HAPBO: HARNESSED α-π BAYESIAN OPTIMIZATION VIA EXPERT NATURAL LANGUAGE PRIORS

Anonymous authors
Paper under double-blind review

ABSTRACT

Bayesian Optimization (BO) can be substantially more sample-efficient when informed by prior knowledge about the problem. In practice, however, such knowledge is often available only in natural language, e.g., expert intuitions or documentation, rather than as the explicit probability distribution landscape that prior-guided BO methods may require. Leveraging it effectively demands both an expressive mapping from language to priors and robustness to information that may be misleading. We propose HAPBO (Harnessed α-π Bayesian Optimization), which elicits beliefs once from a human expert, a large language model (LLM), or both, and deterministically compiles a structured specification into a reusable belief function ($\pi : \mathcal{X} \to [0, 1]$), with a rich representation not restricted to a Gaussian family. Building on α-πBO, HAPBO applies this belief function through a bounded positive multiplier on any nonnegative acquisition, preserving positive acquisition support and requiring no LLM calls during optimization. We prove that, for the Expected Improvement acquisition function (EI), HAPBO preserves the standard simple-regret rate up to a bounded factor, even under adversarial beliefs. We evaluate HAPBO on 40 HPOBench tasks, a synthetic suite in 6–15 dimensions and two real-world applications: cortical neurostimulation and constrained solar design. HAPBO achieves the lowest aggregate HPOBench regret and strong real-world task performance, with zero in-loop token cost. In real-world applications it improves the optimization outcome even at budgets of 8-30 queries, and at a small fraction of the wall-clock and token costs of other LLM-based methods.

1 INTRODUCTION

Bayesian optimization (BO) is widely used for optimizing expensive black-box functions. An evaluation may for instance require patient time, hours of computation, or a physical experiment. A surrogate model, typically a Gaussian process (GP), approximates the objective, and an acquisition function selects the next query (Mockus, 1975; Jones et al., 1998; Frazier, 2018). In most real-world scenarios, BO never truly starts the search from zero, since the practitioner usually has additional knowledge about the problem. For example, a machine learning engineer knows the plausible range of each hyperparameter of the network she is designing (Eggensperger et al., 2021), a neuroscientist knows which nervous system region controls a movement (Bonizzato et al., 2023), and a solar thermal engineer can have an estimation of which design parameters must be balanced against capital cost (Andrés-Thió et al., 2025). Used well, such knowledge buys the same result in fewer evaluations, and the gain is largest where evaluations are scarcest: at small budgets, where the surrogate alone (the internal model used by BO) is not mature yet (Hvarfner et al., 2022; Guay-Hottin et al., 2025). In real BO applications, such knowledge is most naturally expressed in language rather than as a probability distribution: an expert can describe where good configurations lie, and an LLM can produce a comparable description from a task specification alone.

To take advantage of the prior belief, prior-guided BO techniques have been developed. These techniques allow an expert to inject a prior over the location of the optimum of the objective function. Since experts can make mistakes, these techniques may sometimes incorporate mechanisms that limit the damage when the prior is misleading (Hvarfner et al., 2022; Guay-Hottin et al., 2025; Souza et al., 2021; Hvarfner et al., 2024b). However, these techniques assume the prior is already available in numerical form, which practitioners are rarely equipped to provide.

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.

Here, our fundamental question is whether LLMs can bridge the gap between knowledge in its many native forms (e.g., a sentence by an expert, knowledge acquired during pretraining, or documents retrieved by web search) and a compiled belief function that is directly compatible with the landscape of GP models and BO acquisition functions.

Large Language Models (LLMs) trained on internet-scale datasets have demonstrated broad knowledge and reasoning capabilities, so the BO community has tried to incorporate them within the BO loop. Current LLM-guided BO algorithms may use LLMs to 1) warm-start the search, 2) narrow the search space, or 3) replace core BO components (surrogate model or the acquisition function) during optimization (Liu et al., 2024; Cissé et al., 2025; Yuan et al., 2026). Our working hypothesis is that the GP-BO machinery is already well suited to provide high performance in most optimization tasks, from low to high dimension (Hvarfner et al., 2024a). Moreover, in-loop LLM methods may incur substantial per-iteration cost, while lacking the calibrated uncertainty mechanisms developed for prior-guided BO techniques (Kristiadi et al., 2024; Gupta et al., 2025). Methods that elicit the belief once avoid this per-iteration cost, but use it in one of two limited ways: to warm-start the search, so that the belief shapes only the first few evaluations (Liu et al., 2024), or to prune the search space, so that any region it excludes can never be revisited (Lao et al., 2024). The first may discard the belief too early; the second may trust it too much.

To overcome these gaps, we introduce HAPBO (Harnessed α-π Bayesian Optimization). HAPBO elicits beliefs once from a human expert, an LLM, or both, and deterministically compiles a structured specification into a reusable belief function ($\pi\colon \mathcal{X} \to [0,1]$; see Figure 1). Our contributions are as follows:

- **(i)** In order to transform the resulting natural language knowledge into a BO-compatible landscape, we designed a Prior Specification Language (PSL), with a rich representation not restricted to a Gaussian family. The compiled PSL landscape is then compatible with traditional prior injection methods (a prior to the mean or to the acquisition function).
- **(ii)** We choose to build upon α-πBO, correcting its normalization, which allows HAPBO to apply this belief function through a bounded positive multiplier on any nonnegative acquisition, preserving positive acquisition support and requiring no LLM calls during optimization. We prove that, for the Expected Improvement acquisition function (EI), HAPBO preserves the standard simple-regret rate up to a bounded factor, even under adversarial beliefs.
- **(iii)** To evaluate HAPBO in practice, we ask whether HAPBO can improve sample efficiency of State-Of-The-Art (SOTA) methods: 1) one-shot LLM strategies that warm-start or constrain the search space, 2) methods that keep the LLM inside the BO loop, and 3) traditional prior-free search. We specifically ask how robust HAPBO is against misleading LLM-elicited beliefs, a potential roadblock that its component α-πBO is designed to alleviate.
- **(iv)** We also present a series of ablations to interrogate the value of each component equipped to HAPBO by our chain of design choices. Our questions include whether results vary with the injection site, the belief representation, and the source of the belief.

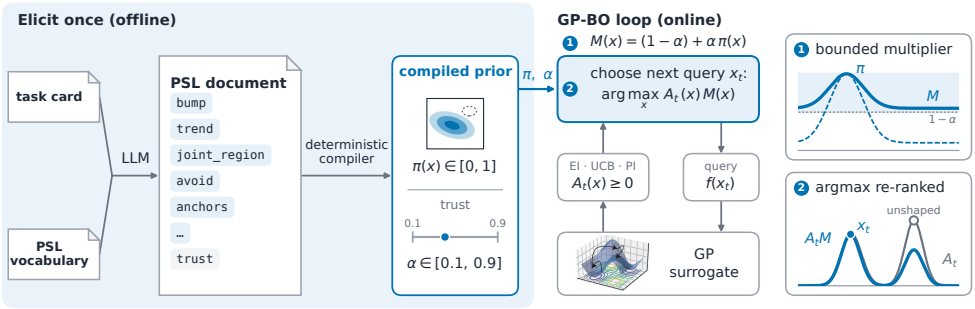


Figure 1: **HAPBO (blue components).** Left to right: an LLM reads the task card and the PSL vocabulary and produces a document in the typed Prior Specification Language (PSL); a compiler turns the document into a prior landscape π and a trust level α. A human expert can inject information in the task card. The prior is injected in the BO loop at the acquisition function level. Right: shaping re-ranks candidates toward plausible regions but cannot zero a positive base acquisition.

