LARGE LANGUAGE MODELS AS PRIOR ELICITORS FOR SAMPLE-EFFICIENT BLACK-BOX OPTIMIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Many scientific and engineering applications require optimizing complex decisions whose outcomes can only be evaluated through costly simulations, experiments, or high-fidelity models, giving rise to black-box optimization (BBO) under tight evaluation budgets. Existing BBO methods typically learn from objective evaluations collected on the target task and therefore rely heavily on costly observation-driven exploration. Large language models (LLMs), with broad cross-domain knowledge, offer potential opportunities to alleviate this dependence. This study introduces a different perspective on using LLMs for BBO: rather than using LLMs to assist sampling, we elicit embedded domain knowledge of LLMs to explicitly form a probabilistic distribution over the objective function. The key idea is to develop an agentic preference elicitation method, where an LLM agent is established to analyze the problem context and to generate pairwise preferences between candidate solutions, thereby deriving a probabilistic distribution through Bayesian preference learning before expensive evaluations are collected. The elicited distribution is then incorporated into Bayesian optimization, serving as a prior of the unknown objective. Theoretically, we show that the elicited distribution contracts predictive uncertainty and information gain and, under an explicit alignment condition, yields a no-larger finite-budget GP-LCB simple-regret bound than the reference prior. We further establish controlled sensitivity to corrupted preferences and show that, under standard GP-LCB conditions, an initially biased prior does not prevent asymptotic convergence to the global optimum. Experiments on six real-world benchmarks demonstrate improved sample efficiency, with the largest gains under limited evaluation budgets.

1 INTRODUCTION

Many scientific and engineering applications require optimizing complex decisions whose impacts can only be evaluated through expensive simulations, field experiments, or high-fidelity models. This setting naturally gives rise to black-box optimization (BBO), where the goal is to identify promising solutions under limited evaluation budgets.

Extensive studies have been devoted to addressing BBO problems, including random search, Bayesian optimization, surrogate-based optimization, and inverse-learning-based BBO methods (Srinivas et al., 2010; Hutter et al., 2011; Eriksson et al., 2019; Zhang et al., 2025; Yun et al., 2025). A common feature of these methods is that they seek promising solutions by learning from evaluations collected on the target problem. These methods have achieved promising results in real-world practice. However, their search behavior is driven almost entirely by observations acquired during optimization. Consequently, their effectiveness depends on collecting sufficiently informative samples to cover the feasible domain, reduce model uncertainty, and reveal promising regions of the objective landscape. Under tight evaluation budgets, such observation-driven exploration becomes costly, limiting their effectiveness when only a small number of queries can be afforded.

Recent advances in large language models (LLMs) offer a new opportunity to alleviate this limitation. Through large-scale pretraining, LLMs encode broad cross-domain knowledge about physical systems, engineering trade-offs, and qualitative relationships among decision variables. Such knowledge can potentially provide task-specific guidance, making LLMs a promising source of prior information for sample-efficient BBO.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Recent studies have begun to integrate LLMs with BBO to improve optimization efficiency (Liu et al., 2024; Suwandi et al., 2026; Chang et al., 2025). The main idea is to embed LLMs into the optimization loop and employ their domain knowledge to enhance the quality of observations (e.g., by suggesting warm-start samples, search regions, and kernel functions). Despite their effectiveness in enhancing observations, a fundamental challenge of BBO is largely unexplored: how can we effectively uncover the implicit input–output relationship of the expensive objective function? This study takes a different perspective of incorporating domain knowledge of LLMs: instead of using LLMs to assist sampling, **we attempt to elicit their embedded domain knowledge to explicitly model the complex input–output relationship of the optimization problem.** This perspective targets a fundamental challenge in BBO, the implicitly defined objective function. Improving the quality of observations can be helpful for learning the latent relationship, but it does not fundamentally reduce the dependence on observations. In contrast, if structural information of a BBO problem can be extracted without relying on any target-task observations, it may help address the difficulty of learning the implicit input–output relationship and thereby improve optimization efficiency.

Some studies have explored related directions by using LLMs to predict objective values or construct surrogate models (Song et al., 2024; Requeima et al., 2024; Nguyen et al., 2025; Chang et al., 2025; Yuan et al., 2026). However, such approaches often rely on additional training, task-specific fine-tuning, or a substantial number of evaluations to make the LLM predictive. In contrast, our goal is to elicit useful prior information directly from off-the-shelf pretrained LLMs, without external optimization datasets or model fine-tuning. Moreover, instead of relying on LLMs as implicit in-context predictors of objective values, we aim to construct an explicit probabilistic distribution over the unknown objective function. This distribution can then serve as a task-specific prior for BBO algorithms and be updated as expensive evaluations become available.

A key technical challenge is that directly eliciting an explicit probabilistic distribution over a complex objective function from an LLM is difficult. Our key idea is to elicit this distribution indirectly through pairwise preferences: relative comparisons between candidate solutions provide a natural interface for expressing LLM domain knowledge and collectively reveal structure of the underlying objective landscape. This shifts the problem from directly specifying a function-space distribution to a more tractable question: how can LLM knowledge be systematically converted into informative pairwise preferences? To address this challenge, we propose an **agentic preference-based prior elicitation framework**. We introduce a preference elicitation agent that analyzes the problem context and autonomously constructs a task-specific preference-generation mechanism without access to objective evaluations. Bayesian preference learning (Chu & Ghahramani, 2005) then converts pairwise preferences into an explicit probabilistic distribution over the unknown objective function. Our contributions are summarized as follows:

- **A prior-elicitation paradigm for sample-efficient BBO.** We introduce a new paradigm that elicits a problem-specific probabilistic distribution of the unknown objective from off-the-shelf LLMs and uses it as an informative prior for Bayesian optimization. This provides a new perspective on incorporating LLM domain knowledge into BBO. We characterize the conditions under which the elicited prior tightens finite-budget GP-LCB simple-regret bounds, and empirically demonstrate its advantages in low-budget regimes across six real-world benchmarks. We further show that, under the stated GP-LCB conditions, even an initially biased prior does not prevent asymptotic convergence in simple regret.
- **Function-level probabilistic elicitation from LLMs.** We propose an **agentic preference-based prior elicitation framework** that converts LLM domain knowledge into an explicit probabilistic model of the unknown objective. Instead of treating the LLM as a direct comparator, a preference elicitation agent autonomously constructs a task-specific mechanism for generating informative pairwise evidence, which is then aggregated into a Gaussian-process prior. To the best of our knowledge, this is the first study to explicitly elicit a probabilistic distribution over an unknown objective function from LLMs for BBO. We further establish controlled sensitivity of the resulting distribution to corrupted preferences.
- **Empirical validation of reliable LLM prior elicitation.** We evaluate the proposed framework on six real-world BBO benchmarks spanning traffic control, EV charging, water-network operation, and materials screening. The elicited prior provides strong early-stage optimization performance under tight evaluation budgets while remaining competitive as more objective evaluations become available. Beyond optimization performance,

our preference- and distribution-level analyses show that the framework extracts informative pairwise structure from pretrained LLMs and converts it into a probabilistic model that captures meaningful ranking information and promising regions of the objective landscape.

2 Problem Setup

Consider a black-box minimization problem over a compact feasible decision space $\mathcal{X} \subseteq \mathbb{R}^d$:

$$x^\star \in \arg\min_{x \in \mathcal{X}} f(x), \tag{1}$$

where $f : \mathcal{X} \to \mathbb{R}$ is an unknown objective function. We assume that f is continuous, so that a global minimizer exists. The function f is treated as a black box: its analytical form, gradients, and structural properties are unavailable to the optimizer. Given x, the objective can only be accessed through direct evaluations: the optimizer does not observe $f(x)$ exactly. Instead, querying a candidate solution $x_t \in \mathcal{X}$ returns a noisy observation $z_t = f(x_t) + \epsilon_t$, where $\epsilon_t \sim \mathcal{N}(0, \sigma_n^2)$ is independent Gaussian evaluation noise.

The main difficulty in this setting is that each evaluation can be computationally expensive, which limits the number of queries available to the optimizer. Therefore, the optimizer cannot rely on dense exploration of the search space, but must instead infer a useful surrogate of the underlying objective from limited observations.

3 Method

We present an LLM-empowered framework for sample-efficient BBO. As illustrated in Figure 1, the framework consists of three stages: agentic preference elicitation, sparse preference learning, and prior-informed Bayesian optimization. First, a preference elicitation agent analyzes the problem context, formulates a task-specific preference-generation mechanism, and operationalizes it through an executable comparison procedure. LLM domain knowledge is embedded into the procedure, and it generates pairwise preferences over candidate solutions without querying the expensive objective function. Second, Bayesian preference learning converts the generated preferences into a probabilistic distribution over the unknown objective function, characterized by a preference-informed mean and covariance. Finally, this distribution is used to construct an informative Gaussian-process prior for Bayesian optimization, which is subsequently updated using a limited number of direct objective evaluations. Detailed methodological formulations are provided in Appendix B.

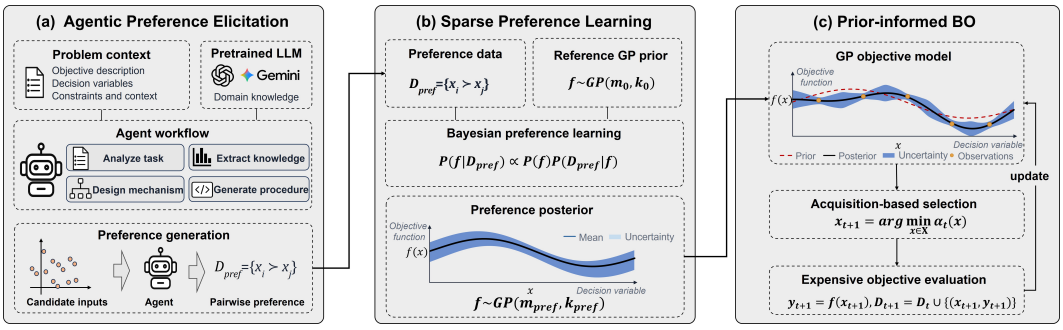

Figure 1: Overview of the proposed framework. Preferences are elicited from an LLM agent and converted into an explicit probabilistic distribution, which is then incorporated into Bayesian optimization.

3.1 Agentic Preference Elicitation

We design a *preference elicitation agent* to transform domain knowledge encoded in an LLM into an executable procedure for comparing candidate solutions. The agent is implemented as a coding

agent that combines LLM reasoning, tool use, and iterative refinement. Given a BBO task, the agent is required to produce a procedure to compare a set of candidate pairs $\mathcal{Q} = \{(x_i, x_j)\}_{i \neq j}$.

The same agent design is used across BBO tasks, while task-specific behavior is determined by the provided problem context. Specifically, for each task, the problem context contains: 1) the optimization specification, including the objective, decision variables, constraints, and optimization direction; and 2) static scenario information available before optimization, such as network structure and operating conditions. Candidate-level objective values, evaluation histories, and the executable black-box evaluator are not accessible to the agent.

Given the problem context, the agent first formalizes the optimization task and identifies how the decision variables affect the underlying system. It then uses the available scenario information and domain knowledge to formulate plausible comparison mechanisms and, based on them, constructs an executable, task-specific procedure for comparing candidate solutions. The procedure is further checked and refined through lightweight validation. The resulting preference procedure is applied to each queried pair $(x_i, x_j) \in \mathcal{Q}$. For each pair (x_i, x_j), $s_{i,j} \in \{-1, 0, +1\}$ denotes the elicited preference: $+1$ favors x_i, -1 favors x_j, and 0 denotes an indistinguishable pair due to insufficient information or a negligible estimated difference. The elicitation stage therefore yields

$$\mathcal{D}_{\text{pref}} = \{(x_i, x_j, s_{i,j}) : (x_i, x_j) \in \mathcal{Q}\}, \tag{2}$$

which serves as the preference information used by the subsequent Bayesian preference learning.

3.2 Sparse Preference Learning

We convert the elicited preferences $\mathcal{D}_{\text{pref}}$ into a probabilistic distribution over f using Gaussian-process preference learning (Chu & Ghahramani, 2005). Starting from the reference prior $f \sim \mathcal{GP}(0, k_0)$, let $\mathcal{E} = \{(i, j) : (x_i, x_j) \in \mathcal{Q},\ s_{i,j} \in \{-1, +1\}\}$ denote the retained comparisons. We use the probit likelihood

$$p(\mathcal{D}_{\text{pref}} \mid f) = \prod_{(i,j)\in\mathcal{E}} \Phi\left(\frac{s_{i,j}[f(x_j) - f(x_i)]}{\tau}\right), \tag{3}$$

where $\Phi(\cdot)$ is the standard normal CDF and $\tau > 0$ controls preference noise.

To scale inference to many comparisons, we operate in an inducing space. Let $\mathcal{S}$ be the distinct candidates in $\mathcal{Q}$ and $\mathcal{Z} \subseteq \mathcal{S}$, with $|\mathcal{Z}| = m \ll |\mathcal{S}|$, be the inducing set. Define $K_{ZZ} = k_0(\mathcal{Z}, \mathcal{Z}) = LL^\top$ and $u = f(\mathcal{Z})$. Under the reference prior, $u \sim \mathcal{N}(0, K_{ZZ})$, so the whitened variable $v = L^{-1}u$ satisfies $v \sim \mathcal{N}(0, I_m)$. With $\phi(x) = L^{-1}k_0(\mathcal{Z}, x)$, we have $f(x) = \phi(x)^\top v + r(x)$ and $k_r(x, x') = k_0(x, x') - \phi(x)^\top\phi(x')$, where $r(\cdot)$ is an independent zero-mean residual GP. Approximating each pairwise difference by its inducing component, $f(x_j) - f(x_i) \approx d_{i,j}^\top v$ with $d_{i,j} = \phi(x_j) - \phi(x_i)$, gives $p(\mathcal{D}_{\text{pref}} \mid v) \approx \prod_{(i,j)\in\mathcal{E}} \Phi\left(\frac{s_{i,j} d_{i,j}^\top v}{\tau}\right)$. Thus, all retained comparisons contribute to learning while inference is performed over the m-dimensional variable v.

Because the probit posterior is non-Gaussian, we use a diagonal Laplace approximation $q(v) = \mathcal{N}(\hat{v}, D^{-1})$, where $\hat{v}$ is the posterior mode and D is the local diagonal precision; details are given in Appendix B. Combining this approximation with the residual GP yields $f|\mathcal{D}_{\text{pref}} \sim \mathcal{GP}(m_{\text{pref}}, k_{\text{pref}})$, where

$$\begin{aligned} m_{\text{pref}}(x) &= \phi(x)^\top \hat{v}, \\ k_{\text{pref}}(x, x') &= k_0(x, x') - \phi(x)^\top\phi(x') + \phi(x)^\top D^{-1}\phi(x'). \end{aligned} \tag{4}$$

3.3 Prior-informed Bayesian Optimization

The elicited GP distribution is used as the prior for Bayesian optimization. We first collect a common set of n_0 initial objective evaluations, $\mathcal{D}_{\text{ini}} = \{(x_j, z_j)\}_{j=1}^{n_0}$, where $z_j = f(x_j) + \epsilon_j$. Conditioning the preference-informed prior on these initial observations yields an initialized posterior, from which subsequent optimization proceeds.

More generally, after $t \geq n_0$ evaluations, let $\mathcal{D}_t = \{(x_j, z_j)\}_{j=1}^t$, $X_t = (x_1, \ldots, x_t)$, $\mathbf{z}_t = (z_1, \ldots, z_t)^\top$, and $K_t = k_{\text{pref}}(X_t, X_t)$. Conditioning the elicited prior on $\mathcal{D}_t$ gives

$$\begin{aligned} \mu_t(x) &= m_{\text{pref}}(x) + k_{\text{pref}}(x, X_t)\left(K_t + \sigma_n^2 I\right)^{-1}\left(\mathbf{z}_t - m_{\text{pref}}(X_t)\right), \\ \sigma_t^2(x) &= k_{\text{pref}}(x, x) - k_{\text{pref}}(x, X_t)\left(K_t + \sigma_n^2 I\right)^{-1} k_{\text{pref}}(X_t, x). \end{aligned} \tag{5}$$

At $t = n_0$, this posterior serves as the starting point for the subsequent optimization. The next candidate is selected by optimizing an acquisition function, $x_{t+1} \in \arg\min_{x \in \mathcal{X}} \alpha_t(x; \mu_t, \sigma_t)$. The new observation $z_{t+1} = f(x_{t+1}) + \epsilon_{t+1}$ is then added to $\mathcal{D}_t$, and the posterior is updated sequentially. In this way, the elicited preferences provide prior structural information about the objective, while the initial and subsequent objective evaluations progressively refine this information through Bayesian updating.

4 Theoretical Analysis

In this section, we address three theoretical questions: 1) when the elicited distribution can improve BO sample efficiency; 2) how preference errors affect the resulting distribution; and 3) whether a biased distribution can persistently mislead optimization. Throughout, $\mathcal{H}_k$ denotes the RKHS induced by kernel k, and we adopt Assumption 1. The sample-efficiency and asymptotic results are established for GP-LCB, while the preference-sensitivity result is acquisition-independent. Proofs are provided in Appendix C.

Assumption 1 (Fixed-prior RKHS setting)**.** $\mathcal{X}$ is nonempty and compact; k_0 is continuous, $K_{ZZ} \succ 0$ with $m \geq 1$, and $\sup_{x \in \mathcal{X}} k_0(x, x) \leq \kappa^2 < \infty$. Moreover, $f \in \mathcal{H}_{k_0}$ with $\|f\|_{\mathcal{H}_{k_0}} \leq B_0$, and all prior-construction choices are fixed before direct evaluations.

