : \mathbb{S}_n \to \mathbb{R}$, dimension n, active dimension n_a, update frequency n_{update}, fill-in strategy Ψ
Initialize: Data $\mathcal{D}_0 \leftarrow$ InitialDesign(), history $\mathcal{H}_0 \leftarrow \mathcal{D}_0$
for $t = 1, \dots, T$ **do**
  // Variable Selection
  if $t \pmod{n_{\text{update}}} == 0$ **then**
    Sample active index set $\mathcal{I}_t \subset \{1, \dots, n\}$ with $|\mathcal{I}_t| = n_a$ uniformly at random
  else
    Keep previous active set $\mathcal{I}_t \leftarrow \mathcal{I}_{t-1}$
  end if
  // Subspace Projection & Modeling
  Project observed permutations: $\mathbf{Z}_{1:t-1} \leftarrow \{\pi(\mathcal{I}_t) \mid (\pi, y) \in \mathcal{D}_{t-1}\}$
  Fit GP surrogate $\mathcal{M}_t$ using $K_{\text{sub}}(\mathbf{z}, \mathbf{z}') = K_{\text{Merge}}(\Phi(\mathbf{z}), \Phi(\mathbf{z}'))$
  // Acquisition Optimization (in Subspace)
  Find candidate: $\mathbf{z}_t^* \leftarrow \arg\max_{\mathbf{z}} \alpha_t(\mathbf{z} \mid \mathcal{M}_t)$ via local search
  // Structure-Aware Fill-in
  Construct full permutation $\pi_t \leftarrow \Psi(\mathbf{z}_t^*; \mathcal{H}_{t-1})$ // Note: Ψ fixes $\mathcal{I}_t$ items and fills $\mathcal{I}_t^c$ using historical bests
  // Evaluation & Update
  Evaluate $y_t \leftarrow f(\pi_t)$ and update $\mathcal{D}_t \leftarrow \mathcal{D}_{t-1} \cup \{(\pi_t, y_t)\}$
  Update history/strategy state (e.g., update Top-K list in Ψ)
end for
Return: Best solution found π_{best}

B. Additional Results

B.1. Low-dimensional comparisons

Low-dimensional experiments provide a boundary condition for our main claim. Figure 5 summarizes the average-rank comparison in these smaller settings, where BOPS and VaPBO-GLS are comparable and BOPS can be highly competitive. Figure 6 further compares against ComBO on representative low-dimensional tasks. Together, these results confirm that subspace reduction is not unconditionally necessary; its advantage becomes clearer in the high-dimensional regime.

B.2. Additional ablations and robustness

Best-K fill-in sensitivity. Figure 7 evaluates different values of K in the Best-K completion strategy. The effect is generally moderate, and $K = 3$ provides a practical

Table 4. Low-dimensional regret results. Each result is mean ± standard deviation over five independent runs. Symbols are defined as in Table 2.

Method	TSP10	QAP15	Floor Planning30	Cell Placement30	Avg. rank
Random	3.82±1.28−	14707.60±5128.01 ≈	43.20±5.78−	71.80±3.71−	12.50
LS	1.01±1.33 ≈	8040.80±2078.41 ≈	36.80±11.79−	59.80±19.64−	7.75
GLS	0.28±0.25 ≈	5136.00±1780.90+	38.00±15.18−	32.80±8.73−	6.75
Vanilla BO (CR)	1.34±1.13 ≈	11897.60±908.88 ≈	30.20±10.48 ≈	17.60±8.36−	7.25
TuRBO (CR)	3.40±2.49−	23452.00±5848.59−	37.00±15.07−	29.80±13.17−	10.50
BOPS	0.05±0.10+	5053.60±3575.96 ≈	37.00±17.89 ≈	8.00±0.89 ≈	3.75
Merge	0.13±0.20 ≈	15536.00±4144.56 ≈	37.20±10.01−	6.80±2.14 ≈	7.50
Vanilla Merge	0.05±0.06+	14934.00±3937.79 ≈	39.40±11.09−	6.40±1.20 ≈	6.25
VaP-LS	4.20±4.04−	10492.40±5213.00 ≈	28.00±3.03−	52.40±32.89−	9.00
VaP-GLS	0.13±0.00+	8210.40±3611.51 ≈	11.60±11.76 ≈	6.40±7.74 ≈	2.75
VaPBO-Grad	0.52±0.34 ≈	17500.80±1189.08−	26.60±13.97−	48.80±4.35−	8.75
VaPBO-EA	2.17±1.03−	15248.80±5784.04 ≈	39.40±9.58−	44.20±10.55−	11.00
VaPBO-LS	2.09±1.85−	13924.40±2607.40 ≈	19.00±6.87 ≈	8.80±3.25 ≈	6.75
VaPBO-GLS	0.36±0.25	10990.40±3278.00	15.80±5.53	5.80±3.12	3.75

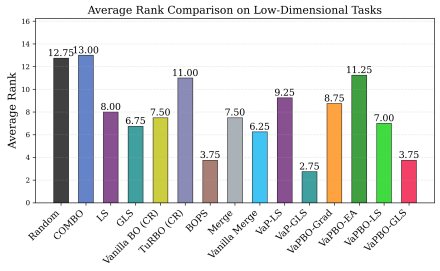

Figure 5. Average rank comparison on low-dimensional tasks. BOPS and VaPBO-GLS are comparable, indicating that specialized full-space permutation methods can remain strong when dimensionality is modest.