2 Related Work

Prior-guided BO. A substantial corpus of work focuses on allowing practitioners to inject a prior over the location of the optimum. The resulting methods differ in how they limit the damage of a misleading belief, which they do in one of four ways.

First, by a schedule. π-BO weights the acquisition function by the prior landscape π, raised to an exponent that decays with the number of iterations, regardless of the observed data (Hvarfner et al., 2022). Second, by accumulating evidence. The prior's influence shrinks as observations contradict it, either through the model, as in BOPrO, which combines the prior with a data-driven model into a pseudo-posterior (Souza et al., 2021), and ColaBO, which conditions the Gaussian process (GP) on the belief (Hvarfner et al., 2024b), or through the sampling policy, as in PriorBand, which adapts its reliance on prior-based samples in multi-fidelity hyperparameter optimization (HPO) (Mallik et al., 2023), and HypBO, which distributes effort between expert-specified hypothesis regions and the full space (Cissé et al., 2024). Third, by explicitly testing the belief. DynaBO combines decay with a surrogate-based safeguard test that rejects flagged priors (Fehring et al., 2025), and PSH proves a no-harm guarantee for prior-guided successive halving (Hennig et al., 2026). Fourth, by bounding the prior's influence structurally. α-πBO (although descending from π-BO) does not insert a fading schedule, but rather places a fixed floor on the acquisition value, which results in quickly incorporating "good" priors, while drowning misleading ones, without a decay schedule and without deciding whether the belief is wrong (Guay-Hottin et al., 2025). We build the HapBO framework on the latter mechanism, but we here correct the normalization of this construction so that the floor yields a true constant-factor guarantee, and we provide the accompanying theory.

A common challenge across all methods, model-based and acquisition-based alike, is that they require beliefs in a formal probabilistic form that the surrogate or the acquisition function can absorb. We here add the language interface that produces it.

LLMs for BO. Another line of work incorporates LLMs within the BO loop. These approaches vary mainly in the frequency with which the LLM is queried during optimization. Some approaches query the LLM before the search as a warm-starting strategy, either to select initial points to initialize the surrogate model (Liu et al., 2024; Cissé et al., 2025), or to produce a prior that persists throughout the search (Lao et al., 2024). Others query the LLM at every iteration of the optimization by using LLMs to replace core components of the BO loop such as the surrogate model or the acquisition function (AF), among others (Liu et al., 2024; Suwandi et al., 2025; Chen et al., 2025; Yuan et al., 2026; Agarwal et al., 2025), or query the LLM on demand to guide the AF in the selection of the next point to sample (Cissé et al., 2025; Chen et al., 2026b). Our working hypothesis is that warm-starting initial points alone may not exploit the full capacity of BO priors. Also, available methods that produce priors that persist during the search may miss a key mechanism for exploiting it effectively, and in consequence can stop their action at producing the prior, and use it to constrain the search space rather than to weight it. Finally, using LLMs in the loop introduces additional challenges of uncalibrated uncertainty, volatility, that can lead to have these methods consistently outperformed by classical surrogate-based methods. They also incur a query cost for knowledge that helps only in domain-specific settings (Kristiadi et al., 2024; Gupta et al., 2025; Chang et al., 2025; Kobalczyk et al., 2026). HapBO therefore proposes to query the LLM before the search, leaving core BO components unchanged. Rather than warm-starting the surrogate model, or constraining the search, we give the acquisition function a prior to use through the entire search.

Belief elicitation. Outside of BO, AutoElicit generates Gaussian priors over the parameters of a linear model (Capstick et al., 2025). Also, LLM-Prior converts rich, unstructured context, e.g. natural-language descriptions, data, or figures into analytically convenient probability distributions suitable for Bayesian inference (Huang, 2025). HapBO differs from these methods, since both stop at the prior, whereas ours is carried into the BO loop.

3 Methodology

3.1 Belief elicitation

The first step of HapBO is eliciting a belief in natural language. This stage requires a prompt to an LLM. The second step is to provide the vocabulary to the LLM to transcribe the elicited belief into

a document that is then compiled deterministically. Two cards are used to drive this step: the task card provides the LLM's context, and the vocabulary card harnesses what the belief document can contain.

task card. The task card gives the context needed by the LLM to write the belief document. At minimum, it shall contain three sections: the domain, which describes the system under study; the search space, which specifies each parameter; and a description of the optimization problem. The card can further contain human expert knowledge, pertinent data, and a scope of the search process (e.g., deep research, looking for literature sources, access to data). The design of the task card is open to the practitioner implementing HAPBO in their research, and benefits from prompt engineering skills developed with common LLM use. In one experiment, we distinguish between cards written from three knowledge sources, whose variants differ only in what the card contains. *LLM-only* cards ask the LLM to act as a domain expert and form the belief from its own knowledge; *human-expert* cards carry substantial expert's belief, including available data from previous experiments; and *mixed* cards carry the expert's belief written in their own words and ask the LLM to complement it with its own knowledge.

Vocabulary card. The vocabulary card is fixed by HAPBO design, and defines the Prior Specification Language (PSL), a closed grammar for writing beliefs, and restricts the LLM to translation: the LLM writes the belief in PSL and never computes a prior. PSL expresses per-dimension shapes, pairwise interactions, penalized regions, and anchor points. The resulting belief is joint over the search space and not restricted to a Gaussian family (see Appendix B).

Belief document. Both cards are sent together in a single LLM call, which returns a belief document containing the typed search space, the list of beliefs, and a qualitative trust block. An example of an elicited document is given in Appendix B.

3.2 DETERMINISTIC COMPILATION

The compiler maps the belief document to the two quantities used by α-πBO: a prior landscape $\pi : \mathcal{X} \to [0, 1]$ and a trust level α. It involves no LLM call and no randomness, so the same document always yields the same prior. To build π, each belief is drawn as a fixed shape on the encoded search space; the shapes are summed, with penalties subtracted, clamped at zero, multiplied by any preference gates, and rescaled to $[0, 1]$, after which anchors set exact values at the configurations they name (Algorithm 1, Appendix B).