We compare the reference prior $\mathcal{GP}(0, k_0)$ and the elicited prior $\mathcal{GP}(m_{\text{pref}}, k_{\text{pref}})$ under the same n_0 initial evaluations $\mathcal{D}_{\text{ini}} = \{(x_i, z_i)\}_{i=1}^{n_0}$. Conditioning on $\mathcal{D}_{\text{ini}}$ gives $\mathcal{GP}(\mu_0^\circ, k_0^\circ)$ and $\mathcal{GP}(\mu_{\text{pref}}^\circ, k_{\text{pref}}^\circ)$. For $j \in \{0, \text{pref}\}$, define $\bar{v}_j = L^{-1}\mu_j^\circ(\mathcal{Z})$, $V_j = L^{-1}k_j^\circ(\mathcal{Z}, \mathcal{Z})L^{-\top}$, and $v_f = L^{-1}f(\mathcal{Z})$. Let $\gamma_N(k)$ denote the maximum information gain from N evaluations under kernel k (Appendix C.1). The following theorem gives a sufficient alignment condition under which the elicited prior yields a no-larger finite-budget GP-LCB simple-regret bound.

Theorem 1. *Under Assumption 1, suppose that*

$$(v_f - \bar{v}_{\text{pref}})^\top V_{\text{pref}}^{-1}(v_f - \bar{v}_{\text{pref}}) \leq q(v_f - \bar{v}_0)^\top V_0^{-1}(v_f - \bar{v}_0), \qquad q \in [0, 1). \tag{6}$$

Consider exact GP-LCB after the shared initialization, using the confidence coefficients β_t^j specified in Appendix C.1, constructed for both runs from the same fixed finite upper bound on $\|f - \mu_0^\circ\|_{\mathcal{H}_{k_0^\circ}}$. For any $\delta \in (0, 1)$, conditional on the elicited prior and $\mathcal{D}_{\text{ini}}$, with probability at least $1 - \delta$ over subsequent evaluation noise, simultaneously for both runs and all $N \geq 1$, we have $S_{n_0+N}^j \leq U_j(N)$ and $U_{\text{pref}}(N) \leq U_0(N)$, $j \in \{0, \text{pref}\}$, where $U_j(N) = \sqrt{\frac{C_{\sigma,\kappa}\,\beta_N^j\,\gamma_N(k_j^\circ)}{N}}$. Here $S_{n_0+N}^j = \min_{1 \leq t \leq n_0+N}[f(x_t^j) - f(x^\star)]$, N counts evaluations after initialization, and $C_{\sigma,\kappa} > 0$ is the common kernel–noise constant defined in Appendix C.1.

Theorem 1 shows that, under the same initialization, a preference-informed posterior yields a tighter finite-budget regret bound when it reduces the precision-weighted residual complexity. Because Eq. equation 6 depends on the unknown values $f(\mathcal{Z})$, it is a sufficient oracle condition rather than an operational criterion and is not used in constructing or tuning the method. Appendix E.2 therefore provides complementary post-hoc diagnostics of the empirical alignment between the preference-informed posterior and the true objective.

We next examine the effect of preference errors. Let $\tilde{s}, s \in \{-1, +1\}^{|\mathcal{E}|}$ denote reference and corrupted preference-label vectors, respectively, and define $E = \{(i, j) \in \mathcal{E} : s_{i,j} \neq \tilde{s}_{i,j}\}$. Let $\tilde{m}_{\text{pref}}$ and m_{pref} be the prior means obtained from their respective exact MAP estimates. We measure the resulting perturbation using the centered RMSE over $\mathcal{S}$.

Proposition 1. *Fix the comparison design $(\mathcal{Q}, \mathcal{E})$, inducing points $\mathcal{Z}$, reference kernel k_0, and preference scale $\tau > 0$. For any pair of label assignments $\tilde{s}$ and s, $\Delta_{\mathcal{S}} :=$*

$\frac{1}{\sqrt{|\mathcal{S}|}} \left\| P_\perp \left[m_{\mathrm{pref}}(\mathcal{S}) - \tilde{m}_{\mathrm{pref}}(\mathcal{S}) \right] \right\|_2 \leq C\sqrt{|E|}$, *where* $P_\perp = I_{|\mathcal{S}|} - |\mathcal{S}|^{-1}\mathbf{1}\mathbf{1}^\top$ *is the centering matrix. The constant* $C < \infty$ *depends only on the fixed comparison design and prior-construction parameters (see Appendix C).*

Proposition 1 establishes that the learned prior mean is robust to corrupted preference data: even when multiple comparison labels are incorrect, its centered deviation grows at most on the order of $\sqrt{|E|}$. A corresponding covariance-sensitivity analysis is provided in Appendix C.2.

The elicited mean may nevertheless be biased; we therefore examine whether a biased preference prior can asymptotically affect the accuracy of the BO algorithm.

Proposition 2. *Under Assumption 1 and the stated observation model with* $\sigma_n > 0$*, assume that* $\gamma_N(k_0) = o(\sqrt{N})$*. Consider exact GP-LCB initialized with* $\mathcal{GP}(\mu^\circ_{\mathrm{pref}}, k^\circ_{\mathrm{pref}})$*, using the confidence coefficients specified in Appendix C.3, based on a fixed finite upper bound on* $\|f - \mu^\circ_{\mathrm{pref}}\|_{\mathcal{H}_{k^\circ_{\mathrm{pref}}}}$*. For any* $\delta \in (0,1)$*, conditional on the elicited prior and* $\mathcal{D}_{\mathrm{ini}}$*, with probability at least* $1-\delta$ *over subsequent evaluation noise,*

$$S^{\mathrm{pref}}_{n_0+N} \longrightarrow 0 \qquad \text{as } N \to \infty. \tag{7}$$

No alignment condition is required.

Proposition 2 shows that initial prior bias may affect finite-budget performance, but does not prevent the best queried objective value from approaching the global minimum under the stated conditions. The information-gain condition $\gamma_N(k_0) = o(\sqrt{N})$ holds for several widely used kernels, including linear and squared-exponential kernels. (Srinivas et al., 2010).

5 Experiments

The experiments answer three research questions (RQs). **RQ1**: Does the elicited distribution improve efficiency of BBO? **RQ2**: Can the proposed framework generate informative preferences? **RQ3**: Does the elicited distribution capture the objective landscape?

5.1 Experimental Setup

Benchmarks We evaluate our method on six real-world BBO benchmarks spanning traffic control, EV charging, water-network operation, and materials screening. *TSC-C3* and *TSC-C8* optimize 11- and 25-dimensional signal timings on the RESCO Cologne3 and Cologne8 networks (Ault & Sharon, 2021); *EVC-Caltech* and *EVC-JPL* optimize 12-dimensional charging policies in SustainGym (Yeh et al., 2023). *Net3* optimizes a 24-dimensional pump schedule using (Klise et al., 2017), while *COF* selects among 608 materials represented by 14-dimensional descriptors for Xe/Kr separation (Gantzler et al., 2023). TSC-C3 and TSC-C8 use fixed search pools of 5,000 pre-evaluated feasible plans, and COF uses 608 released high-fidelity evaluations; EVC-Caltech, EVC-JPL, and Net3 are optimized over continuous search space. See detailed settings in Appendix D.

Baselines We compare against standard BBO methods—random search, vanilla GP-BO with GP-LCB and EI, SMAC (Hutter et al., 2011), and TuRBO (Eriksson et al., 2019)—and LLM-enhanced BO methods, including CAKE with BAKER (hereafter referred to as CAKE) (Suwandi et al., 2026) and LGBO (Yuan et al., 2026). All methods share the same 10 initial points for each seed, use 200 additional evaluations, and are repeated over 20 seeds. Our method uses GP-LCB by default; EI results are reported in Appendix E.3.

Preference data For each task, we generate 30,000 pairwise preferences from randomly sampled candidate pairs. Only comparisons assigned a non-zero preference label by the agent are retained.

LLM configuration We instantiate the preference elicitation agent using the Codex CLI coding-agent environment, with `deepseek-v4-flash` as the underlying LLM.

Metrics For benchmarks with a finite candidate pool (TSC-C3, TSC-C8, and COF), we report simple regret $r_t = f(x_t^{\mathrm{best}}) - f^\star$, where $x_t^{\mathrm{best}} = \arg\min_{i \leq t} f(x_i)$ and $f^\star$ is the best objective value in the candidate pool. For benchmarks with continuous search spaces (EVC-Caltech, EVC-JPL, and Net3), we report the best-so-far objective value.

5.2 RQ1: Does the Elicited Distribution Improve Efficiency of Black-Box Optimization?

Figure 2 shows the mean best-so-far trajectories over 20 runs, while Table 1 reports the average rank across six benchmarks. Our method improves rapidly within the first 20 evaluations and achieves the lowest average rank at every budget, with the largest advantage in the low-budget regime; this advantage narrows as more observations accumulate but persists through 200 evaluations. Notably, these gains require only 0.52M LLM tokens, lower than the other LLM-based baselines, demonstrating both sample and token efficiency. For each benchmark, the detailed mean and standard deviation at each evaluation budget are reported in Appendix E.1.

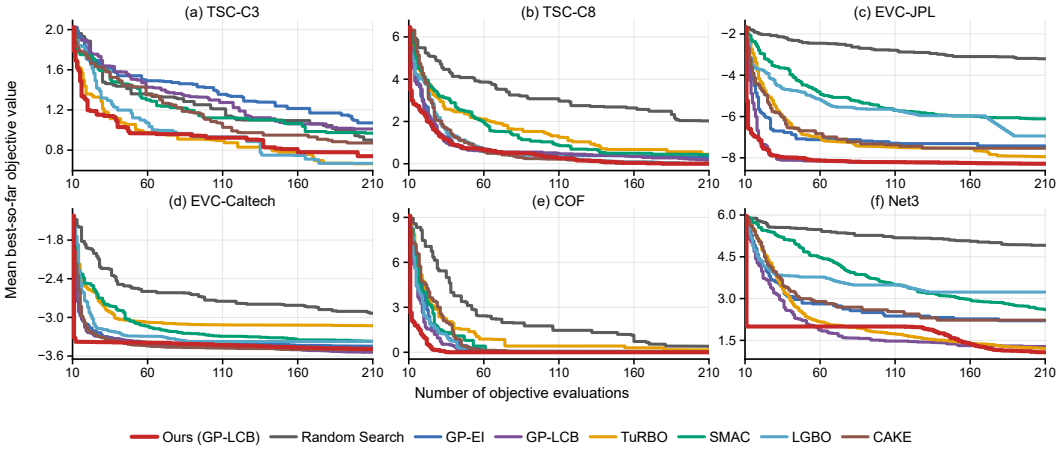

Figure 2: Optimization trajectories across six benchmarks. Curves show mean best-so-far objective values over 20 runs; lower is better. The x-axis counts total objective evaluations, including initialization, highlighting the early-stage advantage of our method.

Table 1: Average rank across six benchmarks at different evaluation budgets (after initialization).

Method	Rank@10	Rank@20	Rank@50	Rank@100	Rank@200	Token Usage ($\times 10^6$)
Random Search	8.00	7.33	7.67	7.67	7.50	–
Vanilla BO (GP-LCB)	2.67	3.17	2.50	3.17	3.33	–
Vanilla BO (EI)	4.00	4.00	4.50	5.17	5.33	–
SMAC	6.33	6.33	6.00	5.50	5.67	–
TuRBO	5.00	5.00	5.00	4.50	4.50	–
LGBO	4.00	4.33	4.67	4.50	4.08	16.92
CAKE	4.83	4.83	3.50	3.00	3.50	0.61
Ours (GP-LCB)	**1.17**	**1.00**	**2.17**	**2.50**	**2.08**	**0.52**

5.3 RQ2: Can the Proposed Framework Generate Informative Preferences?

Table 2 evaluates the generated preferences, where *coverage* is the fraction of pairs receiving a definite preference ($s_{i,j} \in \{-1, +1\}$), and accuracy measures agreement with the ground-truth ordering. Across all six benchmarks, preference accuracy exceeds 60%, indicating that the proposed framework can extract task-relevant preference information from pretrained LLMs across heterogeneous domains. Coverage varies across tasks, reflecting a task-dependent trade-off between generating more preferences and abstaining on uncertain comparisons. The results confirm that the framework produces informative and selective preference signals for prior construction.

Figure 3 visualizes preference quality over the objective-value percentiles of candidate pairs. The heatmap color denotes smoothed preference accuracy, while white points indicate abstentions where the agent cannot reliably distinguish the pair. The results show that preference quality is highly structured across the objective landscape: clearly separated candidates are generally easier to distinguish, whereas ambiguous regions exhibit more abstentions.

Table 2: Accuracy of LLM-generated preference on six benchmarks. Q1–Q4 denote quartiles of normalized decision-space distance.

Benchmark	Coverage	Overall Acc.	Accuracy by decision-space distance			
			Q1 (near)	**Q2**	**Q3**	**Q4** (far)
TSC-C3 (11D)	80.8%	60.7%	61.7%	60.9%	60.3%	59.9%
TSC-C8 (25D)	69.3%	65.7%	66.7%	66.2%	65.4%	64.5%
EVC-Caltech (12D)	82.7%	99.8%	99.7%	99.8%	99.7%	99.9%
EVC JPL (12D)	61.9%	99.8%	99.9%	99.8%	99.9%	99.6%
COF (14D)	82.4%	79.0%	72.5%	77.2%	81.5%	83.5%
Net3 (24D)	85.1%	92.3%	92.2%	91.9%	92.4%	92.8%

5.4 RQ3: Does the Elicited Distribution Capture the Objective Landscape?

We assess the elicited distribution from two perspectives: pairwise ranking and promising-region identification. For pairwise ranking, we randomly sample 1,000 candidate pairs, and we report the fraction correctly ordered with at least 95% posterior confidence; for promising-region identification, we minimize the posterior mean and evaluate the selected solution by its true regret. Compared with the zero-GP prior and GP posteriors fitted with 50, 100, and 200 observations, as shown in Table 3, the elicited posterior achieves the highest Correct@95% across all six benchmarks and identifies competitive, often best, solutions without using target-task observations. These results show that it captures both global ranking structure and promising regions of the objective landscape.

Table 3: Objective-landscape characterization by predictive distributions. "Correct@95%" denotes the fraction of 1,000 sampled pairs that are correctly ordered with posterior confidence above 95%. "Obj." evaluates the posterior-mean minimizer; lower is better.

TSC-C3 (11D)

Predictive distribution	Correct@95% ↑	Obj. ↓
Preference posterior	**53.20**	**0.59**
Zero-GP prior	–	7.21
GP posterior (50)	0.10	1.70
GP posterior (100)	0.10	1.55
GP posterior (200)	0.20	1.32

TSC-C8 (25D)

Predictive distribution	Correct@95% ↑	Obj. ↓
Preference posterior	**51.60**	**2.93**
Zero-GP prior	–	11.07
GP posterior (50)	33.80	7.15
GP posterior (100)	46.10	7.15
GP posterior (200)	42.30	6.93

EVC-JPL (12D)

Predictive distribution	Correct@95% ↑	Obj. ↓
Preference posterior	**77.70**	-2.87
Zero-GP prior	–	0.17
GP posterior (50)	19.40	-2.93
GP posterior (100)	29.70	-2.93
GP posterior (200)	41.20	**-3.27**

EVC-Caltech (12D)

Predictive distribution	Correct@95% ↑	Obj. ↓
Preference posterior	**81.90**	**-2.74**
Zero-GP prior	–	-0.10
GP posterior (50)	21.20	-2.53
GP posterior (100)	33.10	-2.53
GP posterior (200)	56.10	-2.53

COF (14D)

Predictive distribution	Correct@95% ↑	Obj. ↓
Preference posterior	**70.30**	4.52
Zero-GP prior	–	16.84
GP posterior (50)	3.00	7.63
GP posterior (100)	2.60	**1.68**
GP posterior (200)	31.00	**1.68**

Net3 (24D)

Predictive distribution	Correct@95% ↑	Obj. ↓
Preference posterior	**83.80**	3.09
Zero-GP prior	–	16.00
GP posterior (50)	0.00	1.78
GP posterior (100)	0.20	1.74
GP posterior (200)	0.50	**1.04**

We further conduct an ablation study comparing the full elicited distribution $\mathcal{GP}(m_{\text{pref}}, k_{\text{pref}})$ with zero GP prior $\mathcal{GP}(0, k_0)$, mean only prior $\mathcal{GP}(m_{\text{pref}}, k_0)$, and covariance-only prior $\mathcal{GP}(0, k_{\text{pref}})$ under the same BO procedure and evaluation budget. Table 4 shows that the full distribution generally achieves the strongest early-stage performance, while ablating either component degrades optimization quality. The mean-only variant is typically more competitive than the covariance-only

variant, suggesting that the elicited mean provides the primary guidance, with covariance adaptation offering complementary benefits.

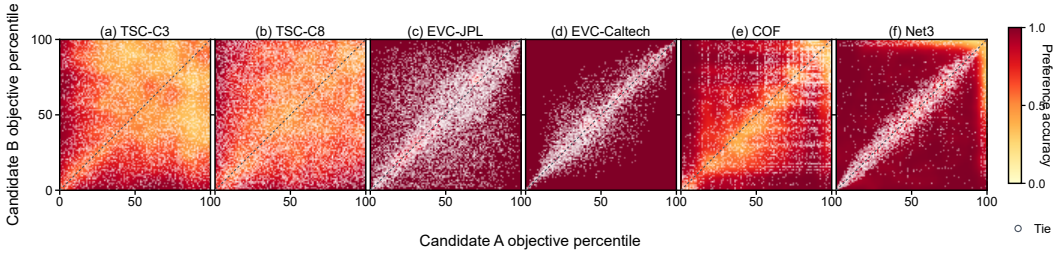

Figure 3: Preference accuracy across six benchmarks. Axes show candidate objective percentiles; color indicates smoothed preference accuracy, and circles denote ties.

Table 4: Ablation of the elicited distribution, comparing the full method with mean-only, covariance-only, and zero-GP variants. Results are mean ± standard deviation; lower is better.