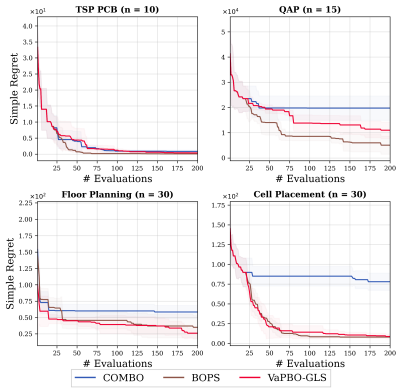

Figure 6. Low-dimensional comparison with ComBO. The plot contrasts ComBO with the main low-dimensional baselines and VaPBO variants, showing that ComBO is competitive in some small problems. We therefore treat ComBO as an important boundary-case baseline, while focusing the main benchmark on high-dimensional settings where scalability becomes challenging.

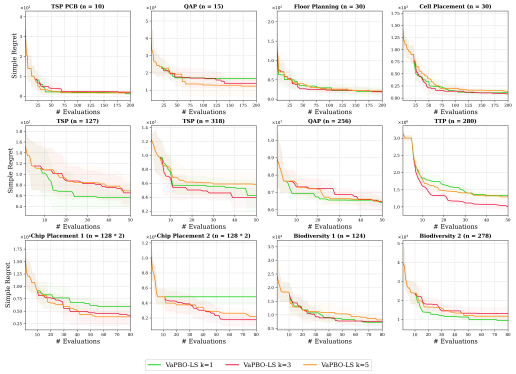

Figure 7. Sensitivity to the Best-K fill-in parameter. The figure compares VaPBO behavior under different numbers of historical templates used for structure-aware completion, showing that a moderate value such as $K = 3$ gives a stable trade-off between exploiting strong incumbents and avoiding overdependence on a single template.

trade-off between exploiting strong historical templates and avoiding excessive dependence on a single incumbent.

LogEI for VaPBO-Grad. Figure 8 compares EI and LogEI for VaPBO-Grad. LogEI slightly improves stability on several tasks, consistent with prior observations about numerical stability in acquisition optimization.

B.3. Unified-budget sensitivity

Figure 9 reports a unified 50-evaluation comparison across high-dimensional tasks. The relative ranking remains stable, supporting that the main conclusions are not an artifact of the 50/80 default split.

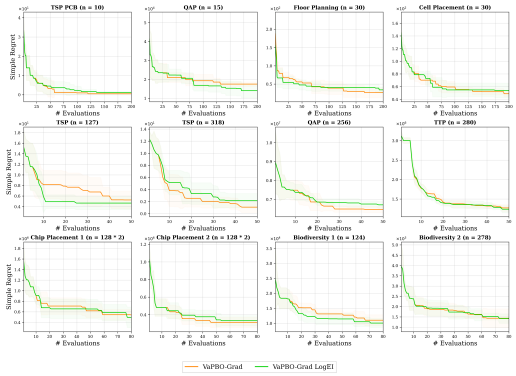

Figure 8. EI versus LogEI for VaPBO-Grad. The comparison isolates the acquisition objective used by the gradient-based variant and shows that LogEI can improve numerical stability and final performance on several tasks without changing the underlying subspace modeling protocol.

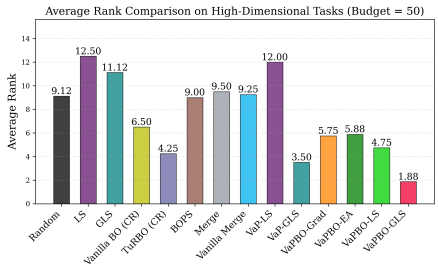

Figure 9. Average rank comparison on high-dimensional tasks under a unified 50-evaluation budget. This control removes the default 50/80-budget split between combinatorial and application tasks, and the stable ranking indicates that the main conclusion is not an artifact of using larger budgets for EDA and biodiversity tasks.

B.4. Original active-dimension and OTSD analyses

The active-ratio sensitivity in Figure B.4 shows that very large active subsets can reintroduce distance concentration, while moderate reduction is more reliable. The updated OTSD analysis in the main text supersedes the original appendix-only OTSD plot and clarifies that trajectory geometry alone does not determine whether surrogate ranking is useful.

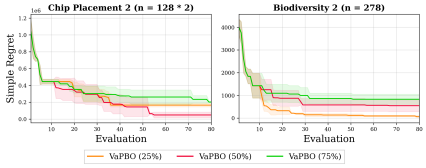

Figure 10. Sensitivity of the active dimension ratio n_a/n. The plot varies the fraction of active permutation elements used by VaPBO, illustrating that overly large active subsets can reintroduce distance concentration, while moderate reduction keeps the subspace more informative.