To determine a trust level α we opted for an automatic simplistic rule, that allows us to illustrate two possible practitioner choices at once. We combine the document's LLM-stated confidence c_{overall} with the provenance of each belief,

$$\alpha = \text{clip}(c_{\text{overall}} \cdot \langle c_{\text{provenance}} \rangle_w,\ 0.1,\ 0.9)\,, \tag{1}$$

where $\langle \cdot \rangle_w$ is the average weighted by the belief weights. A higher α means more trust in the prior. Capping α at 0.9 bounds the worst-case constant of our guarantee at 10 (Proposition 2 in App. M.2). In our implementation, we set the belief stated by the expert to a higher trust level than one supplied by the LLM. This idea is set to showcase at once the combination of an LLM-stated confidence, with a more 'human' fixed trust rule. Nevertheless, a practitioner using HAPBO could effectively substitute our simplified α-determination module with an LLM decision-maker, or even with a human-tuned cursor.

3.3 SHAPING THE ACQUISITION FUNCTION

HAPBO leaves the BO loop (presented in Appendix A) unchanged except for the final ranking. At each query it selects

$$x_t \in \arg\max_{x \in \mathcal{X}} A_t(x)\, M(x), \qquad M(x) = (1 - \alpha) + \alpha\, \pi(x), \tag{2}$$

where $A_t \geq 0$ is the raw base acquisition, e.g., EI:

$$A_t(x) = \big(\mu(x) - \hat{y}\big)\, \Phi\big(z(x)\big) + \sigma(x)\, \phi\big(z(x)\big) \tag{3}$$

with $z(x) = (\mu(x) - \hat{y})/\sigma(x)$; (App. A).

The prior landscape π and the trust α are compiled once before the run (Section 3.2).

Following Guay-Hottin et al. (2025), the belief acts as a bounded multiplier; unlike their formulation, which min–max normalizes the acquisition before multiplying, we apply M to the raw acquisition, which is what makes the guarantees below hold (Definition 1 in App. M). When comparing with Vanilla BO, the surrogate, the base acquisition, and the candidate set are kept identical, and both start from the same random point. Because the belief enters only through M, the rule applies to any nonnegative acquisition (Appendix H).

3.4 Theory guarantees

The PSL harness itself is free of expressive bottlenecks, because 1) anchors-only documents can reproduce any belief table on a finite candidate set exactly, and 2) approximate any continuous belief uniformly with error $O(K^{-1/d_c})$ in the number of anchors K over d_c continuous dimensions (Theorem 1).

Regarding the α-π harness, since $\alpha \leq 0.9$, the multiplier satisfies $0.1 \leq M(x) \leq 1$, so no candidate with positive acquisition is ever ruled out, and the selected point is always within a factor 10 of the raw acquisition's maximum (Proposition 2). For noiseless Expected Improvement with a fixed kernel, shaping preserves the classical simple-regret rate of Bull (2011) up to a constant $1/(1-\alpha\beta_\star) \leq 10$, where $\beta_\star = 1 - \pi(x^\star)$ is the prior's miss at the optimum (Theorem 2, Corollary 1); a prior that is correct at the optimum costs nothing, however wrong it is elsewhere. For noisy GP-UCB, the regret bound gains an additive term that depends on the belief only through its value at the optimum (Theorem 3).

Full statements, assumptions, and proofs are given in Appendix M.

4 Experimental setup

Benchmarks. HPOBench contributes 40 tabular hyperparameter-optimization tasks spanning five learner families (Eggensperger et al., 2021). The NeuroMOSAICS Cortical panel optimizes flexor responses across 16 rat-M1 electrode tasks from six rats (Kardassevitch et al., 2025). SOLAR MAX-EFF_RE is a constrained solar-design black box (Andrés-Thió et al., 2025). We further designed an analytic panel, containing high-dimensional synthetic functions: Hartmann-6, Styblinski–Tang-7, Levy-10, and Ackley-15 (Hartman, 1973; Styblinski & Tang, 1990; Levy & Montalvo, 1985; Ackley, 2012). For this panel, we generate three controlled prior fidelities per experiment, by manipulating the synthetic functions to create alternate functions within the same family but having a diverse landscape (correlation $\rho = 0.1$ to 0.9, from misleading to accurate). We then feed the manipulated function to the elicitor, informing the LLM that this is our expert belief for the desired landscape. The elicited prior is tested against the true exact synthetic function.

Baselines. We compare HAPBO against an array of methods. Some are *prior-free*: vanilla BO, random search and TURBO (Eriksson et al., 2019). Other methods are *prior-guided*, like COLABO (Hvarfner et al., 2024b), which conditions the GP posterior on a user belief, which we provide via an LLM query, and we particularly focus on other methods incorporating a *LLM-in-the-loop* approach: LLAMBO (Liu et al., 2024), which uses the LLM as surrogate and candidate sampler; BORA (Cissé et al., 2025), which lets the LLM propose regions during the search; and LGBO (Yuan et al., 2026), which shifts the GP mean by an LLM preference at each query. The in-loop methods receive the same task card as HAPBO, through their own interface, and query the same locally served LLM. Where a baseline admits more than one faithful implementation, we report the strongest variant we could construct (Appendix J). Two further methods, CAKE (Suwandi et al., 2025) and CALIPER (Chen et al., 2026a), are reported in Appendix L.

Protocol. Every baseline runs with the same locally served LLM (Qwen3.5-122B Qwen Team, 2026) and the same task descriptions, served through its native interface. HAPBO and Vanilla BO share the surrogate, acquisition, candidate set, and initial point (same as random search), and differ only in the presence or absence of the prior.

For HAPBO, priors are elicited once per task, offline and before any optimization, at temperature 0: the elicitor receives a de-identified task card and must answer in the closed PSL vocabulary (Appendix B), with validator errors fed back for up to four corrective turns (Appendix C). Per-

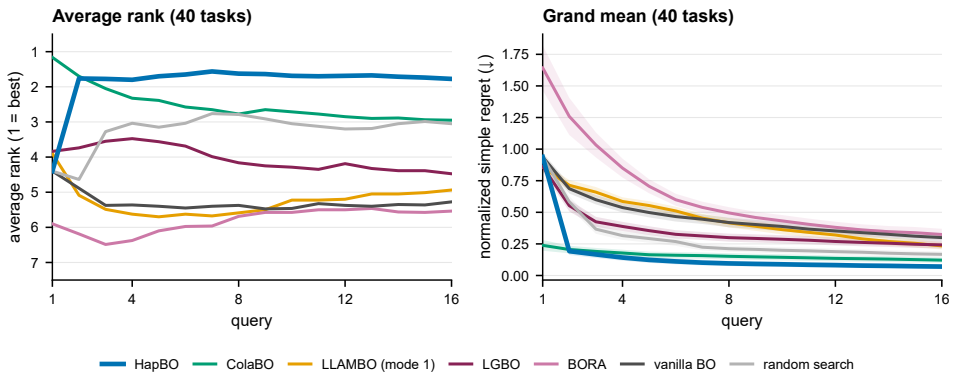


Figure 2: **HPOBench.** Average rank and normalized simple regret as a grand mean over 40 tasks for five model families. LLAMBO omits ∼32 transient run failures. Per-family panels in App. G; bands are ±SEM across tasks after averaging seeds.

benchmark normalizations, SOLAR's constraint handling, and per-query wall-clock and token costs are detailed in Appendix D.