Benchmark	**Variant**	**Obj.@10**	**Obj.@50**	**Obj.@100**	**Obj.@200**
TSC-C3 (11D)	Elicited distribution	1.20 ± 0.31	0.97 ± 0.29	0.92 ± 0.28	0.74 ± 0.33
	Mean only	1.44 ± 0.42	1.16 ± 0.27	1.08 ± 0.26	0.85 ± 0.32
	Covariance only	1.78 ± 0.57	1.48 ± 0.49	1.24 ± 0.34	1.06 ± 0.31
	Zero GP prior	1.87 ± 0.63	1.43 ± 0.46	1.27 ± 0.34	1.01 ± 0.32
TSC-C8 (25D)	Elicited distribution	2.35 ± 0.58	0.66 ± 0.27	0.26 ± 0.25	0.00 ± 0.00
	Mean only	1.72 ± 0.95	0.43 ± 0.00	0.43 ± 0.00	0.13 ± 0.20
	Covariance only	3.22 ± 1.92	0.88 ± 0.57	0.45 ± 0.08	0.19 ± 0.22
	Zero GP prior	2.58 ± 1.86	0.57 ± 0.28	0.47 ± 0.11	0.17 ± 0.21
EVC-JPL (12D)	Elicited distribution	-7.15 ± 0.81	-8.14 ± 0.10	-8.21 ± 0.03	-8.29 ± 0.10
	Mean only	-7.64 ± 0.47	-8.11 ± 0.14	-8.21 ± 0.07	-8.31 ± 0.11
	Covariance only	-5.68 ± 1.91	-8.16 ± 0.14	-8.21 ± 0.00	-8.23 ± 0.06
	Zero GP prior	-7.25 ± 1.41	-8.15 ± 0.14	-8.23 ± 0.12	-8.28 ± 0.11
EVC-Caltech (12D)	Elicited distribution	-3.38 ± 0.03	-3.39 ± 0.02	-3.43 ± 0.05	-3.49 ± 0.03
	Mean only	-3.38 ± 0.02	-3.39 ± 0.01	-3.41 ± 0.04	-3.48 ± 0.04
	Covariance only	-3.29 ± 0.17	-3.39 ± 0.00	-3.39 ± 0.00	-3.39 ± 0.00
	Zero GP prior	-3.24 ± 0.20	-3.42 ± 0.04	-3.46 ± 0.04	-3.54 ± 0.01
COF (14D)	Elicited distribution	0.99 ± 1.05	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
	Mean only	1.90 ± 1.30	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
	Covariance only	1.26 ± 2.62	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
	Zero GP prior	2.01 ± 1.86	0.04 ± 0.11	0.00 ± 0.00	0.00 ± 0.00
Net3 (24D)	Elicited distribution	2.00 ± 0.19	2.00 ± 0.19	2.00 ± 0.19	1.07 ± 0.07
	Mean only	2.76 ± 0.17	2.76 ± 0.17	2.43 ± 0.26	1.61 ± 0.14
	Covariance only	5.92 ± 0.73	4.22 ± 0.73	3.34 ± 0.35	2.58 ± 0.37
	Zero GP prior	4.15 ± 1.45	1.87 ± 0.43	1.47 ± 0.33	1.28 ± 0.24

6 Conclusion

We introduced an agentic preference-elicited Bayesian optimization framework that uses off-the-shelf LLMs to generate pairwise preferences and convert them into an explicit GP prior over the unknown objective. Theoretically, we characterize sufficient conditions under which this prior tightens finite-budget GP-LCB regret bounds. Across six real-world benchmarks, the framework consistently improves early-stage optimization under limited evaluation budgets. These results highlight LLMs as a promising source of structured prior information for sample-efficient BBO.

Broader impacts and limitations. This work may reduce the number of costly simulations or experiments needed in expensive BBO tasks. However, LLM-elicited preferences can be biased and costly to obtain. Future work should improve preference reliability, reduce token cost, and study safeguards for high-stakes decision-making settings.

AI USE STATEMENT

In this work, we used generative AI tools as part of the proposed methodology to elicit pairwise preferences and construct task-specific preference-generation procedures. We also used generative AI tools to assist with code development, language polishing, and improving the clarity of manuscript presentation. We did not use generative AI to generate experimental results, perform statistical analyses, or make final scientific judgments. All AI-assisted code and outputs used in the experiments were inspected and validated by the authors, and all AI-assisted manuscript content was manually reviewed and revised. We take full responsibility for the final content of this work, including all text, claims, code, and experimental results produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

We provide detailed descriptions of the proposed framework, including the preference elicitation agent, sparse preference learning, and prior-informed Bayesian optimization, in Appendix B. Complete proofs of the theoretical results and all associated assumptions are provided in Appendix C. The experimental appendix reports the benchmark definitions, decision spaces, objective functions, data and simulator sources, baseline configurations, initialization protocols, evaluation budgets, and random-seed settings. We also provide the agent instructions and preference-generation procedure used in our experiments to facilitate reproduction of the elicitation stage. Code, simulation environments, and data are publicly available at: `https://anonymous.4open.science/r/LLMs-as-Prior-Elicitors-fo-Black-Box-Optimization-C7D7/`

REFERENCES

Raul Astudillo, Zhiyuan Jerry Lin, Eytan Bakshy, and Peter Frazier. qEUBO: A decision-theoretic acquisition function for preferential bayesian optimization. In *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics*, volume 206 of *Proceedings of Machine Learning Research*, pp. 1093–1114. PMLR, 2023. URL `https://proceedings.mlr.press/v206/astudillo23a.html`.

James Ault and Guni Sharon. Reinforcement learning benchmarks for traffic signal control. In *Proceedings of the Thirty-fifth Conference on Neural Information Processing Systems (NeurIPS 2021) Datasets and Benchmarks Track*, December 2021.

David Eric Austin, Anton Korikov, Armin Toroghi, and Scott Sanner. Bayesian optimization with LLM-based acquisition functions for natural language preference elicitation. In *Proceedings of the 18th ACM Conference on Recommender Systems*, Bari, Italy, 2024. URL `https://arxiv.org/abs/2405.00981`.

James Bergstra and Yoshua Bengio. Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13:281–305, 2012.

Theodore Brown, Alexandru Cioba, and Ilija Bogunovic. Sample-efficient Bayesian optimisation using known invariances. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-1519. URL `https://proceedings.neurips.cc/paper_files/paper/2024/hash/55aeba84b402008d3ed10440d906b4e1-Abstract-Conference.html`.

Paul Brunzema, Mikkel Jordahn, John Willes, Sebastian Trimpe, Jasper Snoek, and James Harrison. Bayesian optimization via continual variational last layer training. In *International Conference on Learning Representations*, 2025. URL `https://proceedings.iclr.cc/paper_files/paper/2025/hash/8df705957a5262de3cb37ba9f1fb96f3-Abstract-Conference.html`.

Alexander Capstick, Rahul Krishnan, and Payam Barnaghi. AutoElicit: Using large language models for expert prior elicitation in predictive modelling. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 6746–6777. PMLR, 2025.

C. Y. Chang, M. Azvar, C. Okwudire, and R. A. Kontar. LLINBO: Trustworthy LLM-in-the-loop Bayesian optimization. *arXiv preprint arXiv:2505.14756*, 2025.

Wei Chu and Zoubin Ghahramani. Preference learning with gaussian processes. In *Proceedings of the 22nd International Conference on Machine Learning*, pp. 137–144, 2005.

David Eriksson, Michael Pearce, Jacob Gardner, Ryan D. Turner, and Matthias Poloczek. Scalable global optimization via local bayesian optimization. In *Advances in Neural Information Processing Systems*, volume 32, 2019. URL https://papers.nips.cc/paper/8788-scalable-global-optimization-via-local-bayesian-optimization.

Nickolas Gantzler, Aryan Deshwal, Janardhan Rao Doppa, and Cory M Simon. Multi-fidelity bayesian optimization of covalent organic frameworks for xenon/krypton separations. *Digital Discovery*, 2(6):1937–1956, 2023.

Javier Gonz'alez, Zhenwen Dai, Andreas Damianou, and Neil D. Lawrence. Preferential bayesian optimization. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1282–1291. PMLR, 2017. URL https://proceedings.mlr.press/v70/gonzalez17a.html.

Henry Gouk and Boyan Gao. Automated prior elicitation from large language models for bayesian logistic regression. In *Proceedings of the 3rd International Conference on Automated Machine Learning*, pp. 1–10, 2024. URL https://openreview.net/forum?id=euLzlnU7gz.

Yongchao Huang. LLM-Prior: A framework for knowledge-driven prior elicitation and aggregation. *arXiv preprint arXiv:2508.03766*, 2025.

Frank Hutter, Holger H. Hoos, and Kevin Leyton-Brown. Sequential model-based optimization for general algorithm configuration. In *Proceedings of the 5th International Conference on Learning and Intelligent Optimization*, pp. 507–523. Springer, 2011.

Carl Hvarfner, Erik Orm Hellsten, and Luigi Nardi. Vanilla Bayesian optimization performs great in high dimensions. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 20793–20817. PMLR, 2024. URL https://proceedings.mlr.press/v235/hvarfner24a.html.

Aya Kayal, Sattar Vakili, Laura Toni, Da-Shan Shiu, and Alberto Bernacchia. Bayesian optimization from human feedback: Near-optimal regret bounds. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 29449–29468. PMLR, 2025. URL https://proceedings.mlr.press/v267/kayal25a.html.

Katherine A Klise, Michael Bynum, Dylan Moriarty, and Regan Murray. A software framework for assessing the resilience of drinking water systems to disasters with an example earthquake case study. *Environmental modelling software*, 95:420–431, 2017.

Katarzyna Kobalczyk, Zhiyuan Jerry Lin, Benjamin Letham, Zhuokai Zhao, Maximilian Balandat, and Eytan Bakshy. LILO: Bayesian optimization with interactive natural language feedback. *arXiv preprint arXiv:2510.17671*, 2025. URL https://arxiv.org/abs/2510.17671.

Daniel Krajzewicz, Georg Hertkorn, Christian R"ossel, and Peter Wagner. Sumo (simulation of urban mobility)-an open-source traffic simulation. In *Proceedings of the 4th middle East Symposium on Simulation and Modelling (MESM20002)*, pp. 183–187, 2002.

Zachary J Lee, Tongxin Li, and Steven H Low. Acn-data: Analysis and applications of an open ev charging dataset. In *Proceedings of the tenth ACM international conference on future energy systems*, pp. 139–149, 2019.

Belinda Z. Li, William Chen, Pratyusha Sharma, and Jacob Andreas. LAMPP: Language models as probabilistic priors for perception and action. *arXiv preprint arXiv:2302.02801*, 2023.

Tennison Liu, Nicol'as Astorga, Nabeel Seedat, and Mihaela van der Schaar. Large language models to enhance Bayesian optimization. In *The Twelfth International Conference on Learning Representations*, 2024.

Satvik Mehul Mashkaria, Siddarth Krishnamoorthy, and Aditya Grover. Generative pretraining for black-box optimization. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 24173–24197, 2023.

Natalie Maus, Kyurae Kim, Geoff Pleiss, David Eriksson, John P. Cunningham, and Jacob R. Gardner. Approximation-aware Bayesian optimization. In *Advances in Neural Information Processing Systems*, volume 37, 2024a. doi: 10.52202/079017-0665. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/257b9a6a0e3856735d0e624e38fb6803-Abstract-Conference.html.

Natalie Maus, Zhiyuan Jerry Lin, Maximilian Balandat, and Eytan Bakshy. Joint composite latent space Bayesian optimization. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 35163–35180. PMLR, 2024b. URL https://proceedings.mlr.press/v235/maus24a.html.

Petrus Mikkola, Osvaldo A. Martin, Suyog Chandramouli, Marcelo Hartmann, Oriol Abril Pla, Owen Thomas, Henri Pesonen, Jukka Corander, Aki Vehtari, Samuel Kaski, Paul-Christian B"urkner, and Arto Klami. Prior knowledge elicitation: The past, present, and future. *Bayesian Analysis*, 19(4):1129–1161, 2024. doi: 10.1214/23-BA1381.

Tung Nguyen and Aditya Grover. LICO: Large language models for in-context molecular optimization. In *International Conference on Learning Representations*, volume 2025, pp. 53576–53597, 2025.

Tung Nguyen, Qiuyi Zhang, Bangding Yang, Chansoo Lee, J"org Bornschein, Yingjie Miao, Sagi Perel, Yutian Chen, and Xingyou Song. Language model embeddings can be sufficient for bayesian optimization. *arXiv preprint arXiv:2410.10190*, 2025.

Jiarong Pan, Stefan Falkner, Felix Berkenkamp, and Joaquin Vanschoren. MALIBO: Meta-learning for likelihood-free Bayesian optimization. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 39102–39134. PMLR, 2024. URL https://proceedings.mlr.press/v235/pan24b.html.

Valerio Perrone, Rodolphe Jenatton, Matthias W. Seeger, and C'edric Archambeau. Scalable hyperparameter transfer learning. In *Advances in Neural Information Processing Systems*, volume 31, 2018. URL https://papers.nips.cc/paper/7917-scalable-hyperparameter-transfer-learning.

Bojana Rankovi'c and Philippe Schwaller. GOLLuM: Gaussian process optimized LLMs – reframing LLM finetuning through bayesian optimization. In *ICLR 2025 Workshop on World Models: Understanding, Modelling and Scaling*, April 2025. URL https://arxiv.org/abs/2504.06265.

James Requeima, John Bronskill, Dami Choi, Richard E. Turner, and David Duvenaud. LLM processes: Numerical predictive distributions conditioned on natural language. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-3479.

Sara Rezaeimanesh and Mohammad M. Ghassemi. What LLMs know but don't say: Non-generative prior extraction for generalization. *arXiv preprint arXiv:2601.17609*, 2026.

Aliaksandra Shysheya, John F. Bronskill, James Requeima, Shoaib Ahmed Siddiqui, Javier Gonzalez, David Duvenaud, and Richard E. Turner. JoLT: Joint probabilistic predictions on tabular data using LLMs. In *AABI 2025 Workshop*, 2025. URL https://openreview.net/forum?id=lvYO6p0QGZ.

Jasper Snoek, Hugo Larochelle, and Ryan Prescott Adams. Practical bayesian optimization of machine learning algorithms. In *Advances in Neural Information Processing Systems*, volume 25, pp. 2960–2968, 2012.

X. Song, O. Li, C. Lee, B. Yang, D. Peng, S. Perel, and Y. Chen. Omnipred: Language models as universal regressors. *Transactions on Machine Learning Research*, 2024.

Niranjan Srinivas, Andreas Krause, Sham M. Kakade, and Matthias Seeger. Gaussian process optimization in the bandit setting: No regret and experimental design. In *Proceedings of the 27th International Conference on Machine Learning*, 2010.

Richard Suwandi, Feng Yin, Juntao Wang, Renjie Li, Tsung-Hui Chang, and Sergios Theodoridis. Adaptive kernel design for bayesian optimization is a piece of cake with llms. *Advances in Neural Information Processing Systems*, 38:132690–132723, 2026.

Kevin Swersky, Jasper Snoek, and Ryan P. Adams. Multi-task Bayesian optimization. In *Advances in Neural Information Processing Systems*, volume 26, 2013. URL https://papers.nips.cc/paper/5086-multi-task-bayesian-optimization.

R. X. Tan, M. Chen, K. Xue, Y. Wang, Y. Wang, S. Fu, and C. Qian. Towards universal offline black-box optimization via learning language model embeddings. *arXiv preprint arXiv:2506.07109*, 2025.

Sattar Vakili, Kia Khezeli, and Victor Picheny. On information gain and regret bounds in gaussian process bandits. In *Proceedings of the 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pp. 82–90. PMLR, 2021. URL https://proceedings.mlr.press/v130/vakili21a.html.

Michael Volpp, Lukas P. Fr"ohlich, Kirsten Fischer, Andreas Doerr, Stefan Falkner, Frank Hutter, and Christian Daniel. Meta-learning acquisition functions for transfer learning in Bayesian optimization. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=ryeYpJSKwr.

Zi Wang and Stefanie Jegelka. Max-value entropy search for efficient bayesian optimization. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 3627–3635. PMLR, 2017. URL https://proceedings.mlr.press/v70/wang17e.html.

Yunyue Wei, Vincent Zhuang, Saraswati Soedarmadji, and Yanan Sui. Scalable Bayesian optimization via focalized sparse gaussian processes. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-3828. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/da30215ee52c1daaaaddada8137cfd0b-Abstract-Conference.html.

Wenjie Xu, Wenbin Wang, Yuning Jiang, Bratislav Svetozarevic, and Colin Jones. Principled preferential bayesian optimization. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 55305–55336. PMLR, 2024. URL https://proceedings.mlr.press/v235/xu24y.html.

Christopher Yeh, Victor Li, Rajeev Datta, Julio Arroyo, Nicolas Christianson, Chi Zhang, Yize Chen, Mehdi Hosseini, Azarang Golmohammadi, Yuanyuan Shi, Yisong Yue, and Adam Wierman. SustainGym: Reinforcement learning environments for sustainable energy systems. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, New Orleans, LA, USA, 12 2023. URL https://openreview.net/forum?id=vZ9tA3o3hr.

Yuxuan Yin, Yu Wang, and Peng Li. High-dimensional Bayesian optimization via semi-supervised learning with optimized unlabeled data sampling. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 57085–57100. PMLR, 2024. URL https://proceedings.mlr.press/v235/yin24d.html.

Xinzhe Yuan, Zhuo Chen, Jianshu Zhang, Huan Xiong, Nanyang Ye, Yuqiang Li, and Qinying Gu. Unleashing llms in bayesian optimization: Preference-guided framework for scientific discovery. In *The Fourteenth International Conference on Learning Representations*, 2026.

Taeyoung Yun, Kiyoung Om, Jaewoo Lee, Sujin Yun, and Jinkyoo Park. Posterior inference with diffusion models for high-dimensional black-box optimization. *arXiv preprint arXiv:2502.16824*, 2025.

Yimeng Zeng, Natalie Maus, Haydn Thomas Jones, Jeffrey Tao, Fangping Wan, Marcelo Der Torossian Torres, Cesar de la Fuente-Nunez, Ryan Marcus, Osbert Bastani, and Jacob R. Gardner. Large scale multi-task bayesian optimization with large language models. *CoRR*, abs/2503.08131, 2025.

Xinyu Zhang, Daolang Huang, Samuel Kaski, and Julien Martinelli. PABBO: Preferential amortized black-box optimization. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=YhfrKB3Ah7.

A Related Work

Sample-efficient black-box optimization. Existing methods for sample-efficient black-box optimization can be roughly grouped into single-task and cross-task approaches. Single-task methods optimize a target problem mainly from observations collected on that problem itself. This category includes model-free search strategies such as random search (Bergstra & Bengio, 2012), surrogate-based optimization methods such as SMAC and Bayesian optimization (Hutter et al., 2011; Snoek et al., 2012), and recent inverse or generative approaches that propose promising designs from observed input–output data (Mashkaria et al., 2023; Yun et al., 2025). Among them, Bayesian optimization (BO) is one of the most widely used frameworks for expensive black-box objectives, as it combines a probabilistic surrogate with an acquisition function to balance exploration and exploitation (Srinivas et al., 2010; Wang & Jegelka, 2017). Recent single-task BO studies further improve sample efficiency by adapting to high-dimensional search spaces, using auxiliary unlabeled samples, learning structured latent representations, exploiting known invariances, or developing scalable and approximation-aware surrogates (Eriksson et al., 2019; Hvarfner et al., 2024; Yin et al., 2024; Maus et al., 2024b; Brown et al., 2024; Wei et al., 2024; Maus et al., 2024a; Brunzema et al., 2025). Despite these advances, single-task methods are still fundamentally observation-driven: their search behavior depends on the quality and coverage of target-task evaluations, which can be difficult to obtain under tight evaluation budgets.

Cross-task methods instead seek to transfer information from related optimization problems. Representative approaches include multi-task Bayesian optimization, scalable transfer-learning surrogates, meta-learned acquisition functions, and amortized optimization policies (Swersky et al., 2013; Perrone et al., 2018; Volpp et al., 2020; Pan et al., 2024; Zhang et al., 2025). These methods can accelerate early-stage optimization when informative historical tasks or meta-training distributions are available. However, their effectiveness depends on the relevance, scale, and quality of these auxiliary sources, and may degrade when the target task differs substantially from the available training tasks.

Large language models for black-box optimization. Recent studies have explored several ways of using large language models (LLMs) for black-box optimization (BBO). One representative line uses LLMs as candidate generators or search-guidance modules, exploiting their ability to model structured sequences and semantic patterns to warm-start optimization, recommend promising search regions, propose or revise candidate solutions, and adaptively design search components such as kernels (Liu et al., 2024; Chang et al., 2025; Suwandi et al., 2026). However, they mainly use LLMs to shape the search policy, while the implicit objective, which is the key challenge for BBO, is still learned from evaluated samples (e.g., using surrogate models). LLMs improves the quality of queried samples, but the optimization process remains observation-driven, and its dependence on observations is only indirectly reduced. This potentially limits the ability of existing LLM-assisted BBO methods to address the sample-efficiency bottleneck under tight evaluation budgets.

Another line uses LLMs as surrogate modeling or representation learning tools. These methods can be further divided into two groups. The first group relies on multi-task optimization data to map heterogeneous inputs into a shared latent space and learn transferable input–objective relationships for downstream optimization, typically through sequence-level modeling (Mashkaria et al., 2023; Zeng et al., 2025) or embedding-based regression (Nguyen & Grover, 2025; Song et al., 2024; Tan et al., 2025). Although these methods are more amenable to cross-task transfer, their effectiveness depends on multi-task optimization data to learn useful inductive biases, and the resulting representations are not necessarily aligned with the local numerical structure of a new task, which can limit extrapolation accuracy and optimization stability. The second group builds task-specific LLM-based

predictors or surrogates from observations collected on the target task, either by conditioning LLMs on observed input–output pairs or by adapting LLM-based representations for Gaussian-process surrogate modeling (Liu et al., 2024; Rankovi'c & Schwaller, 2025; Yuan et al., 2026). However, the effectiveness of these methods still depends on the availability and quality of observations, and their advantages can be limited when the evaluation budget is severely constrained. In this line, the most closely related work is Yuan et al. (2026), which also incorporates LLM-derived preferences into Bayesian optimization. Our framework differs in two fundamental aspects. First, the two methods construct different probabilistic objects from LLM knowledge. LGBO uses LLMs to propose promising points or regions, and translates this information into region-lifted updates that shift the surrogate mean at each BO iteration. In contrast, we use pairwise preferences between candidate solutions to infer an explicit function-space distribution, $\mathcal{GP}(m_{\mathrm{pref}}, k_{\mathrm{pref}})$, whose mean and covariance jointly encode the elicited knowledge and which subsequently serves as the prior for BO. Second, the preference-elicitation mechanisms are different. LGBO directly queries the LLM during optimization for preferred points or regions based on the evolving optimization context, whereas our method employs a coding agent to autonomously construct an executable, task-specific preference generator before target-task evaluations are collected. The resulting generator produces pairwise preferences over candidate solutions, which are then aggregated probabilistically rather than directly steering individual BO iterations.

In this study, we aim to directly elicit domain knowledge from LLMs and to explicitly characterize the objective function without any multi-task optimization data or target-task evaluations.

Large language models for prior elicitation and probabilistic prediction. Prior specification is a central part of Bayesian modeling, yet eliciting informative priors remains difficult because it often requires translating domain knowledge into valid probability distributions (Mikkola et al., 2024). This has motivated recent attempts to use large language models (LLMs) as scalable sources of prior knowledge. For example, LLMPrior formalizes an operator that maps unstructured context to a valid distribution by asking an LLM to specify the parameters of a tractable generative model (Huang, 2025). Related work instantiates this idea in more task-specific settings. LAMPP uses language models to parameterize priors over labels, decisions, and model parameters in probabilistic graphical models (Li et al., 2023). For tabular prediction, Gouk and Gao (Gouk & Gao, 2024) construct priors for Bayesian logistic regression from LLM-generated synthetic data; AutoElicit extracts priors for interpretable predictive models through natural-language interaction (Capstick et al., 2025); and LoID derives Gaussian priors over linear or logistic regression coefficients from token-level logits (Rezaeimanesh & Ghassemi, 2026). These methods demonstrate the value of LLM-derived priors, but the elicited distributions are mainly defined over finite-dimensional parameters, labels, decisions, or structured latent variables. A second line of work uses LLMs as probabilistic predictors. LLM Processes elicits numerical predictive distributions conditioned on data and natural language, while JoLT uses in-context learning to define joint predictive distributions for tabular data (Requeima et al., 2024; Shysheya et al., 2025). Although these methods provide uncertainty estimates, their probabilistic objects are predictive distributions over specified query points or finite target sets. They do not construct an explicit distribution over the underlying function that can be reused as a prior in downstream Bayesian updating. In contrast, we elicit pairwise preferences from the LLM and convert them into a GP-compatible function-space distribution over the black-box objective. This distribution has an explicit mean and covariance, initializes the BO surrogate, and is subsequently refined by direct objective observations.

Preferential Bayesian optimization. Preferential Bayesian optimization (PBO) studies black-box optimization when scalar objective values are unavailable and the learner can only access pairwise comparisons or other preference feedback. A common formulation places a Gaussian process prior over a latent utility function and connects pairwise outcomes to utility differences through Bradley–Terry–Luce or Thurstone–Mosteller likelihoods (Chu & Ghahramani, 2005; Gonz'alez et al., 2017). Recent work has further developed decision-theoretic acquisition functions, such as qEUBO, and principled regret analyses for preference-based feedback (Astudillo et al., 2023; Xu et al., 2024; Kayal et al., 2025). These studies mainly target settings where preferences are the primary observation channel, often reflecting a human decision maker's subjective or latent utility.

Our setting is different. We consider standard black-box optimization problems whose objective values are objectively evaluable but expensive to obtain. LLM-generated preferences are not used

as the online feedback signal to replace objective evaluations; instead, they are elicited before optimization and converted into a probabilistic prior over the objective function. This design is also distinct from recent LLM-assisted BO methods that use LLMs for warm-starting, candidate generation, or acquisition guidance (Liu et al., 2024; Austin et al., 2024; Kobalczyk et al., 2025). In our framework, the role of the LLM is to provide task-specific structural preference information without requiring online human annotations, while subsequent Bayesian optimization is still driven by objective observations from the target black-box function.

B Technique Details of the Proposed Method

This section provides implementation details for the proposed method.

B.1 Preference Elicitation Agent

The preference elicitation agent is designed to transform domain knowledge encoded in an LLM into a task-specific mechanism for comparing candidate solutions. Rather than directly asking the LLM to judge each candidate pair, we instantiate the agent as a coding agent that can inspect task information, reason about the underlying system, construct executable analysis procedures, and refine them based on intermediate results. The same agent instructions are used across all tasks; task-specific behavior arises from the problem and scenario information provided to the agent.

Agent inputs. The agent operates entirely on information available before any target-task objective evaluations are collected. Its input contains three main components. First, the *optimization specification* describes the objective, optimization direction, decision variables, their physical meanings, constraints, and their mappings to the underlying system. Second, the *scenario description* provides static information characterizing the specific problem instance, such as network structure, demand conditions, operating settings, controller configurations, or other physical parameters. Third, the agent receives the candidate solutions and the candidate pairs to be compared.

The agent may use lightweight computational tools to parse these inputs, construct derived quantities, and execute its own analysis code. However, it has no access to candidate objective values, previous optimization histories, pairwise ground-truth labels, or the executable black-box evaluator. The original simulator is also unavailable during preference elicitation. This separation is important: the resulting preferences encode information elicited from the LLM and the pre-evaluation problem context, rather than information learned from expensive objective evaluations.

Autonomous preference-model construction. Given these inputs, the agent follows a structured workflow to construct a preference generator. It first formalizes the optimization problem and identifies how the decision variables affect the underlying system. It then analyzes the available scenario information and identifies physical, operational, or behavioral mechanisms that are plausibly related to the objective. Based on these mechanisms, the agent designs a task-specific procedure for comparing candidate solutions and implements the procedure as executable analysis code.

Importantly, we do not prescribe a fixed form for this procedure. Depending on the task, the agent may construct physically motivated indicators, analytical approximations, surrogate scores, rule-based comparisons, or combinations of multiple simplified models. Nor do we require the agent to predict numerical objective values. The only requirement is that the resulting procedure provides a reproducible basis for comparing candidate solutions. This flexibility allows the same agent architecture to adapt to heterogeneous BBO problems without task-specific prompt engineering.

During this process, the agent is instructed to distinguish information directly provided by the task from quantities derived through computation and assumptions introduced from general domain knowledge. Before finalizing the preference generator, it performs lightweight internal checks, such as examining boundary cases, perturbing key assumptions or parameters, comparing alternative formulations, or testing whether the resulting rankings are stable. If the available evidence is insufficient to support a reliable comparison, the agent is allowed to abstain rather than force a preference.

Preference generation. After the task-specific comparison procedure has been constructed, it is applied to the queried candidate pairs. For each pair $(x_i, x_j) \in \mathcal{Q}$, the agent produces $s_{i,j} \in$

$\{-1, 0, +1\}$, where $+1$ indicates a preference for x_i, -1 indicates a preference for x_j, and 0 denotes an abstention when the two candidates cannot be reliably distinguished from the available evidence.

Instructions for the Preference Elicitation Agent. Instruction for the preference elicitation agent is represented as follows

```
ROLE
You are an autonomous research-oriented preference annotation agent for
rapid scientific validation. Your inputs describe an optimization problem,
static simulator inputs, candidate solutions, and candidate pairs. You must
not apply a predefined scoring formula. Instead, understand the problem,
propose an interpretable estimation method using domain knowledge, implement
and test the method, and generate pairwise preferences according to the
strength of the available evidence.

EVIDENCE TYPES
Always distinguish:
1. Scenario facts explicitly provided by the input files.
2. Derived quantities computed reproducibly from those files.
3. Research assumptions introduced using domain knowledge.
Never describe assumptions or estimated quantities as observed simulator
outcomes.

INPUTS
- input/problem_context.json: objective, direction, decision variables,
  simulator mappings, constraints, fixed settings, assumptions, and limits.
- input/simulation/: static simulator and scenario files.
- input/candidates.csv: complete candidate vectors indexed by
  candidate_index.
- input/pairs.tsv: pair_id, a_candidate_index, and b_candidate_index.
- tools/: optional neutral parsers and candidate-decoding utilities.
  These tools must not provide scores, preferences, or domain conclusions.

WORKFLOW
1. Read the problem context and formalize the objective, variables,
   simulator mappings, constraints, and unavailable information.
2. Inspect the static scenario inputs and extract facts relevant to
   candidate comparison.
3. Recall applicable theories and research methods, but use them only to
   formulate hypotheses rather than scenario-specific observations.
4. Design the simplest defensible estimation method supported by the
   available inputs. Explain its indicators, assumptions, computation,
   alternatives, and failure modes.
5. Write all analysis code to output/analysis/. Analyze all candidates in
   a batch before comparing pairs.
6. Perform at least one lightweight validation, such as a boundary check,
   perturbation test, component analysis, alternative-model comparison, or
   sensitivity analysis. Revise the method or reduce confidence if needed.
7. Compare every pair. Output A or B only when the estimated difference is
   sufficiently clear and stable; otherwise output tie or
   insufficient_information.
8. Check complete pair coverage, duplicate IDs, candidate indices, and
   consistency among preference, label, confidence, and weight.

REQUIRED ANALYSIS ARTIFACTS
Write the following files under output/analysis/:
- simulation_summary.json: scenario facts extracted from static inputs.
- candidate_plans.json: mapping from candidates to simulator objects.
- method.md: domain understanding, considered methods, selected method,
  facts, derived quantities, assumptions, validation, and limitations.
- estimate_preferences.py: executable analysis and comparison procedure.

FINAL OUTPUT
Write exactly one JSON object for each input pair to
output/preferences.jsonl:
{"pair_id":"pair_0","a_candidate_index":0,
 "b_candidate_index":1,"preference":"A","label":1,
 "confidence":0.7,"weight":0.6,
 "justification":["Pair-specific quantitative evidence and uncertainty."]}

A requires label=1 and B requires label=0. tie and
insufficient_information require label=null and weight=0. Confidence and
weight must lie in [0,1], and a hard preference must satisfy
weight <= confidence. Justification must be a non-empty string list.

WORKSPACE RULES
Treat input/ and tools/ as read-only. Do not access evaluation data,
historical outputs, previous skills, run logs, or files outside the workspace.
Do not use the Internet or launch the black-box simulator. Do not fabricate
```

```
simulator observations. Write final preferences only to
output/preferences.jsonl and analysis artifacts only to output/analysis/.
Keep the method minimal, reproducible, and focused on the current task.
```

B.2 SPARSE PREFERENCE LEARNING

We detail the sparse construction of the preference-induced distribution.We retain only hard comparisons $\mathcal{E} = \{(i,j) : s_{ij} \in \{-1,+1\}\}$; abstentions do not contribute to preference learning.

Let $\mathcal{S} = \{u_1, \dots, u_n\}$ denote the candidate support set used to represent the preference prior, and let $\mathcal{Z} = \{z_1, \dots, z_m\} \subseteq \mathcal{S}$ be a set of inducing points, where $m \ll n$ when the support set is large. We use an RBF kernel,

$$k_0(x, x') = \exp\left(-\frac{\|\tilde{x} - \tilde{x}'\|_2^2}{2\ell^2}\right),$$

where ℓ is set to the median nonzero pairwise distance among sampled support candidates.

Let $K_{\mathcal{Z}\mathcal{Z}} = LL^\top$ be the Cholesky factorization of the inducing-point kernel matrix, and define the whitened inducing variable

$$\mathbf{v} = L^{-1} f(\mathcal{Z}), \qquad \mathbf{v} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_m).$$

For any $x \in \mathcal{X}$, define

$$\boldsymbol{\phi}(x) = L^{-1} k_0(\mathcal{Z}, x).$$

The GP can then be decomposed as

$$f(x) = \boldsymbol{\phi}(x)^\top \mathbf{v} + r(x),$$

where the residual process has covariance

$$k_r(x, x') = k_0(x, x') - \boldsymbol{\phi}(x)^\top \boldsymbol{\phi}(x').$$

For a comparison $(i,j) \in \mathcal{E}$, let

$$\mathbf{d}_{ij} = \boldsymbol{\phi}(x_j) - \boldsymbol{\phi}(x_i).$$

Because the objective is minimized, the event $s_{ij}\mathbf{d}_{ij}^\top \mathbf{v} > 0$ agrees with the elicited preference. We therefore use the sparse probit likelihood

$$p(\mathcal{D}_{\text{pref}} \mid \mathbf{v}) \approx \prod_{(i,j)\in\mathcal{E}} \Phi\left(\frac{s_{ij}\mathbf{d}_{ij}^\top \mathbf{v}}{\tau}\right),$$

where $\tau > 0$ controls the assumed preference noise.

Let $\psi(z) = -\log \Phi(z)$. The latent preference representation is obtained from the MAP estimate

$$\hat{\mathbf{v}} = \arg\min_{\mathbf{v}\in\mathbb{R}^m} \left\{ \frac{1}{2}\|\mathbf{v}\|_2^2 + \sum_{(i,j)\in\mathcal{E}} \psi\left(\frac{s_{ij}\mathbf{d}_{ij}^\top \mathbf{v}}{\tau}\right) \right\}.$$

This optimization is performed with mini-batches over the retained comparisons, allowing the method to scale with the number of preference pairs without constructing an $n \times n$ kernel matrix.

We use a diagonal Laplace approximation around $\hat{\mathbf{v}}$. Define

$$q_{ij} = \frac{s_{ij}\mathbf{d}_{ij}^\top \hat{\mathbf{v}}}{\tau}, \qquad \rho(q) = \frac{\varphi(q)}{\Phi(q)}, \qquad \omega_{ij} = \frac{\rho(q_{ij})\big[q_{ij} + \rho(q_{ij})\big]}{\tau^2},$$

where $\varphi(\cdot)$ is the standard normal density. The diagonal precision approximation is

$$D = \mathbf{I}_m + \operatorname{Diag}\left(\sum_{(i,j)\in\mathcal{E}} \omega_{ij}\big(\mathbf{d}_{ij} \odot \mathbf{d}_{ij}\big)\right),$$

which yields

$$q(\mathbf{v} \mid \mathcal{D}_{\text{pref}}) \approx \mathcal{N}(\hat{\mathbf{v}}, D^{-1}).$$

Finally, marginalizing this Gaussian approximation induces a GP distribution over the full search space:

$$m_{\text{pref}}(x) = \boldsymbol{\phi}(x)^\top \hat{\mathbf{v}},$$
$$k_{\text{pref}}(x, x') = k_0(x, x') - \boldsymbol{\phi}(x)^\top \boldsymbol{\phi}(x') + \boldsymbol{\phi}(x)^\top D^{-1} \boldsymbol{\phi}(x').$$

Thus, the elicited distribution is $f \sim \mathcal{GP}(m_{\text{pref}}, k_{\text{pref}})$, which provides the prior mean and covariance subsequently updated by direct black-box evaluations during Bayesian optimization.