Metrics. We evaluate query-wise best-so-far performance. Simple regret is used when a reference optimum is known; SOLAR reports best feasible efficiency because its optimum is unknown. Means and uncertainty bands are computed over tasks or subjects after averaging paired seeds within the unit. Endpoint summaries use ten seeds.

5 Results

5.1 HPOBench and real-world tasks

We first compared HapBO to the SOTA and traditional baselines on HPOBench. HapBO has the lowest grand-average regret curve throughout (Fig. 2). Its mean raw endpoint regret, averaged within and then across the five model families, is 0.0030, compared with 0.0094 for ColaBO, 0.0145 for its matched random-search control, 0.0199 for LLAMBO, 0.0243 for its matched vanilla-BO control, 0.0243 for LGBO, and 0.0304 for BORA. The rank panel in Fig. 2 positions HapBO ahead of every baseline in the performance ranking across HPOBench problem families. Two further SOTA methods, CAKE and CALIPER, are competitive under a faithful prior but do not overtake HapBO (Appendix L, Fig. 15).

In two real-world experiments, pertaining to cortical neurostimulation and solar design, HapBO performs on par with SOTA, whereas no other method does (ColaBO being as performant in Cortical, but largely underperforming in SOLAR, and LLAMBO vice-versa). Under SOLAR, LLAMBO completes 14 of 20 runs, compared to HapBO's 20 of 20. Table 1 collects the endpoints.

Cost. The efficiency gains are bought with modest compute overhead. HapBO pays the LLM once, at setup (∼32,500 tokens to elicit and compile one HPO prior), then runs as a pure GP loop at 0.39 s and zero tokens per query, a tenth of a second slower than vanilla BO. Conversely, SOTA baselines have larger expenditures, with LLAMBO spending ∼89,400 tokens and 30.6 s per query, ColaBO 68 s per query for its prior-conditioned sampling. BORA and LGBO, which engage in more compute time per query, manage to mitigate the tokens/query figure, sitting at few hundreds of tokens per query on average (App. Table 3).

5.2 Robustness in the accurate-to-misleading prior spectrum

We next ask what price an adversarial misbelief would carry within the optimization process. An enticing feature of prior-based BO methods is the capacity to disregard prior information that is proven non-performant. Eventually, the ideal method would be expected to close the gap, query after query, with prior-free variants under these conditions.

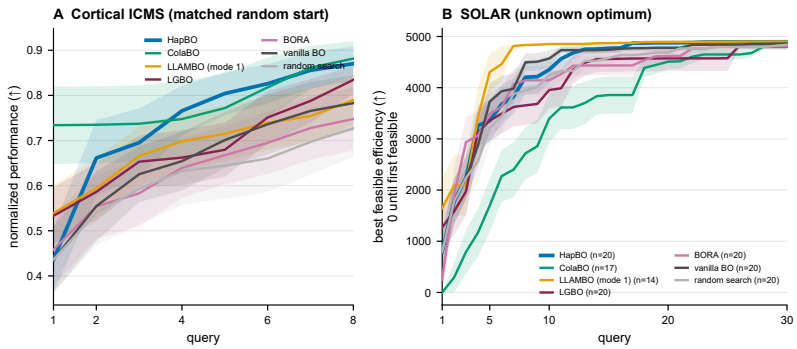


Figure 3: **Real-world tasks. A:** NeuroMOSAICS cortical normalized performance (higher is better); means and SEM are clustered over six experimental subjects. **B:** SOLAR_MAXEFF best feasible efficiency through 30 queries; zero denotes that no feasible point has yet been found. Legend counts are completed runs out of 20 attempts.

Table 1: **Endpoint performance per task family.** HPOBench simple regret for each of the five model families (lr, rf, xgb, nn, svm; lower is better), cortical normalized performance (best-so-far over the subject optimum, higher is better — the same unit as Fig. 3A), and SOLAR best feasible efficiency (higher); regrets are comparable across HPO families because all are validation error. *fail%* is the pooled fraction of task–seed runs that did not complete across every family (HPO, cortical, and SOLAR). HAPBO uses matched random initialization on HPO, cortical, and SOLAR alike. Bold marks the best value in each performance column.

Method	lr	rf	xgb	nn	svm	cortical	SOLAR	fail%
HapBO	**0.002**	0.001	**0.002**	**0.006**	**0.004**	0.870	**4896.6**	0.0
ColaBO	0.005	**0.001**	0.002	0.018	0.021	**0.882**	4798.5†	0.6
LLAMBO	0.004†	0.050	0.004†	0.019†	0.022†	0.790	***4901.7***†	7.9
LGBO	0.006	0.047	0.003	0.019	0.046	0.834	4823.9	0.0
BORA	0.013	0.012	0.003	0.042	0.081	0.748	4813.0	0.0
vanilla BO	0.010	0.048	0.005	0.034	0.026	0.783	4863.0	0.0
random	0.003	0.039	0.002	0.009	0.019	0.727	4838.1	0.0

† performance limited to successful runs, presented undiscounted for failures.

We also propose a synthetic function experiment, where the prior fidelity (as a modified function expressed in natural language with an equation) is controlled for exact correlation levels with respect to the true target function to be optimized. Our experiment makes the cost of prior quality visible for every method on one display (Fig. 4). At $\rho \approx 0.9$ prior-using methods can separate from the prior-free controls; as ρ falls to 0.5 and 0.1 they lose the early advantage and approach or fall behind the controls, which are unchanged across columns. Fig. 4 shows that HAPBO's performance degrades gracefully, and even in the most misleading case its curve converges back toward the no-prior trajectory rather than diverging. ColaBO is the most performant method at the high prior quality point, but HAPBO outperforms it for lower ρ values.

We then tested HAPBO and SOTA methods performance under matched adversarial priors for HPO (prior placed at each task's empirical anti-optimum), with the same hostile description passed to every baseline. HAPBO initially pays for the misleading belief but recovers better than any SOTA alternative toward the strongest prior-free curves of random search (Fig. 5).

5.3 ELICITATION MODALITIES

Knowledge sources. The same compiled interface accepts LLM-only, human-only, and combined beliefs. We show an example of this three-way usage of the LLM elicitor interface on the NeuroMOSAICS Cortical panel (Appendix F). We elicit the prior with 1) no hint (LLM only), 2) a modest injection of human-level expertise in the form of a single informative sentence (LLM+human), and

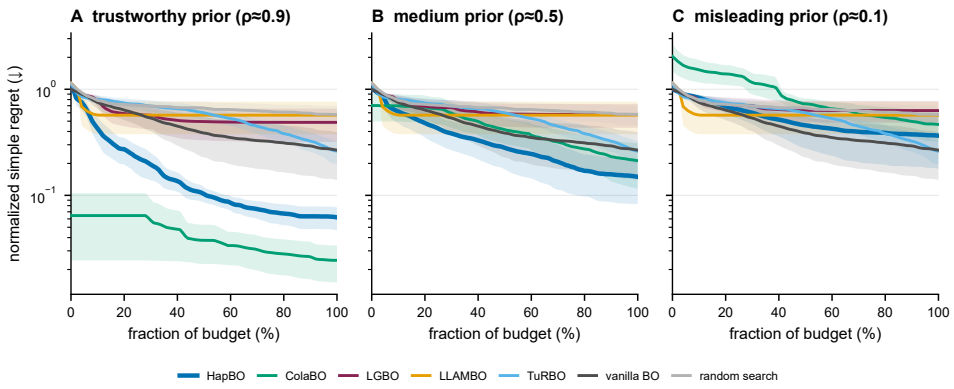


Figure 4: **Synthetic prior-fidelity sweep.** Grand-average simple regret under trustworthy ($\rho \approx 0.9$), imperfect ($\rho \approx 0.5$), and misleading ($\rho \approx 0.1$) priors on 4 synthetic functions. Each function is evaluated at its own budget and aligned only by fraction of budget for this summary.

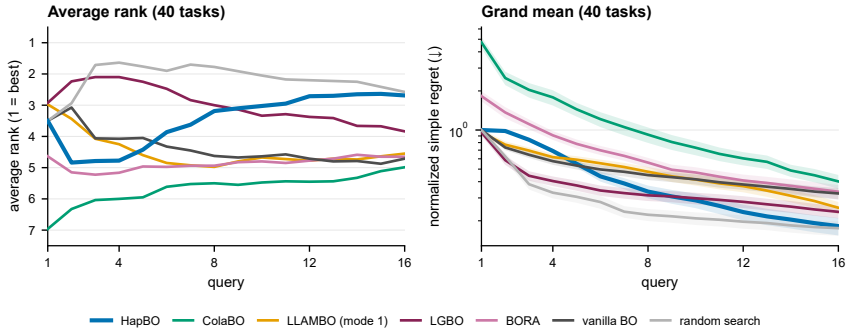


Figure 5: **Adversarial HPOBench.** Average task-wise rank (left; over the task grid complete for every method) and grand-mean normalized simple regret over 40 tasks (right) when every method receives the same hostile task description through its native interface. Per-family panels in App. G; bands are ±SEM across tasks after averaging ten seeds.

3) substantial prior information in the form of previous experimental data not included in the benchmark dataset (human only). Each mode exceeds its matched vanilla trajectory (Fig. 8G–I).

Elicitor LLM. We further test different elicitor LLMs, including commercial frontier models (App. I). The final regret seems to track elicitor quality, suggesting that leveraging more potent models can result in a higher BO benefit from the HAPBO harness.

5.4 ABLATIONS

Injection site. In this ablation we ask whether changing the injection site for a prior leads to a difference in performance, as expected from literature (Hvarfner et al., 2022; Guay-Hottin et al., 2025). We fixed the representation and moved the site: 1) the bounded acquisition multiplier of HAPBO, 2) the no-floor π-BO, 3) a prior to the mean of the surrogate (prior-to-mean), or 4) the initial queries (prior-warm-start), with a prior-only (non-BO) greedy arm and no-prior controls. Across HPOBench (good and adversarial prior) and cortical, HAPBO results the strongest landscaped method (vs π-BO and prior-to-mean). As expected, all landscaped algorithms avoid dramatic collapse of performance when the prior is misleading (Fig. 7).

Representation. In this ablation we ask whether PSL holds a substantial advantage with respect to three more naive prior-building languages (point-only, direct LLM elicitation and gaussian-only). We show that, whereas in low dimension all techniques offer similar results, in high dimension PSL offers a substantial advantage, as shown in our synthetic suite. Of note, direct-LLM elicitation is not possible in such case, since the fixed anchor set degenerates in 7-15D (Fig. 7).

The final regrets are displayed in App. E.

BO variants. In this ablation we ask whether the bounded multiplier depends on the BO machinery around it, since it acts only on the acquisition ranking. We repeat the matched prior-on/prior-off contrast under EI, PI, and UCB, and under sequential versus batch selection (Kriging Believer (Ginsbourger et al., 2010), $q = 3$). Across all three acquisitions, shaping reduces final normalized HPOBench regret by $\approx 10\times$, with the largest cortical gains at low budget, and the advantage is preserved under batch selection (Appendix H, Fig. 11).

5.5 KNOWLEDGE LEAK CONTROL

The objective of this paper is to determine, when LLM knowledge about a given problem exists, whether this can be integrated, by proper harnessing, in BO. However, there is a type of existing knowledge that is less desirable when testing any LLM-prior method on public benchmarks, which is contamination: a strong prior might reflect recall of the final benchmark answer rather than transferable expertise. We address this with two complementary controls: a direct probe of what the elicitor natively knows, and an additional objective whose optimum cannot have been memorized (Appendix K).

First, we show that no elicitor recalls an optimum on any real task: the blind guess recovers 49–77% of the head-room on HPOBench and falls below the map average on the cortical grid, while naming a synthetic function outright recovers its optimum exactly. Second, we build a bespoke CNN benchmark whose response surface is thus completely unseen to the model. There, the ordering of the main results is preserved, with HAPBO reaching the lowest simple regret (Fig. 14).

6 CONCLUSION AND FUTURE WORK

We presented HAPBO, a framework that takes natural-language beliefs, from a human expert, an LLM, or both, and converts them into a structured specification, which is then deterministically compiled into a reusable belief function with a rich representation palette. This belief function can be used by any BO method admitting a landscaped prior; we instantiate it with α-πBO, which is robust to erroneous beliefs. Since the belief function is compiled once before optimization, the resulting pipeline requires no LLM calls during optimization, and rather admits a traditional (shaped) BO loop. Results on 40 hyperparameter optimization problems, a synthetic suite in 6–15 dimensions, and two real-world applications show that HAPBO outperforms methods that keep the LLM in the loop under small budgets, in terms of performance, optimization time, and token cost; our experiments also show that HAPBO, among all the prior-modulated methods, is the most robust to misleading priors, since a misleading prior tends to only waste a small budget, and no region of the search space is ruled out by the prior-injection technique.

Several limitations remain. First, the quality of the compiled belief depends on how information is presented to the LLM when converting natural language into a specification; more broadly, each component of the pipeline is modular and further research could improve some of our design choices. Elicitation design in particular is a promising direction for future work. Second, HAPBO relies on one-shot elicitation: the belief function is compiled once before optimization and cannot be updated with what the search reveals. A natural extension is a hybrid approach in which the LLM occasionally intervenes during optimization to refine the belief. Finally, a misleading prior can still degrade performance relative to vanilla BO under small budgets; further limiting this negative effect is an additional available direction for future work.