B.3 Prior-informed Bayesian Optimization

The preference-learning stage provides the pre-evaluation distribution $\mathcal{GP}(m_{\text{pref}}, k_{\text{pref}})$, which is used as the prior for Bayesian optimization. Before sequential optimization, we first collect a common set of n_0 initial objective evaluations,

$$\mathcal{D}_{\text{ini}} = \{(x_i, z_i)\}_{i=1}^{n_0}, \qquad z_i = f(x_i) + \epsilon_i,$$

where $\epsilon_i \sim \mathcal{N}(0, \sigma_n^2)$. Let $X_{\text{ini}} = (x_1, \ldots, x_{n_0})$, $\mathbf{z}_{\text{ini}} = (z_1, \ldots, z_{n_0})^\top$, and $K_{\text{ini}} = k_{\text{pref}}(X_{\text{ini}}, X_{\text{ini}})$. Conditioning the elicited prior on these initial observations gives

$$\mu^\circ_{\text{pref}}(x) = m_{\text{pref}}(x) + k_{\text{pref}}(x, X_{\text{ini}}) \left(K_{\text{ini}} + \sigma_n^2 I\right)^{-1} \left[\mathbf{z}_{\text{ini}} - m_{\text{pref}}(X_{\text{ini}})\right],$$
$$k^\circ_{\text{pref}}(x, x') = k_{\text{pref}}(x, x') - k_{\text{pref}}(x, X_{\text{ini}}) \left(K_{\text{ini}} + \sigma_n^2 I\right)^{-1} k_{\text{pref}}(X_{\text{ini}}, x').$$

The resulting $\mathcal{GP}(\mu^\circ_{\text{pref}}, k^\circ_{\text{pref}})$ serves as the initialized surrogate for subsequent Bayesian optimization.

More generally, after $t \geq n_0$ objective evaluations, let

$$\mathcal{D}_t = \{(x_i, z_i)\}_{i=1}^{t}, \qquad X_t = (x_1, \ldots, x_t), \qquad \mathbf{z}_t = (z_1, \ldots, z_t)^\top,$$

and $K_t = k_{\text{pref}}(X_t, X_t)$. Equivalently, conditioning the original elicited prior on all observations collected so far gives

$$\mu_t(x) = m_{\text{pref}}(x) + k_{\text{pref}}(x, X_t) \left(K_t + \sigma_n^2 I\right)^{-1} \left[\mathbf{z}_t - m_{\text{pref}}(X_t)\right],$$
$$\sigma_t^2(x) = k_{\text{pref}}(x, x) - k_{\text{pref}}(x, X_t) \left(K_t + \sigma_n^2 I\right)^{-1} k_{\text{pref}}(X_t, x).$$

At $t = n_0$, these expressions recover the initialized posterior above; subsequent observations progressively refine the preference-informed distribution through standard Bayesian updating.

In the main experiments, we use GP-LCB as the acquisition function. For the minimization problem, the next solution is selected as

$$x_{t+1} \in \arg\min_{x \in \mathcal{X}} \left[\mu_t(x) - \sqrt{\beta^{\text{exp}}_{t-n_0+1}}\, \sigma_t(x)\right],$$

where β^{exp}_s follows the experimental GP-LCB schedule specified in Appendix D.2. After evaluating x_{t+1}, the observation $z_{t+1} = f(x_{t+1}) + \epsilon_{t+1}$ is added to $\mathcal{D}_t$, and the posterior is updated.

C Proofs for Section 4

We prove the three main-text results in order under Assumption 1. Throughout, f is fixed, not a sample from either GP. The elicited prior is constructed as in Eq. (4) from finitely many comparisons; no transitivity or probabilistic correctness assumption is imposed on their labels. Observations have independent noise $\epsilon_t \sim \mathcal{N}(0, \sigma_n^2)$ with $\sigma_n > 0$, and GP updates use this same fixed variance. Queries are selected before their noises are observed. The optimization results are conditional on the pre-evaluation information and the shared initial data $\mathcal{D}_{\text{ini}}$; all subsequent probability statements concern future evaluation noise.

For a kernel k, define

$$\gamma_N(k) = \frac{1}{2} \sup_{X_N \in \mathcal{X}^N} \log\det\left(I_N + \sigma_n^{-2} k(X_N, X_N)\right), \qquad \gamma_0(k) = 0. \tag{8}$$

Repeated inputs are allowed. Kernel inequalities denote Loewner order on every finite Gram matrix. Write $X_{\text{ini}} = (x_1, \ldots, x_{n_0})$ and $\mathbf{z}_{\text{ini}} = (z_1, \ldots, z_{n_0})^\top$. The index t in a post-initialization confidence coefficient counts additional evaluations, whereas $\mu^j_{n_0+t}$ and $\sigma^j_{n_0+t}$ use all $n_0 + t$ observations of run j.

C.1 Proof of Theorem 1

We first establish the prior geometry, then its conditional analogue after initialization, and finally the GP-LCB regret comparison.

Lemma 1 (Prior geometry). The projected probit objective has a unique finite mode $\hat{v}$, and its diagonal precision satisfies $I_m \preceq D \preceq d_{\max} I_m$ for some finite $d_{\max} = \lambda_{\max}(D)$. Moreover,

$$0 \preceq k_r \preceq k_{\mathrm{pref}} \preceq k_0, \qquad d_{\max}^{-1} k_0 \preceq k_{\mathrm{pref}}. \tag{9}$$

The RKHSs $\mathcal{H}_{k_0}$ and $\mathcal{H}_{k_{\mathrm{pref}}}$ have the same function set. With $f_\perp = f - \phi(\cdot)^\top v_f$,

$$\begin{aligned} \|f\|^2_{\mathcal{H}_{k_0}} &= \|f_\perp\|^2_{\mathcal{H}_{k_r}} + \|v_f\|_2^2, \\ \|f - m_{\mathrm{pref}}\|^2_{\mathcal{H}_{k_{\mathrm{pref}}}} &= \|f_\perp\|^2_{\mathcal{H}_{k_r}} + (v_f - \hat{v})^\top D (v_f - \hat{v}). \end{aligned} \tag{10}$$

Proof. For $a = (i, j) \in \mathcal{E}$, write $d_a = d_{i,j}$ and $s_a = s_{i,j}$. The negative log posterior is

$$\mathcal{L}_s(v) = \tfrac{1}{2}\|v\|_2^2 + \sum_{a \in \mathcal{E}} \psi(s_a d_a^\top v / \tau), \qquad \psi(z) = -\log \Phi(z). \tag{11}$$

Let $\rho(z) = \varphi(z)/\Phi(z)$, where φ is the standard normal density. Then $\psi'(z) = -\rho(z)$ and $\psi''(z) = \rho(z)[z + \rho(z)] \in (0, 1)$: a standard normal variable truncated above at z has mean $-\rho(z) < z$ and variance $1 - \rho(z)[z + \rho(z)] > 0$. Thus

$$H_s(v) := \nabla^2 \mathcal{L}_s(v) = I_m + \tau^{-2} \sum_{a \in \mathcal{E}} \psi''(s_a d_a^\top v / \tau) d_a d_a^\top \succeq I_m. \tag{12}$$

Since $\mathcal{L}_s(v) \geq \|v\|_2^2/2$, it is coercive and strongly convex. Its unique mode is finite, and $D = \mathrm{diag}(H_s(\hat{v}))$ satisfies $I_m \preceq D \preceq (1 + \tau^{-2} \sum_a \|d_a\|_2^2) I_m$. Here diag retains the diagonal as a diagonal matrix.

For a finite design X, form F_X from the rows $\phi(x)^\top$. The resulting Schur complement satisfies $k_r(X, X) = k_0(X, X) - F_X F_X^\top \succeq 0$. Since $d_{\max}^{-1} I_m \preceq D^{-1} \preceq I_m$,

$$k_{\mathrm{pref}} = k_r + \phi^\top D^{-1} \phi, \qquad k_0 = k_r + \phi^\top \phi$$

imply Eq.(9). Kernel order also implies information-gain order by monotonicity of $\log\det(I + A)$ on positive semidefinite matrices.

Since $L^{-1} K_{ZZ} L^{-\top} = I_m$, the components of ϕ are orthonormal and span the kernel sections at $\mathcal{Z}$. The coefficients of the projection of f onto this span are $v_f = L^{-1} f(\mathcal{Z})$. Its orthogonal complement has kernel k_r and consists of functions vanishing on $\mathcal{Z}$. The finite-rank kernel $\phi(x)^\top D^{-1} \phi(x')$ assigns squared norm $b^\top D b$ to $\phi(\cdot)^\top b$. Since this function has inducing values Lb, the two component spaces intersect only at zero. Their unique direct-sum norms give Eq. (10) and the equality of function sets. □

RKHS norms after noisy conditioning. For any kernel k, let k° be its posterior kernel after the initial design, with noise variance $\lambda = \sigma_n^2$. If $\mathsf{A}h = h(X_{\mathrm{ini}})$ is the evaluation operator in $\mathcal{H}_k$, then $\mathsf{T} = I + \lambda^{-1} \mathsf{A}^* \mathsf{A}$ is bounded and boundedly invertible. The posterior kernel has sections $k^\circ(\cdot, x) = \mathsf{T}^{-1} k(\cdot, x)$, so the inner product $\langle h, g \rangle_{k^\circ} = \langle \mathsf{T} h, g \rangle_k$ has reproducing kernel k°. Hence

$$\mathcal{H}_{k^\circ} = \mathcal{H}_k, \qquad \|h\|^2_{\mathcal{H}_{k^\circ}} = \|h\|^2_{\mathcal{H}_k} + \sigma_n^{-2} \|h(X_{\mathrm{ini}})\|_2^2. \tag{13}$$

This also holds for positive semidefinite kernels such as k_r.

For a prior mean m, put $y = \mathbf{z}_{\mathrm{ini}} - m(X_{\mathrm{ini}})$ and $K = k(X_{\mathrm{ini}}, X_{\mathrm{ini}})$. The posterior mean shift $p = \mu^\circ - m = k(\cdot, X_{\mathrm{ini}})(K + \sigma_n^2 I)^{-1} y$ satisfies

$$\|p\|^2_{\mathcal{H}_{k^\circ}} = \sigma_n^{-2} y^\top K (K + \sigma_n^2 I)^{-1} y \leq \sigma_n^{-2} \|y\|_2^2.$$

Consequently, if $\|f - m\|_{\mathcal{H}_k} \leq B$ and $\sup_x k(x, x) \leq \kappa^2$, then

$$\|f - \mu^\circ\|_{\mathcal{H}_{k^\circ}} \leq B \sqrt{1 + n_0 \kappa^2 / \sigma_n^2} + \sigma_n^{-1} \|\mathbf{z}_{\mathrm{ini}} - m(X_{\mathrm{ini}})\|_2. \tag{14}$$

This bound holds for every realized initial data set.

Lemma 2 (Geometry after shared initialization). The initialized kernels satisfy $k^\circ_{\mathrm{pref}} \preceq k^\circ_0$. Let k°_r be k_r conditioned on the initial design. There is a common residual $h^\circ \in \mathcal{H}_{k^\circ_r}$ such that, for $j \in \{0, \mathrm{pref}\}$,

$$\|f - \mu^\circ_j\|^2_{\mathcal{H}_{k^\circ_j}} = \|h^\circ\|^2_{\mathcal{H}_{k^\circ_r}} + (v_f - \bar{v}_j)^\top V_j^{-1}(v_f - \bar{v}_j). \tag{15}$$

Proof. Let $F = F_{X_{\mathrm{ini}}}$ and define

$$A = k_r(X_{\mathrm{ini}}, X_{\mathrm{ini}}) + \sigma_n^2 I_{n_0}, \qquad \Lambda = F^\top A^{-1} F, \qquad g = F^\top A^{-1}\mathbf{z}_{\mathrm{ini}}.$$

Integrating the common residual out of the likelihood gives the same Gaussian factor $\mathcal{N}(\mathbf{z}_{\mathrm{ini}}; Fv, A)$ for v in both models. Gaussian conditioning therefore gives

$$\begin{aligned} V_0 &= (I_m + \Lambda)^{-1}, \qquad \bar{v}_0 = V_0 g, \\ V_{\mathrm{pref}} &= (D + \Lambda)^{-1}, \quad \bar{v}_{\mathrm{pref}} = V_{\mathrm{pref}}(D\hat{v} + g). \end{aligned} \tag{16}$$

These are model-conditioning identities; the true f remains fixed.

Conditioning additionally on v gives the same conditional process under both models. Its mean and kernel can be expressed using

$$\begin{aligned} c^\circ(x) &= k_r(x, X_{\mathrm{ini}})A^{-1}\mathbf{z}_{\mathrm{ini}}, \\ \phi^\circ(x) &= \phi(x) - F^\top A^{-1} k_r(X_{\mathrm{ini}}, x), \\ k^\circ_r(x, x') &= k_r(x, x') - k_r(x, X_{\mathrm{ini}})A^{-1}k_r(X_{\mathrm{ini}}, x'). \end{aligned} \tag{17}$$

Marginalizing v yields

$$\mu^\circ_j(x) = c^\circ(x) + \phi^\circ(x)^\top \bar{v}_j, \qquad k^\circ_j(x, x') = k^\circ_r(x, x') + \phi^\circ(x)^\top V_j \phi^\circ(x'). \tag{18}$$

Because $D \succeq I_m$, Eq. (16) gives $V_{\mathrm{pref}} \preceq V_0$, proving the kernel order.

Since $k_r(\mathcal{Z}, \cdot) = 0$, the matrix with rows $\phi^\circ(z)^\top$, $z \in \mathcal{Z}$, is L, and $c^\circ(\mathcal{Z}) = 0$. Thus Eq. (16) agrees with the main-text definitions of $\bar{v}_j$ and V_j. Define

$$h^\circ = f - c^\circ - \phi^\circ(\cdot)^\top v_f = f_\perp - k_r(\cdot, X_{\mathrm{ini}})A^{-1}(\mathbf{z}_{\mathrm{ini}} - Fv_f).$$

Equation (13) applied to k_r implies $h^\circ \in \mathcal{H}_{k^\circ_r}$. Every function in this RKHS vanishes on $\mathcal{Z}$, whereas $\phi^\circ(\cdot)^\top b$ has values Lb there. The component spaces in Eq. (18) therefore intersect only at zero; the finite component has squared norm $b^\top V_j^{-1} b$. Their direct-sum norm proves Eq. (15). □

Lemma 3 (Time-uniform confidence bound). For a fixed prior mean $\bar{m}$ and continuous kernel k, suppose $\|f - \bar{m}\|_{\mathcal{H}_k} \le B < \infty$. After t adaptive noisy evaluations, its posterior mean μ_t and latent standard deviation σ_t satisfy, with probability at least $1 - \eta$,

$$|f(x) - \mu_t(x)| \le \left[B + \sqrt{2\left(\gamma_t(k) + \log\frac{1}{\eta}\right)}\right]\sigma_t(x), \qquad x \in \mathcal{X}, \quad t \ge 0, \tag{19}$$

simultaneously, for any $\eta \in (0, 1)$.

Proof. We give the Gaussian-noise specialization of the kernel self-normalized argument. In $\mathcal{H}_k$, put $h = f - \bar{m}$, $k_x = k(\cdot, x)$, $\lambda = \sigma_n^2$, and let $\mathsf{A}_t h = h(X_t)$. Define $K_t = \mathsf{A}_t\mathsf{A}_t^*$, $\mathsf{W}_t = \lambda I + \mathsf{A}_t^*\mathsf{A}_t$, and $w_t = \mathsf{A}_t^*\boldsymbol{\epsilon}_t$, where $\boldsymbol{\epsilon}_t = (\epsilon_1, \ldots, \epsilon_t)^\top$ and A_t^* denotes the adjoint. The posterior identities and Cauchy–Schwarz give

$$\begin{aligned} f(x) - \mu_t(x) &= \lambda\langle k_x, \mathsf{W}_t^{-1} h\rangle - \langle k_x, \mathsf{W}_t^{-1} w_t\rangle, \\ \sigma_t^2(x) &= \lambda\langle k_x, \mathsf{W}_t^{-1} k_x\rangle, \\ |f(x) - \mu_t(x)| &\le \left(B + \lambda^{-1/2}\|w_t\|_{\mathsf{W}_t^{-1}}\right)\sigma_t(x), \end{aligned} \tag{20}$$

where $\|w\|^2_{\mathsf{W}_t^{-1}} = \langle w, \mathsf{W}_t^{-1} w\rangle$.