Overall, we here advocate for a black-box optimization approach that uses the LLM offline to construct an auditable belief, then let an ordinary BO loop decide how much the data flow should trust or overturn it.

AI USE STATEMENT

In this work, the authors used generative AI tools to formulate mathematical claims, provide critical ingredients for proving mathematical claims, assist in the writing of proofs, implement methods, and assist with translation. The authors did not use generative AI tools to help develop theoretical models or conceptual frameworks, propose or refine hypotheses, design or provide feedback on

research methodology or experiments, or interpret results. Supporting qualitative and thematic data analysis is not applicable to this work. Additionally, the authors used generative AI tools to generate synthetic datasets (limited to the synthetic function experiment and the bespoke CNN experiment in appendix), reformat datasets (limited to SOLAR, allowing a tabular format for testing on an offline batch run), create or modify scientific figures or images, create or edit software code, draft parts of the paper, and edit the paper to improve readability. In each case, the authors verified that the scientific figures were accurate and free of errors, two authors verified and tested the code generated by AI coding agents, and the authors revised the drafted text. The authors take responsibility for the final content of this work, including any text, claims, or artifacts produced with the aid of generative AI.

Ethics statement

The cortical experiments use a public rodent neurostimulation dataset (Kardassevitch et al., 2025); no animal experiments were conducted for this work.

Reproducibility statement

In order to make HapBO results reproducible, we provide the source code for this project at `https://anonymous.4open.science/r/HapBO-27E5/README.md`. There, readers have access to documentation with instructions on how to run each of the experiments presented in the main text and the appendix. We also provide readers with the pipeline that allows them to generate their own priors by modifying the appropriate `task_card.md`. The data processing used for each dataset and instructions on how to download the datasets can be found in `benchmark/README.md`. For the theoretical results, we also provide explanations of any assumptions and a complete proof of the claims in the appendix.

References

David Ackley. *A connectionist machine for genetic hillclimbing*. Springer science & business media, 2012.

Dhruv Agarwal, Manoj Ghuhan Arivazhagan, Rajarshi Das, Sandesh Swamy, Sopan Khosla, and Rashmi Gangadharaiah. Searching for optimal solutions with LLMs via Bayesian optimization. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 67180–67201, 2025. URL `https://proceedings.iclr.cc/paper_files/paper/2025/file/a79237d6c47feef24c1df723cb41f674-Paper-Conference.pdf`.

Nicolau Andrés-Thió, Charles Audet, Miguel Diago, Aïmen E. Gheribi, Sébastien Le Digabel, Xavier Lebeuf, Mathieu Lemyre Garneau, and Christophe Tribes. Solar: a solar thermal power plant simulator for blackbox optimization benchmarking. *Optimization and Engineering*, 26(3): 1815–1861, 2025. doi: 10.1007/s11081-024-09952-x.

Anthropic. System card: Claude Haiku 4.5, October 2025. URL `https://www.anthropic.com/claude-haiku-4-5-system-card`.

Anthropic. System card: Claude Opus 4.8, May 2026a. URL `https://www-cdn.anthropic.com/0b4915911bb0d19eca5b5ee635c80fef830a37ea.pdf`.

Anthropic. System card: Claude Sonnet 4.6, February 2026b. URL `https://www.anthropic.com/claude-sonnet-4-6-system-card`.

Marco Bonizzato, Rose Guay Hottin, Sandrine L. Côté, Elena Massai, Léo Choinière, Uzay Macar, Samuel Laferrière, Parikshat Sirpal, Stephan Quessy, Guillaume Lajoie, Marina Martinez, and Numa Dancause. Autonomous optimization of neuroprosthetic stimulation parameters that drive the motor cortex and spinal cord outputs in rats and monkeys. *Cell Reports Medicine*, 4(4): 101008, 2023. doi: 10.1016/j.xcrm.2023.101008.

Adam D. Bull. Convergence rates of efficient global optimization algorithms. *Journal of Machine Learning Research*, 12(88):2879–2904, 2011. URL `http://jmlr.org/papers/v12/bull11a.html`.

Alexander Capstick, Rahul G. Krishnan, and Payam Barnaghi. AutoElicit: Using large language models for expert prior elicitation in predictive modelling. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 6746–6777. PMLR, 2025. URL https://proceedings.mlr.press/v267/capstick25a.html.

Chih-Yu Chang, Milad Azvar, Chinedum Okwudire, and Raed Al Kontar. LLINBO: Trustworthy LLM-in-the-loop Bayesian optimization, 2025. URL https://arxiv.org/abs/2505.14756.

Jiangyu Chen, Yi Ban, and Tianfan Fu. Evidence-gated LLM priors for multi-objective Bayesian optimization, 2026a. URL https://arxiv.org/abs/2606.01730.

Qi Chen, Fabio Ramos, Alán Aspuru-Guzik, and Florian Shkurti. Informing acquisition functions via foundation models for molecular discovery, 2025. URL https://arxiv.org/abs/2512.13935.

Zhuo Chen, Xinzhe Yuan, Jianshu Zhang, Jinzong Dong, Ruichen Zhou, Yingchun Niu, Tianhang Zhou, Yu Yang Fredrik Liu, Yuqiang Li, Nanyang Ye, and Qinying Gu. LABO: LLM-accelerated Bayesian optimization through broad exploration and selective experimentation, 2026b. URL https://arxiv.org/abs/2605.22054. To appear at ICML 2026.

Sayak Ray Chowdhury and Aditya Gopalan. On kernelized multi-armed bandits. In Doina Precup and Yee Whye Teh (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 844–853. PMLR, 06–11 Aug 2017. URL https://proceedings.mlr.press/v70/chowdhury17a.html.

Abdoulatif Cissé, Xenophon Evangelopoulos, Sam Carruthers, Vladimir V. Gusev, and Andrew I. Cooper. HypBO: Accelerating black-box scientific experiments using experts' hypotheses. In Kate Larson (ed.), *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24*, pp. 3881–3889. International Joint Conferences on Artificial Intelligence Organization, 8 2024. doi: 10.24963/ijcai.2024/429. URL https://doi.org/10.24963/ijcai.2024/429. Main Track.

Abdoulatif Cissé, Xenophon Evangelopoulos, Vladimir V. Gusev, and Andrew I. Cooper. Language-based Bayesian optimization research assistant (BORA). In James Kwok (ed.), *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence, IJCAI-25*, pp. 4967–4975. International Joint Conferences on Artificial Intelligence Organization, 8 2025. doi: 10.24963/ijcai.2025/553. URL https://doi.org/10.24963/ijcai.2025/553. Main Track.

Katharina Eggensperger, Philipp Müller, Neeratyoy Mallik, Matthias Feurer, René Sass, Aaron Klein, Noor Awad, Marius Lindauer, and Frank Hutter. HPOBench: A collection of reproducible multi-fidelity benchmark problems for HPO. In J. Vanschoren and S. Yeung (eds.), *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1, 2021. URL https://datasets-benchmarks-proceedings.neurips.cc/paper_files/paper/2021/file/93db85ed909c13838ff95ccfa94cebd9-Paper-round2.pdf.

David Eriksson, Michael Pearce, Jacob Gardner, Ryan D Turner, and Matthias Poloczek. Scalable global optimization via local bayesian optimization. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/6c990b7aca7bc7058f5e98ea909e924b-Paper.pdf.

Lukas Fehring, Marcel Wever, Maximilian Spliethöver, Leona Hennig, Henning Wachsmuth, and Marius Lindauer. Dynamic priors in Bayesian optimization for hyperparameter optimization, 2025. URL https://arxiv.org/abs/2511.02570.

Peter I. Frazier. A tutorial on Bayesian optimization, 2018. URL https://arxiv.org/abs/1807.02811.

David Ginsbourger, Rodolphe Le Riche, and Laurent Carraro. Kriging is well-suited to parallelize optimization. In *Computational intelligence in expensive optimization problems*, pp. 131–162. Springer, 2010.

Rose Guay-Hottin, Lison Kardassevitch, Hugo Pham, Guillaume Lajoie, and Marco Bonizzato. Robust prior-biased acquisition function for human-in-the-loop Bayesian optimization. *Knowledge-Based Systems*, 311:113039, 2025. ISSN 0950-7051. doi: 10.1016/j.knosys.2025.113039. URL https://www.sciencedirect.com/science/article/pii/S0950705125000863.

Rushil Gupta, Jason Hartford, and Bang Liu. LLMs for Bayesian optimization in scientific domains: Are we there yet? In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 15482–15510, Suzhou, China, November 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.findings-emnlp.838. URL https://aclanthology.org/2025.findings-emnlp.838/.

James K. Hartman. Some experiments in global optimization. *Naval Research Logistics Quarterly*, 20(3):569–576, 1973. doi: https://doi.org/10.1002/nav.3800200316. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/nav.3800200316.

Leona Hennig, Jasmin Brandt, Lukas Fehring, Barbara Hammer, Marius Lindauer, and Marcel Wever. Provably reduced sample cost in prior-guided hyperparameter optimization, 2026. URL https://arxiv.org/abs/2606.04866.

Yongchao Huang. LLM-prior: A framework for knowledge-driven prior elicitation and aggregation, 2025. URL https://arxiv.org/abs/2508.03766.

Carl Hvarfner, Danny Stoll, Artur Souza, Marius Lindauer, Frank Hutter, and Luigi Nardi. πBO: Augmenting acquisition functions with user beliefs for Bayesian optimization. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=MMAeCXIa89.

Carl Hvarfner, Erik Orm Hellsten, and Luigi Nardi. Vanilla Bayesian optimization performs great in high dimensions. In *Forty-first International Conference on Machine Learning*, 2024a. URL https://openreview.net/forum?id=OfT8MgIqHT.

Carl Hvarfner, Frank Hutter, and Luigi Nardi. A general framework for user-guided Bayesian optimization. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 42393–42410, 2024b. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/b95f3f8b6cff9e7ac6553a56b895cf70-Paper-Conference.pdf.

Donald R Jones, Matthias Schonlau, and William J Welch. Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, 13(4):455–492, 1998. doi: 10.1023/A:1008306431147.

Lison Kardassevitch, Davide Burchielli, Numa Dancause, and Marco Bonizzato. NeuroMOSAICS: A collection of neurostimulation datasets - multi-scale open-source across interfaces conditions & species. *NeuroLibre Reproducible Preprints*, 33, 2025. doi: 10.55458/neurolibre.00033. URL https://doi.org/10.55458/neurolibre.00033. Data archive: https://doi.org/10.5281/zenodo.15692258.

Katarzyna Kobalczyk, Zhiyuan Jerry Lin, Benjamin Letham, Zhuokai Zhao, Maximilian Balandat, and Eytan Bakshy. LILO: Bayesian optimization with natural language feedback. In *International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=zVrbE9ZtEU.

Agustinus Kristiadi, Felix Strieth-Kalthoff, Marta Skreta, Pascal Poupart, Alán Aspuru-Guzik, and Geoff Pleiss. A sober look at LLMs for material discovery: Are they actually good for Bayesian optimization over molecules? In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st*

International Conference on Machine Learning, volume 235 of *Proceedings of Machine Learning Research*, pp. 25603–25622. PMLR, 2024. URL https://proceedings.mlr.press/v235/kristiadi24a.html.

H. J. Kushner. A New Method of Locating the Maximum Point of an Arbitrary Multipeak Curve in the Presence of Noise. *Journal of Basic Engineering*, 86(1):97–106, March 1964. ISSN 0021-9223. doi: 10.1115/1.3653121. URL https://doi.org/10.1115/1.3653121.

Jiale Lao, Yibo Wang, Yufei Li, Jianping Wang, Yunjia Zhang, Zhiyuan Cheng, Wanghu Chen, Mingjie Tang, and Jianguo Wang. GPTuner: A manual-reading database tuning system via GPT-guided Bayesian optimization. *Proceedings of the VLDB Endowment*, 17(8):1939–1952, April 2024. ISSN 2150-8097. doi: 10.14778/3659437.3659449. URL http://dx.doi.org/10.14778/3659437.3659449.

Alejandro Velasco Levy and Antonio Montalvo. The tunneling algorithm for the global minimization of functions. *SIAM Journal on Scientific and Statistical Computing*, 6(1):15–29, 1985.

Tennison Liu, Nicolás Astorga, Nabeel Seedat, and Mihaela van der Schaar. Large language models to enhance Bayesian optimization. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 31252–31284, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/84b8d9fcb4e262fcd429544697e1e720-Paper-Conference.pdf.

Neeratyoy Mallik, Edward Bergman, Carl Hvarfner, Danny Stoll, Maciej Janowski, Marius Lindauer, Luigi Nardi, and Frank Hutter. PriorBand: Practical hyperparameter optimization in the age of deep learning. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 7377–7391. Curran Associates, Inc., 2023. doi: 10.52202/075280-0323. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/1704fe7aaff33a54802b83a016050ab8-Paper-Conference.pdf.

J. Mockus. On the Bayes methods for seeking the extremal point. *IFAC Proceedings Volumes*, 8(1, Part 1):428–431, 1975. ISSN 1474-6670. doi: 10.1016/S1474-6670(17)67769-3. URL https://www.sciencedirect.com/science/article/pii/S1474667017677693. 6th IFAC World Congress (IFAC 1975) - Part 1: Theory, Boston/Cambridge, MA, USA, August 24-30, 1975.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026. URL https://qwen.ai/blog?id=qwen3.5.

Artur Souza, Luigi Nardi, Leonardo B. Oliveira, Kunle Olukotun, Marius Lindauer, and Frank Hutter. Bayesian optimization with a prior for the optimum. In Nuria Oliver, Fernando Pérez-Cruz, Stefan Kramer, Jesse Read, and Jose A. Lozano (eds.), *Machine Learning and Knowledge Discovery in Databases. Research Track*, Lecture Notes in Computer Science, pp. 265–296, Cham, 2021. Springer International Publishing. ISBN 978-3-030-86523-8. doi: 10.1007/978-3-030-86523-8_17.