For completeness, let $k^{(r)}$ be the projection of k onto the first r vectors of a fixed orthonormal RKHS basis. For each fixed function g, predictability and the Gaussian moment-generating function make

$\exp\{\sum_{i=1}^{t} \epsilon_i g(x_i)/\sigma_n - \frac{1}{2}\sum_{i=1}^{t} g(x_i)^2\}$ a nonnegative martingale. Mixing over an independent $g \sim \mathcal{GP}(0, k^{(r)}/\lambda)$ gives

$$M_t^{(r)} = \frac{\exp\Big\{\boldsymbol{\epsilon}_t^\top K_t^{(r)}(K_t^{(r)} + \lambda I_t)^{-1}\boldsymbol{\epsilon}_t/(2\sigma_n^2)\Big\}}{\sqrt{\det(I_t + \lambda^{-1}K_t^{(r)})}}.$$

A continuous kernel on compact $\mathcal{X} \subseteq \mathbb{R}^d$ has a separable RKHS. The projections therefore converge pointwise; conditional Fatou's lemma shows that their limit M_t, given by the same expression with K_t, is a nonnegative supermartingale with $M_0 = 1$. Finite-dimensional RKHSs need no limiting step. Ville's inequality and $\|w_t\|^2_{\mathsf{W}_t^{-1}} = \boldsymbol{\epsilon}_t^\top K_t(K_t + \lambda I_t)^{-1}\boldsymbol{\epsilon}_t$ imply, simultaneously for all t,

$$\|w_t\|^2_{\mathsf{W}_t^{-1}} \le 2\sigma_n^2\left[\tfrac{1}{2}\log\det(I_t + \lambda^{-1}K_t) + \log\frac{1}{\eta}\right] \le 2\sigma_n^2\left[\gamma_t(k) + \log\frac{1}{\eta}\right] \tag{21}$$

with probability at least $1-\eta$. Substitution into Eq. (20) proves the claim, since $\lambda = \sigma_n^2$. The same event controls every x. □

Proof of Theorem 1. Conditional on $\mathcal{D}_{\mathrm{ini}}$, choose a fixed finite $B_{\mathrm{ini}} \ge \|f - \mu_0^\circ\|_{\mathcal{H}_{k_0^\circ}}$. One admissible choice, not requiring unknown objective values, is

$$B_{\mathrm{ini}} = B_0\sqrt{1 + n_0\kappa^2/\sigma_n^2} + \sigma_n^{-1}\|\mathbf{z}_{\mathrm{ini}}\|_2, \tag{22}$$

by Eq. (14). Lemma 2 and Eq. (6) yield

$$\begin{aligned}\|f - \mu_{\mathrm{pref}}^\circ\|^2_{\mathcal{H}_{k_{\mathrm{pref}}^\circ}} &\le \|f - \mu_0^\circ\|^2_{\mathcal{H}_{k_0^\circ}} \\ &\quad - (1-q)(v_f - \bar{v}_0)^\top V_0^{-1}(v_f - \bar{v}_0) \le B_{\mathrm{ini}}^2.\end{aligned} \tag{23}$$

Use the common radius to define

$$\beta_t^j = \left[B_{\mathrm{ini}} + \sqrt{2\left(\gamma_{t-1}(k_j^\circ) + \log\frac{2}{\delta}\right)}\right]^2, \qquad C_{\sigma,\kappa} = \frac{8\kappa^2}{\log(1 + \kappa^2/\sigma_n^2)}. \tag{24}$$

Exact GP-LCB chooses a measurable minimizer

$$x_{n_0+t}^j \in \arg\min_{x\in\mathcal{X}}\left[\mu_{n_0+t-1}^j(x) - \sqrt{\beta_t^j}\,\sigma_{n_0+t-1}^j(x)\right].$$

Continuity and compactness ensure existence; ties are resolved measurably. Sequential conditioning from the initialized GP is identical to updating the original GP with all observations. Lemma 3 with $\eta = \delta/2$ for each run and a union bound gives simultaneous confidence coverage at every subsequent iteration with conditional probability at least $1-\delta$. No independence between the two runs is needed.

On this event, let $r_t^j = f(x_{n_0+t}^j) - f(x^\star)$. LCB selection and confidence coverage at $x^\star$ and $x_{n_0+t}^j$ give $r_t^j \le 2\sqrt{\beta_t^j}\,\sigma_{n_0+t-1}^j(x_{n_0+t}^j)$. For the post-initialization design $Y_N^j = (x_{n_0+1}^j, \dots, x_{n_0+N}^j)$, successive Schur complements give (Srinivas et al., 2010)

$$\sum_{t=1}^{N}\log\Big(1 + \sigma_n^{-2}(\sigma_{n_0+t-1}^j(x_{n_0+t}^j))^2\Big) = \log\det\Big(I_N + \sigma_n^{-2}k_j^\circ(Y_N^j, Y_N^j)\Big) \le 2\gamma_N(k_j^\circ).$$

Since $0 \le (\sigma_{n_0+t-1}^j(x_{n_0+t}^j))^2 \le \kappa^2$, concavity of $\log(1 + u/\sigma_n^2)$ on $[0, \kappa^2]$ implies

$$\sum_{t=1}^{N}(\sigma_{n_0+t-1}^j(x_{n_0+t}^j))^2 \le \frac{2\kappa^2\gamma_N(k_j^\circ)}{\log(1 + \kappa^2/\sigma_n^2)}. \tag{25}$$

Both $\gamma_t(k_j^\circ)$ and β_t^j are nondecreasing. Cauchy–Schwarz therefore yields

$$R_N^j := \sum_{t=1}^{N} r_t^j \le \sqrt{C_{\sigma,\kappa}N\beta_N^j\gamma_N(k_j^\circ)}, \qquad S_{n_0+N}^j \le R_N^j/N \le U_j(N), \tag{26}$$

where $U_j(N)$ is defined in the main text. The simple-regret inequality holds because its minimum also includes the initial points. Finally, $k_{\mathrm{pref}}^\circ \preceq k_0^\circ$ implies $\gamma_N(k_{\mathrm{pref}}^\circ) \le \gamma_N(k_0^\circ)$ and $\beta_N^{\mathrm{pref}} \le \beta_N^0$, proving $U_{\mathrm{pref}}(N) \le U_0(N)$ on the same event. □

C.2 Proof of Proposition 1

The comparison design, support, inducing points, input transformations, noise scale τ, and kernel parameters are fixed, including the abstention pattern. Write $p = |\mathcal{S}| \geq 1$ and $M = |\mathcal{E}|$, and let $\hat{v}, \tilde{v}$ minimize $\mathcal{L}_s, \mathcal{L}_{\tilde{s}}$. Since $\mathcal{L}_s(0) = M \log 2$ and all likelihood terms are nonnegative,

$$\max\{\|\hat{v}\|_2, \|\tilde{v}\|_2\} \leq R_M := \sqrt{2M \log 2}. \tag{27}$$

Let G have rows $d_a^\top$, $a \in \mathcal{E}$, and G_E retain its rows in the corrupted set E. With $\|\cdot\|_{\mathrm{op}}$ denoting the Euclidean operator norm, define

$$\begin{aligned} d_* &= \max_{a \in \mathcal{E}} \|d_a\|_2, & A_M &= d_* R_M / \tau, \\ \alpha_* &= \tau^{-1} \max_{|z| \leq A_M} [\rho(z) + \rho(-z)], & C_v &= \alpha_* \|G\|_{\mathrm{op}}, \\ C &= p^{-1/2} \|P_\perp F_{\mathcal{S}}\|_{\mathrm{op}} C_v. \end{aligned} \tag{28}$$

These constants are finite and independent of the label assignments and the choice of E. For $M = 0$, set $d_* = 0$ and take zero-row matrix norms to be zero; both modes and C vanish.

Proof. For $a \in E$, let

$$\alpha_a = \tau^{-1} \left[s_a \psi'(s_a d_a^\top \tilde{v}/\tau) - \tilde{s}_a \psi'(\tilde{s}_a d_a^\top \tilde{v}/\tau) \right].$$

Because $\tilde{s}_a = -s_a$, $|\alpha_a| = \tau^{-1}[\rho(z_a) + \rho(-z_a)]$, where $z_a = d_a^\top \tilde{v}/\tau$. Equation (27) gives $|z_a| \leq A_M$, hence $\|\alpha_E\|_2 \leq \alpha_* \sqrt{|E|}$ for $\alpha_E = (\alpha_a)_{a \in E}$. The first-order conditions and the integral mean-value identity imply

$$\bar{H}(\hat{v} - \tilde{v}) + G_E^\top \alpha_E = 0, \qquad \bar{H} = \int_0^1 H_s(\tilde{v} + u(\hat{v} - \tilde{v}))\, du \succeq I_m. \tag{29}$$

Thus $\|\hat{v} - \tilde{v}\|_2 \leq \|G_E\|_{\mathrm{op}} \alpha_* \sqrt{|E|} \leq C_v \sqrt{|E|}$. Since the support mean difference equals $F_{\mathcal{S}}(\hat{v} - \tilde{v})$,

$$\Delta_{\mathcal{S}} \leq p^{-1/2} \|P_\perp F_{\mathcal{S}}\|_{\mathrm{op}} \|\hat{v} - \tilde{v}\|_2 \leq C \sqrt{|E|}.$$

Uniqueness handles $E = \varnothing$. □

The argument uses the full Hessian $\bar{H}$, not the diagonal approximation D. It permits dependent or systematic label errors. The constant C may depend on M and the fixed design; the result is not a design-uniform error-rate bound or an optimization-regret comparison.

Covariance sensitivity. Let $\tilde{D}, \tilde{k}_{\mathrm{pref}}$ be obtained from $\tilde{s}$, and set

$$\begin{aligned} L_M &= \max_{|z| \leq A_M} |\psi'''(z)|, \qquad h_2 = \max_{1 \leq r \leq m} \left(\sum_{a \in \mathcal{E}} d_{a,r}^4 \right)^{1/2}, \\ C_D &= \frac{h_2}{\tau^2} \left(1 + \frac{L_M \|G\|_{\mathrm{op}} C_v}{\tau} \right). \end{aligned} \tag{30}$$

Proposition 3 (Supplementary covariance sensitivity)**.** *Under the same fixed-design conditions,*

$$\begin{aligned} \|D^{-1} - \tilde{D}^{-1}\|_{\mathrm{op}} &\leq \min\{1, C_D \sqrt{|E|}\}, \\ \sup_{x, x' \in \mathcal{X}} |k_{\mathrm{pref}}(x, x') - \tilde{k}_{\mathrm{pref}}(x, x')| &\leq \kappa^2 \min\{1, C_D \sqrt{|E|}\}. \end{aligned} \tag{31}$$

Proof. Split each curvature difference into

$$\begin{aligned} u_a &= \psi''(s_a d_a^\top \hat{v}/\tau) - \psi''(s_a d_a^\top \tilde{v}/\tau), \\ w_a &= \psi''(s_a d_a^\top \tilde{v}/\tau) - \psi''(\tilde{s}_a d_a^\top \tilde{v}/\tau). \end{aligned}$$

Let $u = (u_a)_{a \in \mathcal{E}}$ and $w = (w_a)_{a \in \mathcal{E}}$. All arguments lie in $[-A_M, A_M]$, so $\|u\|_2 \leq L_M \|G\|_{\mathrm{op}} \|\hat{v} - \tilde{v}\|_2 / \tau$. Since $w_a = 0$ off E and $0 < \psi'' < 1$, $\|w\|_2 \leq \sqrt{|E|}$. Column-wise Cauchy–Schwarz yields

$$\|D - \tilde{D}\|_{\mathrm{op}} \leq \tau^{-2} h_2 (\|u\|_2 + \|w\|_2) \leq C_D \sqrt{|E|}.$$

The inverse identity and $D, \tilde{D} \succeq I_m$ give the same bound for $D^{-1} - \tilde{D}^{-1}$, whose operator norm is also at most one. Finally, $\|\phi(x)\|_2 \leq \kappa$ and $k_{\mathrm{pref}} - \tilde{k}_{\mathrm{pref}} = \phi^\top (D^{-1} - \tilde{D}^{-1}) \phi$ prove Eq. (31). □

These are moment-sensitivity bounds, not probability-calibration claims; covariance perturbations need not have a fixed sign.

Numerical MAP solutions. For approximate modes $v^{\mathrm{num}}, \tilde{v}^{\mathrm{num}}$, strong convexity bounds their distances to the exact modes by $e_{\mathrm{opt}} = \|\nabla \mathcal{L}_s(v^{\mathrm{num}})\|_2$ and $\tilde{e}_{\mathrm{opt}} = \|\nabla \mathcal{L}_{\tilde{s}}(\tilde{v}^{\mathrm{num}})\|_2$. Consequently,

$$\Delta_{\mathcal{S}}^{\mathrm{num}} \le C\sqrt{|E|} + p^{-1/2}\|P_{\perp} F_{\mathcal{S}}\|_{\mathrm{op}}(e_{\mathrm{opt}} + \tilde{e}_{\mathrm{opt}}).$$

These are gradients of the full objective, not mini-batch gradients. The covariance bound above uses precisions evaluated at exact modes. The sensitivity arguments also apply to any fixed stabilized features.

C.3 Proof of Proposition 2

We first obtain a finite residual radius without alignment, then apply the GP-LCB regret argument and the assumed information-gain rate. All quantities determined by the initial data are fixed throughout the conditional analysis.

Proof. Lemma 1 implies $\mathcal{H}_{k_{\mathrm{pref}}} = \mathcal{H}_{k_0}$ and $\|h\|_{\mathcal{H}_{k_{\mathrm{pref}}}} \le \sqrt{d_{\max}}\|h\|_{\mathcal{H}_{k_0}}$. Since $\|m_{\mathrm{pref}}\|_{\mathcal{H}_{k_0}} = \|\hat{v}\|_2$,

$$\|f - m_{\mathrm{pref}}\|_{\mathcal{H}_{k_{\mathrm{pref}}}} \le \overline{B}_{\mathrm{pref}} := \sqrt{d_{\max}}(B_0 + \|\hat{v}\|_2) < \infty. \tag{32}$$

Equation (14) provides the fixed finite bound

$$\begin{aligned}\|f - \mu^{\circ}_{\mathrm{pref}}\|_{\mathcal{H}_{k^{\circ}_{\mathrm{pref}}}} \le B^{\circ}_{\mathrm{pref}} := \overline{B}_{\mathrm{pref}}\sqrt{1 + n_0\kappa^2/\sigma_n^2} \\ + \sigma_n^{-1}\|\mathbf{z}_{\mathrm{ini}} - m_{\mathrm{pref}}(X_{\mathrm{ini}})\|_2.\end{aligned} \tag{33}$$

Any smaller valid fixed finite bound may be used instead. For this proposition, choose

$$\beta_t^{\mathrm{pref}} = \left[B^{\circ}_{\mathrm{pref}} + \sqrt{2\left(\gamma_{t-1}(k^{\circ}_{\mathrm{pref}}) + \log\frac{1}{\delta}\right)}\right]^2. \tag{34}$$

Unlike Eq. (24), this coefficient uses a radius valid without alignment, not the common reference radius.

Apply Lemma 3 conditionally with $(\bar{m}, k) = (\mu^{\circ}_{\mathrm{pref}}, k^{\circ}_{\mathrm{pref}})$, $B = B^{\circ}_{\mathrm{pref}}$, and $\eta = \delta$. The LCB selection and variance-budget argument in Eqs. (25)–(26) then give, on one event of conditional probability at least $1 - \delta$,

$$S^{\mathrm{pref}}_{n_0+N} \le \frac{R^{\mathrm{pref}}_N}{N} \le \sqrt{\frac{C_{\sigma,\kappa}\beta^{\mathrm{pref}}_N \gamma_N(k^{\circ}_{\mathrm{pref}})}{N}}, \qquad N \ge 1. \tag{35}$$

No use of Eq. (6) is made.

Conditioning reduces covariance, so $k^{\circ}_{\mathrm{pref}} \preceq k_{\mathrm{pref}} \preceq k_0$. Let $g_N = \gamma_N(k^{\circ}_{\mathrm{pref}})$. The hypothesis implies $g_N \le \gamma_N(k_0) = o(\sqrt{N})$, while Eq. (34) gives $\beta^{\mathrm{pref}}_N = O(1 + g_N)$. Thus the right-hand side of Eq. (35) is $O(\sqrt{(g_N + g_N^2)/N}) \to 0$. This proves Eq. (7) on the same event. □

Kernels satisfying the information-gain condition. With hyperparameters and positive noise variance fixed, the information gain is $O(\log N)$ for linear kernels and $O((\log N)^{d+1})$ for squared-exponential kernels (Srinivas et al., 2010). For Matérn kernels, the refined bound (Vakili et al., 2021) is

$$\gamma_N(k_0) = O\left(N^{d/(2\nu+d)}(\log N)^{2\nu/(2\nu+d)}\right),$$

which is $o(\sqrt{N})$ when $\nu > d/2$. These standard Euclidean kernels can be considered on a bounded box containing $\mathcal{X}$; restricting the domain cannot increase information gain. On a fixed finite domain of size J, any kernel with $k_0(x, x) \le \kappa^2$ instead satisfies

$$\gamma_N(k_0) \le \frac{J}{2}\log\left(1 + \frac{N\kappa^2}{J\sigma_n^2}\right),$$

because every Gram matrix has rank at most J and trace at most $N\kappa^2$. These are sufficient cases, not necessary conditions for optimization convergence.

D Implementation Details of Experiments

D.1 Benchmark Details

We evaluate the proposed method on six real-world black-box optimization benchmarks spanning traffic control, electric-vehicle charging, water-network operation, and materials discovery. For consistency, all tasks are formulated as minimization problems. We describe below the optimization objective, decision space, key simulation settings, and data sources for each benchmark.

D.1.1 Traffic Signal Control: Cologne3 (TSC-C3)

Problem setting. The Cologne3 benchmark is based on the RESCO traffic-signal-control suite (Ault & Sharon, 2021) and represents a corridor-scale network with three controlled intersections. The objective is to determine fixed-time signal timings that minimize average vehicle waiting time across the network. The problem is inherently coupled: increasing the green time for one movement may reduce its local queue but simultaneously reduce service to conflicting movements and alter downstream traffic arrivals.

Each candidate solution is evaluated using a one-hour microscopic traffic simulation from $25{,}200$ s to $28{,}800$ s. The control step is 5 s, and yellow transitions are fixed at 3 s.

Decision variables and objective. The decision vector is

$$\mathbf{x} = (x_1, \ldots, x_{11}) \in \mathbb{R}^{11}, \tag{36}$$

where each variable specifies the green duration, in seconds, of one selected signal phase. Variables x_1–x_3, x_4–x_7, and x_8–x_{11} correspond to the three controlled intersections, respectively. Their bounds are

$$\begin{aligned} &(x_1, x_3) \in [25, 50], \qquad x_2 \in [5, 12], \\ &(x_4, x_6) \in [22, 42], \qquad (x_5, x_7) \in [5, 12], \\ &(x_8, x_{10}) \in [30, 60], \qquad (x_9, x_{11}) \in [5, 12]. \end{aligned} \tag{37}$$

The objective is the average vehicle waiting time over the network:

$$\min_{\mathbf{x} \in \mathcal{X}} f_{\mathrm{C3}}(\mathbf{x}). \tag{38}$$

Data and simulator. The road network and traffic demand are taken from the Cologne3 scenario provided by RESCO, while vehicle movements are simulated using SUMO (Krajzewicz et al., 2002).

D.1.2 Traffic Signal Control: Cologne8 (TSC-C8)

Problem setting. Cologne8 is a larger traffic-signal-control instance from RESCO (Ault & Sharon, 2021), containing eight controlled intersections or aggregated signal groups. Compared with Cologne3, the problem contains more controlled phases and stronger network interactions. Signal changes at one intersection alter downstream arrival flows, queue formation, and potentially queue spillback, making Cologne8 a higher-dimensional and more strongly coupled optimization problem.

Each candidate is evaluated over the same one-hour interval, $25{,}200$–$28{,}800$ s, using a 5-s control step and fixed 3-s yellow transitions.

Decision variables and objective. The decision vector is

$$\mathbf{x} = (x_1, \ldots, x_{25}) \in \mathbb{R}^{25}. \tag{39}$$

The 25 variables control $4, 2, 3, 4, 3, 2, 3$, and 4 green phases in the eight signal groups, respectively. Most main-road or through-movement phases have bounds of $[20, 60]$ s, whereas turning or secondary phases have bounds of $[5, 15]$ s. The mapping between decision variables and signal phases is fixed by the RESCO scenario configuration.

The objective is the network-wide average vehicle waiting time,

$$\min_{\mathbf{x} \in \mathcal{X}} f_{\mathrm{C8}}(\mathbf{x}), \tag{40}$$

where a smaller value indicates lower overall traffic delay.

Data and simulator. The road network and traffic demand are taken from the RESCO Cologne8 scenario, and microscopic vehicle dynamics are simulated using SUMO (Krajzewicz et al., 2002).

D.1.3 EV CHARGING: CALTECH (EVC-CALTECH)

Problem setting. The Caltech benchmark considers electric-vehicle charging control in a campus parking facility using the SustainGym EV-charging environment (Yeh et al., 2023). Vehicles arrive at different times, request different amounts of energy, and have different expected departure times. Aggressive charging can improve service completion but may concentrate electricity consumption during periods with high marginal carbon emissions. Conversely, excessive postponement can leave vehicles undercharged at departure. The optimization task therefore balances charging completion and carbon emissions.

The network contains 54 charging stations. Each evaluation simulates a 24-hour period using 288 five-minute time steps.

Decision variables and control policy. We optimize a 12-dimensional parameterized charging policy. The day is divided into four periods:

$$00{:}00\text{–}08{:}00, \quad 08{:}00\text{–}11{:}00, \quad 11{:}00\text{–}15{:}00, \quad 15{:}00\text{–}24{:}00.$$

For each period $s \in \{1, \dots, 4\}$, the policy contains three parameters,

$$b_s \in [-4, 2], \qquad k_{u,s} \in [0, 15], \qquad k_{c,s} \in [0, 30]. \tag{41}$$

For vehicle i connected at time t, the normalized charging action is

$$a_{i,t} = \mathbb{I}(d_{i,t} > 0)\, \sigma\left(b_s + k_{u,s} \frac{d_{i,t}}{e_{\max} \max(\Delta_{i,t}, 1)} - k_{c,s} \left[m_t - \min_{1 \le h \le 12} \widehat{m}_{t+h} \right] \right), \tag{42}$$

where $d_{i,t}$ denotes the remaining requested energy, $\Delta_{i,t}$ is the number of time steps until the expected departure, $e_{\max} \approx 0.5547$ kWh is the reference energy delivered by charging at 32 A for five minutes, and m_t denotes the marginal operating emissions rate (MOER).

Objective. Let $R(\mathbf{x})$ denote the fraction of the total requested energy delivered before vehicle departure, and let $C(\mathbf{x})$ denote the total marginal carbon emissions. Because these two quantities have different physical scales, we normalize them using fixed empirical ranges estimated from an independent set of feasible charging policies sampled before the optimization experiments. Specifically,

$$R_N(\mathbf{x}) = \frac{R(\mathbf{x}) - R_{\min}}{R_{\max} - R_{\min}}, \qquad C_N(\mathbf{x}) = \frac{C(\mathbf{x}) - C_{\min}}{C_{\max} - C_{\min}}, \tag{43}$$

and define the minimization objective as

$$f(\mathbf{x}) = C_N(\mathbf{x}) - R_N(\mathbf{x}). \tag{44}$$

For the Caltech instance, the fixed calibration ranges are $(C_{\min}, C_{\max}) = (9.224171, 48.451679)$ and $(R_{\min}, R_{\max}) = (0.200677, 0.849536)$. These constants are used only to place the two objective components on comparable scales and are kept fixed for all methods and all runs; the calibration samples are not used to initialize the optimizers or to construct their surrogates.

For presentation, we report $50 f(\mathbf{x})$ in the experimental tables and figures to improve numerical readability. This positive rescaling is applied only to the reported values and does not affect solution rankings or the underlying optimization problem.

Data and simulator. Vehicle-session statistics are derived from the Caltech ACN-Data dataset (Lee et al., 2019). Sessions are generated using a 30-component Gaussian mixture model fitted to data from May 1 to August 31, 2021. We fix the evaluation seed to 46 and use MOER conditions corresponding to August 20, 2021. The resulting scenario contains 13 charging sessions with a total requested energy of 197.106 kWh. Charging dynamics are simulated using SustainGym and ACN-Sim (Yeh et al., 2023).

D.1.4 EV CHARGING: JPL (EVC-JPL)

Problem setting. The JPL benchmark models workplace electric-vehicle charging at the Jet Propulsion Laboratory using the SustainGym and ACN-Sim framework (Yeh et al., 2023). Compared with the Caltech scenario, vehicles are more concentrated around working hours and the aggregate charging demand is substantially larger. The objective is again to exploit temporal charging flexibility to reduce carbon emissions while satisfying vehicle energy requests before departure.

The JPL network contains 52 charging stations and is simulated for 24 hours using 288 five-minute time steps.

Decision variables and objective. We use the same 12-dimensional four-period policy as in the Caltech benchmark:

$$\mathbf{x} = (b_1, k_{u,1}, k_{c,1}, \ldots, b_4, k_{u,4}, k_{c,4}), \tag{45}$$

with

$$b_s \in [-4, 2], \qquad k_{u,s} \in [0, 15], \qquad k_{c,s} \in [0, 30]. \tag{46}$$

At each time step, the controller maps the vehicle's remaining demand, time to departure, and current and predicted MOER values to station-level charging actions using the policy in Eq. (42).

Objective. Let $R(\mathbf{x})$ denote the fraction of requested energy delivered before departure and $C(\mathbf{x})$ the total marginal carbon emissions. As in the Caltech benchmark, the two components are normalized using fixed empirical ranges estimated from an independent set of feasible charging policies before the optimization experiments:

$$f(\mathbf{x}) = \frac{C(\mathbf{x}) - C_{\min}}{C_{\max} - C_{\min}} - \frac{R(\mathbf{x}) - R_{\min}}{R_{\max} - R_{\min}}. \tag{47}$$

For the JPL instance, $(C_{\min}, C_{\max}) = (90.939282, 206.174107)$ and $(R_{\min}, R_{\max}) = (0.289229, 0.722820)$. These ranges are fixed for all methods and runs and are used only to place the two objective components on comparable scales, not for optimizer initialization or surrogate fitting. The normalization is performed independently for the JPL and Caltech instances, so their absolute objective values are not intended for direct cross-task comparison. For readability, tables and figures report $50f(\mathbf{x})$; this positive rescaling does not affect solution rankings.

Data and simulator. The vehicle-session model is fitted to JPL records from ACN-Data (Lee et al., 2019). Using the fixed evaluation seed 46 and the same August 20, 2021 MOER conditions yields 32 charging sessions with a total requested energy of 1123.548 kWh. Simulations are performed using SustainGym and ACN-Sim.

D.1.5 WATER NETWORK OPERATION: NET3

Problem setting. The Net3 benchmark considers daily variable-speed pump scheduling in a municipal water-distribution network. Pumping is a major source of energy consumption, but excessively reducing pump speeds can lead to inadequate service pressure. Frequent changes in pump speeds can also increase operational variability. The task therefore seeks a 24-hour operating policy that balances pumping energy, low-pressure performance, and control smoothness.

The base system is the standard EPANET Net3 network, containing 92 demand junctions, 117 pipes, two reservoirs, three tanks, and two pumps. Each objective evaluation simulates a complete 24-hour operating horizon.

Decision variables and objective. Pump speeds are adjusted every two hours, resulting in the 24-dimensional decision vector

$$\mathbf{x} = (s_{10,1}, \ldots, s_{10,12}, s_{335,1}, \ldots, s_{335,12}) \in [0.7, 1.2]^{24}, \tag{48}$$

where $s_{p,t}$ denotes the speed multiplier of pump $p \in \{10, 335\}$ during period t. The original Net3 operating rules are removed so that pump operations are fully determined by the decision vector.

We use the Profile-F objective

$$f_F(\mathbf{x}) = 0.75N_E(\mathbf{x}) + 0.05N_L(\mathbf{x}) + 0.20V(\mathbf{x}), \tag{49}$$

Table 5: Descriptor ranges for the COF Xe/Kr benchmark. The ranges describe the observed candidate library and do not define an unconstrained continuous hyperrectangle.

Descriptor	**Observed range**
Pore diameter (Å)	$[3.5094, 56.39856]$
Void fraction	$[0.16356, 0.92784]$
Surface area ($m^2\ g^{-1}$)	$[1996.63, 6357.01]$
Crystal-density descriptor	$[102.7248, 1610.7018]$
B fraction	$[0, 0.1818]$
O fraction	$[0, 0.25]$
C fraction	$[0.325, 0.6667]$
H fraction	$[0, 0.5]$
Si fraction	$[0, 0.02952]$
N fraction	$[0, 0.3333]$
S fraction	$[0, 0.14286]$
P fraction	$[0, 0.06667]$
Halogen fraction	$[0, 0.28571]$
Metal fraction	$[0, 0.02381]$

where N_E denotes normalized pumping-energy cost and N_L is a normalized low-pressure tail loss. The latter is constructed from the lowest 10% of pressure observations across demand nodes and reporting times, such that its minimization favors higher pressure in the lower tail. Pump-speed variability is measured by

$$V(\mathbf{x}) = \frac{1}{24} \sum_{p,t} \left(\frac{s_{p,t} - s_{p,t-1}}{1.2 - 0.7} \right)^2, \tag{50}$$

where cyclic indexing is used between the last and first control periods. The energy and pressure components are normalized using fixed affine ranges determined before optimization. For readability, tables and figures report $20 f_F(\mathbf{x})$; The positive linear transformation does not affect the relative ordering of the methods

Data and simulator. The hydraulic network is defined by the standard EPANET Net3 model. Simulation is performed using the Water Network Tool for Resilience (WNTR) with its EPANET-based hydraulic simulator (Klise et al., 2017). Hydraulic states are reported hourly, while pump controls are updated every two hours.

D.1.6 COF Materials Screening for Xe/Kr Separation

Problem setting. This benchmark considers screening covalent organic frameworks (COFs) for xenon/krypton separation. Xe and Kr have similar physical properties, making their separation energy intensive. Nanoporous materials with appropriate pore geometry and chemical composition can preferentially adsorb Xe, but evaluating adsorption selectivity using high-fidelity grand-canonical Monte Carlo (GCMC) simulations is computationally expensive (Gantzler et al., 2023). The optimization task is therefore to identify highly selective materials using a limited number of candidate evaluations.

Decision space and objective. The search space contains 608 feasible COF candidates. Each material is represented by 14 structural and compositional descriptors. Their observed ranges in the candidate library are summarized in Table 5.

Importantly, arbitrary combinations of these descriptor values do not necessarily correspond to physically realizable materials. The feasible domain is therefore the finite candidate set

$$\mathcal{X} = \{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(608)}\}. \tag{51}$$

The original task maximizes high-fidelity Xe/Kr adsorption selectivity. To maintain a common minimization convention across benchmarks, we define

$$\min_{\mathbf{x} \in \mathcal{X}} f(\mathbf{x}), \qquad f(\mathbf{x}) = -S^{\mathrm{HF}}_{\mathrm{Xe/Kr}}(\mathbf{x}). \tag{52}$$

The high-fidelity selectivity values correspond to 298 K, a Xe partial pressure of 0.2 bar, a Kr partial pressure of 0.8 bar, the UFF force field, and binary GCMC conditions (Gantzler et al., 2023).

Data and simulation framework. The material descriptors and high-fidelity targets are obtained from the public dataset released by Gantzler et al. (Gantzler et al., 2023). In the original study, structural descriptors were computed using Zeo++ and adsorption properties were evaluated through molecular simulation. In our benchmark, we query the released high-fidelity objective values rather than rerunning GCMC simulations. The resulting task is therefore a deterministic finite-candidate black-box optimization problem that retains the objective structure of the original computationally expensive materials-screening problem.

D.2 Baseline Configuration

Common evaluation protocol. All optimization methods follow the same evaluation protocol. For each independent run, the methods are initialized with the same set of 10 evaluated solutions and are subsequently allowed 200 sequential objective evaluations. Performance is measured solely by the number of black-box evaluations; model fitting and acquisition optimization do not consume the evaluation budget. For continuous domains, the initial designs are generated using scrambled Sobol sequences. For finite candidate spaces, the initial solutions are sampled uniformly without replacement from the candidate set. Across methods, the same initial design is used for a given random seed.

Gaussian-process Bayesian optimization. We consider standard GP-based Bayesian optimization with expected improvement (EI) and GP lower confidence bound (GP-LCB) acquisition functions. For EI, we use an exploration parameter $\xi = 0.01$.

For GP-LCB, after $t \geq n_0$ objective evaluations have been collected, the next candidate is selected by minimizing

$$\mu_t(x) - \sqrt{\beta^{\mathrm{exp}}_{t-n_0+1}}\,\sigma_t(x), \tag{53}$$

where the experimental exploration coefficient is

$$\beta^{\mathrm{exp}}_{t-n_0+1} = 2\log\left(\frac{t^2\pi^2}{6\delta}\right), \qquad \delta = 0.1. \tag{54}$$

The same GP-LCB exploration schedule is used for Vanilla BO (GP-LCB) and Ours (GP-LCB).

The acquisition function is optimized according to the structure of the search space. In continuous domains, we use multi-start bounded optimization initialized from a set of randomly generated candidates. In finite candidate spaces, the acquisition value is evaluated for every unevaluated candidate, and the candidate attaining the best acquisition value is selected. This avoids introducing an additional approximation to acquisition optimization in discrete search spaces.

CAKE. We use CAKE with BIC-Acquisition Kernel Ranking (BAKER) (Suwandi et al., 2026). The kernel search combines six base kernels (SE, PER, LIN, RQ, M3, and M5) through addition and multiplication. LLM-guided crossover and mutation evolve the kernel population, while BAKER selects the next query using BIC-weighted expected improvement. Acquisition optimization is performed over all unevaluated candidates in finite domains and through bounded multi-start optimization in continuous domains.

LGBO. We use a point-guided implementation of LGBO (Yuan et al., 2026). The LLM receives a compact history of up to 12 evaluated solutions, while the GP is fitted to all collected observations. Candidate selection uses qLogEI with LLM preference guidance; the preference-lifting grid size is 256 and the guidance scale is 3.0. Acquisition values are evaluated over up to 128 randomly sampled unevaluated candidates in finite domains or 2,048 novel Sobol candidates in continuous domains. For COF and Net3, we additionally use a local-anchor variant of the preference lift.

TuRBO. We include TuRBO as a representative trust-region Bayesian optimization baseline (Eriksson et al., 2019). For continuous domains, we use the single-trust-region TuRBO-1 formulation with a Matérn-$5/2$ Gaussian-process surrogate and Thompson sampling within the current

trust region. Decision variables are normalized to the unit hypercube, and the initial trust-region side length is set to 0.8. The trust region expands following consecutive successful iterations and contracts following consecutive unsuccessful iterations, following the standard TuRBO mechanism.

For finite candidate spaces, we use the same trust-region principle while restricting candidate selection to the available discrete solutions. The trust region is centered at the best observed solution in the normalized decision space, and only unevaluated candidates within the current region are considered. The region is expanded or contracted according to observed improvement. This preserves the local-search mechanism of TuRBO while respecting the finite feasible domain.

SMAC. We use the official SMAC3 implementation (v2.4.1) with the `HyperparameterOptimizationFacade` (Hutter et al., 2011). SMAC is configured for deterministic single-objective optimization using its default random-forest surrogate. To ensure identical initialization across methods, SMAC's internal initial design is disabled and the same 10 initial configurations are supplied through `additional_configs`, followed by 200 additional objective evaluations.

Candidate selection uses logarithmic expected improvement with $\xi = 0$. At each step, the acquisition optimizer considers 10,000 challengers, equally split between randomly generated configurations and local-search candidates. The random-design probability is set to 0.2, and the surrogate is retrained after every eight new evaluations. All other model and optimization settings follow the SMAC3 defaults. We only fix the ordering of local-search starting points to ensure reproducibility across processes; this does not modify the surrogate, acquisition function, or candidate-generation procedure.

E Additional Experimental Results

E.1 Performance of algorithms

We evaluate the optimization performance of the proposed method against seven baselines under limited evaluation budgets (after initialization). Table 6 groups the results by benchmark. Within each group, the rows correspond to the proposed method and seven baseline algorithms, while the columns report performance after 10, 50, 100, and 200 evaluations following the ten-point initial design. Each entry gives the mean and sample standard deviation over 20 seeds.

Under the smallest budget, the proposed method achieves the best mean performance on five of the six benchmarks and is close to the best result on EVC-JPL. At 200 iterations, it remains the best or tied-best method on TSC-C8, EVC-JPL, COF, and Net3. The results therefore demonstrate strong evaluation efficiency and broad performance across heterogeneous optimization problems.

E.2 Post-hoc Diagnostics of Prior–Objective Alignment

The alignment condition in Theorem 1 is a sufficient condition for comparing the finite-budget regret bounds of the preference-informed and reference GP-LCB runs. It is an oracle condition rather than an observable requirement of the proposed method: evaluating Eq. (6) requires $v_f = L^{-1} f(\mathcal{Z})$, and hence the true objective values at the inducing locations. These values are unavailable to the optimizer by construction. For the continuous benchmarks, obtaining them exactly would require additional high-fidelity evaluations outside the optimization budget. Although an oracle check is possible retrospectively for fully evaluated finite candidate pools, such a check is benchmark-specific and, because Eq. (6) is sufficient rather than necessary, failure to satisfy it would not imply that the preference prior cannot improve realized optimization performance.

We therefore use a complementary post-hoc diagnostic to examine the empirical notion underlying the alignment assumption: after conditioning on the same initial objective observations, does the preference-informed posterior capture the objective ordering more faithfully than the reference posterior? For each seed, the preference-informed prior and the reference zero-mean GP prior are conditioned on the same ten initial observations. The resulting posteriors are then evaluated on an external set of candidate points and candidate pairs. Objective values used for this diagnostic are not exposed during preference elicitation, prior construction, or Bayesian optimization.

Table 6: Optimization performance under limited evaluation budgets (after initialization). Lower is better, and the best result for each benchmark and budget is highlighted in bold.

Method	Objective@10	Objective@50	Objective@100	Objective@200
		TSC-C3 (11D)		
Random Search	1.89 ± 0.60	1.36 ± 0.59	1.21 ± 0.47	0.89 ± 0.27
Vanilla BO (GP-LCB)	1.87 ± 0.63	1.43 ± 0.46	1.27 ± 0.34	1.01 ± 0.32
Vanilla BO (EI)	1.78 ± 0.64	1.49 ± 0.43	1.36 ± 0.33	1.07 ± 0.28
SMAC	1.75 ± 0.59	1.30 ± 0.42	1.12 ± 0.44	0.97 ± 0.49
TuRBO	1.36 ± 0.53	0.97 ± 0.44	0.89 ± 0.38	0.66 ± 0.40
LGBO	1.70 ± 0.75	1.07 ± 0.47	0.93 ± 0.40	0.67 ± 0.40
CAKE	1.71 ± 0.63	1.35 ± 0.55	1.03 ± 0.39	0.87 ± 0.33
Ours (GP-LCB)	1.20 ± 0.31	0.97 ± 0.29	0.92 ± 0.28	0.74 ± 0.33
		TSC-C8 (25D)		
Random Search	5.48 ± 0.74	3.87 ± 1.34	2.96 ± 1.18	2.03 ± 1.03
Vanilla BO (GP-LCB)	2.58 ± 1.86	0.57 ± 0.28	0.47 ± 0.11	0.17 ± 0.21
Vanilla BO (EI)	2.81 ± 1.76	0.60 ± 0.36	0.42 ± 0.18	0.26 ± 0.21
SMAC	4.12 ± 1.73	1.93 ± 1.57	0.85 ± 0.90	0.42 ± 0.39
TuRBO	3.90 ± 1.77	2.11 ± 1.68	1.25 ± 1.33	0.45 ± 0.76
LGBO	3.85 ± 1.87	0.70 ± 0.69	0.34 ± 0.40	0.00 ± 0.00
CAKE	4.05 ± 1.72	0.56 ± 0.56	0.22 ± 0.32	0.02 ± 0.10
Ours (GP-LCB)	2.35 ± 0.58	0.66 ± 0.27	0.26 ± 0.25	0.00 ± 0.00
		EVC-JPL (12D)		
Random Search	-1.99 ± 0.45	-2.45 ± 0.37	-2.79 ± 0.50	-3.20 ± 0.59
Vanilla BO (GP-LCB)	-7.25 ± 1.41	-8.15 ± 0.14	-8.23 ± 0.12	-8.28 ± 0.11
Vanilla BO (EI)	-5.78 ± 1.57	-7.12 ± 1.06	-7.29 ± 0.89	-7.42 ± 0.77
SMAC	-2.94 ± 0.74	-4.91 ± 0.82	-5.70 ± 0.62	-6.11 ± 0.50
TuRBO	-4.32 ± 1.68	-7.02 ± 1.64	-7.49 ± 1.40	-7.94 ± 1.03
LGBO	-3.73 ± 0.96	-5.19 ± 0.51	-5.69 ± 0.14	-6.94 ± 0.00
CAKE	-4.63 ± 1.42	-6.83 ± 1.01	-7.36 ± 0.65	-7.53 ± 0.51
Ours (GP-LCB)	-7.15 ± 0.81	-8.14 ± 0.10	-8.21 ± 0.03	-8.29 ± 0.10
		EVC-Caltech (12D)		
Random Search	-1.93 ± 0.89	-2.60 ± 0.54	-2.75 ± 0.39	-2.94 ± 0.24
Vanilla BO (GP-LCB)	-3.24 ± 0.20	-3.42 ± 0.04	-3.46 ± 0.04	-3.54 ± 0.01
Vanilla BO (EI)	-3.12 ± 0.32	-3.39 ± 0.06	-3.41 ± 0.03	-3.45 ± 0.05
SMAC	-2.47 ± 0.75	-3.14 ± 0.21	-3.29 ± 0.09	-3.37 ± 0.04
TuRBO	-2.56 ± 1.03	-3.07 ± 0.94	-3.12 ± 0.95	-3.13 ± 0.96
LGBO	-2.84 ± 0.54	-3.29 ± 0.04	-3.38 ± 0.01	-3.38 ± 0.01
CAKE	-3.20 ± 0.20	-3.44 ± 0.05	-3.48 ± 0.05	-3.50 ± 0.02
Ours (GP-LCB)	-3.38 ± 0.03	-3.39 ± 0.02	-3.43 ± 0.05	-3.49 ± 0.03
		COF (14D)		
Random Search	7.45 ± 4.19	2.43 ± 2.86	1.46 ± 2.03	0.39 ± 0.77
Vanilla BO (GP-LCB)	2.01 ± 1.86	0.04 ± 0.11	0.00 ± 0.00	0.00 ± 0.00
Vanilla BO (EI)	3.75 ± 4.45	0.14 ± 0.39	0.07 ± 0.15	0.04 ± 0.11
SMAC	4.00 ± 4.53	0.41 ± 1.75	0.00 ± 0.00	0.00 ± 0.00
TuRBO	4.66 ± 4.45	0.88 ± 2.14	0.42 ± 1.01	0.16 ± 0.62
LGBO	2.43 ± 2.88	0.08 ± 0.38	0.00 ± 0.00	0.00 ± 0.00
CAKE	4.64 ± 3.98	0.13 ± 0.18	0.00 ± 0.00	0.00 ± 0.00
Ours (GP-LCB)	0.99 ± 1.05	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
		Net3 (24D)		
Random Search	5.76 ± 0.62	5.48 ± 0.44	5.18 ± 0.45	4.91 ± 0.46
Vanilla BO (GP-LCB)	4.15 ± 1.45	1.87 ± 0.43	1.47 ± 0.33	1.28 ± 0.24
Vanilla BO (EI)	4.37 ± 1.01	2.81 ± 0.51	2.37 ± 0.42	2.21 ± 0.45
SMAC	5.55 ± 0.51	4.48 ± 0.73	3.50 ± 0.60	2.60 ± 0.32
TuRBO	4.94 ± 0.66	2.16 ± 0.19	1.73 ± 0.20	1.21 ± 0.07
LGBO	4.35 ± 0.56	3.77 ± 0.09	3.49 ± 0.00	3.23 ± 0.00
CAKE	5.02 ± 0.78	2.89 ± 0.35	2.53 ± 0.60	2.23 ± 0.68
Ours (GP-LCB)	2.00 ± 0.19	2.00 ± 0.19	2.00 ± 0.19	1.07 ± 0.07

Table 7 reports two complementary metrics. Pairwise accuracy measures whether the posterior mean correctly predicts which candidate in a pair has the lower objective value, whereas Spearman correlation measures agreement between the posterior-mean ranking and the objective-based ranking. All entries report the mean and sample standard deviation over 20 seeds.

The preference-informed posterior improves both metrics over the reference GP on five of the six benchmarks, with particularly large improvements on EVC-JPL, EVC-Caltech, and Net3. TSC-C8 is the exception, where the reference GP obtains higher pairwise accuracy and rank correlation. These results indicate that preference information often improves ranking-level alignment with the objective after shared initialization, but that the benefit remains task dependent. Importantly, pairwise accuracy and rank correlation do not evaluate the precision-weighted residual in Eq. (6); the results should therefore be interpreted as empirical diagnostics related to the theorem's motivating notion of alignment, rather than as direct verification of its sufficient condition.

Table 7: Posterior ranking prediction accuracy after conditioning on the same ten initial samples for each seed. Values are mean ± sample standard deviation over 20 seeds. Objective values are used only for external evaluation. Higher is better.

	Pairwise accuracy (%)		Spearman correlation	
Benchmark	Preference distribution	GP zero prior	Preference distribution	GP zero prior
TSC-C3	**57.71 ± 3.14**	51.38 ± 4.92	**0.216 ± 0.066**	0.045 ± 0.137
TSC-C8	62.38 ± 2.06	**66.05 ± 4.87**	0.352 ± 0.047	**0.452 ± 0.137**
EVC-JPL	**85.14 ± 1.35**	65.13 ± 5.02	**0.881 ± 0.011**	0.453 ± 0.146
EVC-Caltech	**89.06 ± 1.31**	67.82 ± 5.31	**0.937 ± 0.007**	0.523 ± 0.134
COF	**73.44 ± 2.75**	66.76 ± 9.60	**0.623 ± 0.054**	0.457 ± 0.257
Net3	**87.95 ± 0.75**	71.45 ± 3.67	**0.851 ± 0.015**	0.603 ± 0.067

E.3 Performance under the EI acquisition function

We further examine whether the benefit of the preference-derived prior is specific to the GP-LCB acquisition function. To this end, we combine the same preference-informed GP prior with expected improvement (EI) and compare *Ours (EI)* with *Ours (GP-LCB)* and the seven baselines under identical initial designs and evaluation protocols.

Table 8 reports the average rank across the six benchmarks at different numbers of post-initialization objective evaluations. At each checkpoint, the nine methods are ranked independently within each benchmark using their unrounded mean objective values, with lower values ranked higher; exact ties are assigned mid-ranks. The resulting ranks are then averaged equally across the six benchmarks. The best average rank in each column is shown in bold and the second-best is underlined.

Ours (EI) achieves average ranks of 1.83, 1.67, 2.42, 3.75, and 3.75 after 10, 20, 50, 100, and 200 post-initialization evaluations, respectively. It ranks second at 10 and 20 evaluations, first at 50 evaluations, and ties for second at 200 evaluations. Together with the consistently strong performance of Ours (GP-LCB), these results indicate that the benefit of the elicited prior is not specific to GP-LCB. In particular, the preference-informed prior remains highly competitive when combined with EI, especially in the low- and intermediate-budget regimes.

Table 8: Average rank across six benchmarks at different evaluation budgets. Lower is better. The best result is highlighted in bold, and the second-best result is underlined.

Method	Rank@10	Rank@20	Rank@50	Rank@100	Rank@200
Random Search	9.00	8.33	8.67	8.67	8.50
Vanilla BO (GP-LCB)	3.50	4.00	3.00	3.58	<u>3.75</u>
Vanilla BO (EI)	5.00	5.00	5.33	5.83	6.00
SMAC	7.33	7.33	7.00	6.42	6.58
TuRBO	6.00	6.00	6.00	5.17	5.00
LGBO	5.00	5.33	5.67	5.42	4.92
CAKE	5.83	5.83	4.33	<u>3.42</u>	4.08
Ours (GP-LCB)	**1.50**	**1.50**	<u>2.58</u>	**2.75**	**2.42**
Ours (EI)	<u>1.83</u>	<u>1.67</u>	**2.42**	3.75	<u>3.75</u>

E.4 Comparison with an Agent-Generated Low-Fidelity Model

A natural alternative to our preference-to-GP construction is to ask the agent to construct a scalar scoring model and use its predictions as a low-fidelity proxy of the expensive objective. This alternative directly addresses whether the agent-derived knowledge needs to be converted through pairwise preference learning, or whether the resulting scoring model can be used more directly in Bayesian optimization. We therefore evaluate an *Agent-score GP-LCB* baseline on all six benchmarks.

The comparison is designed to isolate the representation of agent-derived knowledge. Both approaches provide the agent with the same pre-evaluation information, including the problem description, decision-variable definitions, admissible ranges, and static scenario files, while withholding all high-fidelity objective values, simulator outputs, and optimization histories. The difference lies in how the extracted knowledge is represented: our method uses the agent-generated model to produce pairwise preferences, whereas the baseline asks the agent to construct a scalar scoring function once before optimization and subsequently treats its output as low-fidelity information.

Agent-score low-fidelity model. The raw agent scores are sign-aligned with the common minimization convention, such that lower values correspond to more promising designs, and are standardized using only the low-fidelity score data. We fit a sparse radial-basis-function (RBF) GP to these scores, using a fixed standardized low-fidelity observation-noise standard deviation of $\sigma_L = 0.5$. Let $m_L(x)$ denote its posterior mean. We then define the high-fidelity surrogate as

$$f_H(x) = m_L(x) + \delta(x), \qquad \delta \sim \mathcal{GP}(0, k_{\mathrm{RBF}}), \tag{55}$$

where $m_L(x)$ is fixed after the low-fidelity model is constructed and $\delta(x)$ models the discrepancy between the agent-derived score model and the high-fidelity objective. Only the posterior mean of the low-fidelity GP is transferred to the high-fidelity model; its posterior covariance is not propagated. The discrepancy GP is subsequently updated using high-fidelity objective evaluations, with observation-noise standard deviation fixed at $\sigma_H = 0.05$. These noise settings are specified in advance and are not tuned using benchmark ground-truth values.

Evaluation protocol. Both methods use the same ten initial high-fidelity evaluations for each random seed, followed by 200 sequential high-fidelity evaluations. Both use GP-LCB under the common minimization convention, with $\delta_{\mathrm{LCB}} = 0.1$, and each benchmark is evaluated over 20 independent seeds. TSC-C3, TSC-C8, and COF are evaluated using simple regret relative to the best design in their corresponding finite candidate spaces, whereas EVC-JPL, EVC-Caltech, and Net3 are evaluated using the best-so-far objective value in their continuous domains. Lower values indicate better performance throughout.

Figure 4 shows that the scalar Agent-score model is a competitive alternative, although neither representation dominates uniformly. Agent-score GP-LCB improves more rapidly in the early stage on TSC-C8 and Net3, and also exhibits a pronounced early advantage on EVC-JPL. In contrast, preference-to-GP improves faster on TSC-C3 and COF. At the end of the evaluation budget, preference-to-GP achieves better performance on TSC-C3 and both EV-charging benchmarks, while the two approaches are close on COF and Net3 and achieve comparable final performance on TSC-C8.

The comparison supports the broader premise of our work: domain knowledge encoded in an LLM can be operationalized into informative approximations of complex black-box objectives before expensive evaluations are collected. Directly using the agent-generated scalar model is therefore a viable alternative when its cardinal output is sufficiently meaningful. The preference-to-GP formulation provides a different interface that relies only on relative comparisons rather than interpreting the magnitude of the raw agent scores as calibrated objective information. Consequently, it avoids the need to specify an explicit score-to-objective calibration relationship before incorporating the elicited knowledge into BO. We therefore do not claim that scalar Agent-score models are ineffective, or that preference-to-GP dominates universally. Instead, the two approaches represent complementary ways of transferring agent-derived domain knowledge, with preference-to-GP providing a direct ordinal route to constructing an informative probabilistic prior.

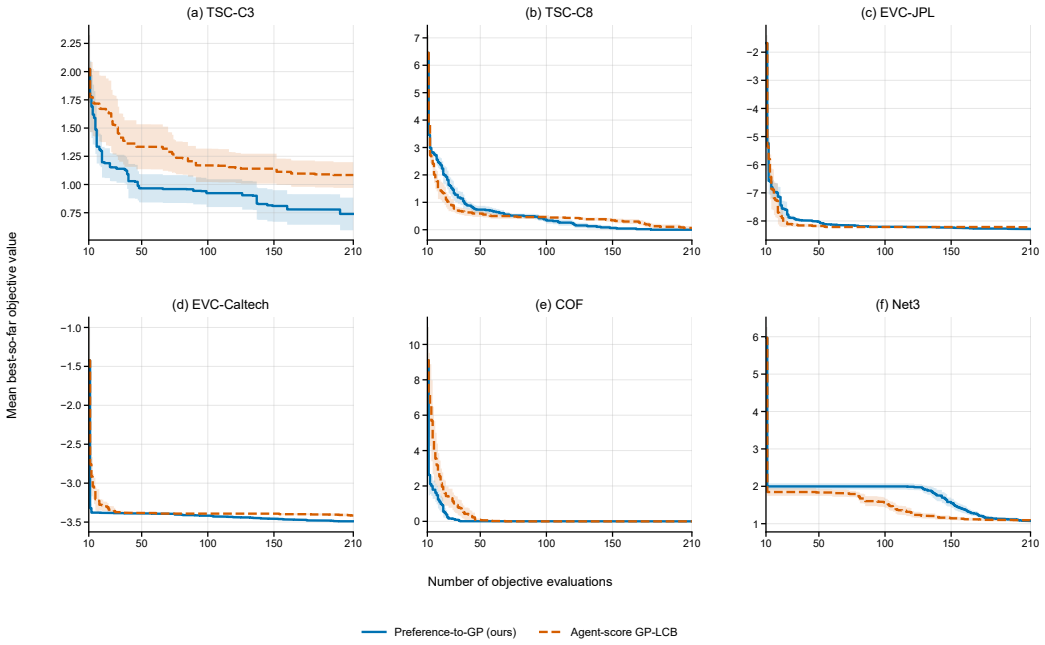

Figure 4: Comparison between preference-to-GP and Agent-score GP-LCB across six benchmarks. Both methods use the same ten initial high-fidelity evaluations followed by 200 sequential high-fidelity evaluations, with 20 independent seeds for every benchmark. Curves show the mean best-so-far performance, and shaded regions denote pointwise 95% confidence intervals of the mean. The horizontal axis counts total high-fidelity objective evaluations, including the ten shared initialization evaluations. TSC-C3, TSC-C8, and COF report finite-pool simple regret, whereas EVC-JPL, EVC-Caltech, and Net3 report the incumbent objective value in their continuous search spaces. Lower is better.