Niranjan Srinivas, Andreas Krause, Sham M. Kakade, and Matthias Seeger. Gaussian process optimization in the bandit setting: No regret and experimental design, 2009. URL https://arxiv.org/abs/0912.3995. Extended version of the ICML 2010 paper, with proofs.

Niranjan Srinivas, Andreas Krause, Sham M. Kakade, and Matthias W. Seeger. Gaussian process optimization in the bandit setting: No regret and experimental design. In Johannes Fürnkranz and Thorsten Joachims (eds.), *Proceedings of the 27th International Conference on Machine Learning (ICML-10), June 21-24, 2010, Haifa, Israel*, pp. 1015–1022. Omnipress, 2010. URL https://icml.cc/Conferences/2010/papers/422.pdf.

Michael A Styblinski and T-S Tang. Experiments in nonconvex optimization: stochastic approximation with function smoothing and simulated annealing. *Neural Networks*, 3(4):467–483, 1990.

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. In D. Belgrave,

C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, pp. 132690–132723. Curran Associates, Inc., 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/c03a2610bca2712b984b331fd4f7bb6f-Paper-Conference.pdf.

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 *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=LktUOZayG9.

A GAUSSIAN-PROCESS BAYESIAN OPTIMIZATION

BO seeks to find $x^\star \in \arg\max_{x\in\mathcal{X}} f(x)$ for an expensive black-box function f in as few evaluations as possible (Mockus, 1975; Jones et al., 1998; Frazier, 2018). Starting from initial observations $D_n = \{(x_i, y_i)\}_{i=1}^n$ with a possibly noisy $y_i = f(x_i) + \epsilon_i$, each iteration 1) fits a probabilistic surrogate to D_n, 2) maximizes an acquisition function that scores candidates by the surrogate's mean and uncertainty, 3) evaluates the maximizer, and 4) augments D_n. The incumbent $\hat{y}$ is the best response observed so far. The surrogate throughout this paper is a Gaussian process, $f \sim \mathcal{GP}(0, k)$: conditioned on D_n under a Gaussian likelihood with noise variance σ^2_{lik}, the posterior at a test point x is Gaussian with

$$\mu(x) = \mathbf{k}_x^\top (K + \sigma^2_{\text{lik}} I)^{-1}\mathbf{y}, \qquad \sigma^2(x) = k(x,x) - \mathbf{k}_x^\top (K + \sigma^2_{\text{lik}} I)^{-1}\mathbf{k}_x,$$

where $\mathbf{k}_x = [k(x, x_1), \ldots, k(x, x_n)]^\top$ and K is the covariance matrix of the observed inputs. The kernel sets the correlation structure; we use the Matérn-5/2 kernel, with distance $r = \|x - x'\|$ and lengthscale ℓ, giving $k(x, x') = \big(1 + \sqrt{5}\, r/\ell + 5r^2/(3\ell^2)\big) \exp\big(-\sqrt{5}\, r/\ell\big)$.

Every query maximizes the acquisition over the candidate set (the grid of options for each task).

The three standard acquisitions used in this paper are

- $\text{EI}(x) = \big(\mu(x) - \hat{y}\big)\, \Phi\big(z(x)\big) + \sigma(x)\, \phi\big(z(x)\big), (Mockus,\ 1975;\ Joneset\ al.,\ 1998),$
- $\text{PI}(x) = \Phi\big(z(x)\big), \quad (Kushner,\ 1964)$
- $\text{UCB}(x) = \mu(x) + \sqrt{\beta}\, \sigma(x), (Srinivaset\ al.,\ 2010)$

where $z(x) = (\mu(x) - \hat{y})/\sigma(x)$, with Φ and ϕ the standard normal CDF and PDF and β the confidence parameter of GP-UCB (Srinivas et al., 2010). HAPBO leaves every element of this loop unchanged except the final ranking, which it multiplies by the bounded factor of equation 2.

B THE PRIOR SPECIFICATION LANGUAGE (PSL)

A belief is one JSON document with three parts:

- a `schema` (the typed search space restated by the source),
- a list of `beliefs`,
- a `trust` block.

Table 2 declares the vocabulary (Fig. 6). Three primitives (`bump`, `joint_region`, `anchors`) form a complete basis, exact on finite candidate sets and dense in continuous landscapes, with an explicit anchor-budget rate (App. M) The remaining tokens are macros that name recurring patterns one token each; adding macros cannot reduce expressivity (App. M). Every shaping belief carries a weight $w \in [0, 1]$ and a provenance tag; `anchors` carries neither (its values are exact by definition), and the meta tokens describe the objective rather than the optimum's location.

Compilation of π. In plain terms, the compiler draws every stated shape on the encoded unit cube, adds them up (penalties subtract), keeps the total nonnegative, applies the preference gates, rescales the result to $[0, 1]$, and finally lets exact anchors override the field wherever the source knew the

Table 2: **Declaration of the PSL vocabulary.** Required fields in parentheses; all shaping tokens additionally require `weight` $\in [0, 1]$ and `provenance` $\in$ {`user_stated`, `user_confirmed`, `literature`, `llm_domain`, `llm_default`}. CI fields are 90% intervals (half-width $= 1.645\sigma$).

Token	Fields	Semantics
Base layer		
`baseline`	`level` $[0, 1]$ (default 0.1); `tilt` per-dim $[-1, 1]$	default plausibility floor plus optional global slope
Marginal shapes (one dimension)		
`bump`	(`dim`, `mode`); `ci`; `skew` $[-1, 1]$; `sign` {peak, trough}	single peak or trough; split-Gaussian, σ from the CI
`trend`	(`dim`, `direction` {increase, decrease}); `knee`	monotone lean, optional saturation at the knee
`threshold`	(`dim`, `at`, `active` {above, below}); `sharpness` {soft, sharp}	regime change at a boundary (logistic step)
`periodic`	(`dim`, `period`); `phase`	oscillatory structure (raised cosine)
`categorical`	(`dim`, `preferences` level$\rightarrow [0, 1]$)	multiplicative preference gate over unordered levels
Interactions (two dimensions)		
`joint_region`	(`dims`, `mode` per-dim); `ci` per-dim	localized joint optimum (product of per-dim Gaussians)
`ridge`	(`dims`, `relation` {positive, negative}); `width` {narrow, medium, wide}; `through`	trade-off line through two dimensions
`conditional`	(`when` {dim, region}, `then` nested belief); `mode` {gate, synergy}	one dimension's shape activates within a region of another
Regions and exact knowledge		
`plateau`	(`region` per-dim interval)	indifference band (flat raise)
`avoid`	(`region` box *or* `dim`/`mode`/`ci`)	soft penalized region (subtractive; can never remove a candidate)
`anchors`	(`points` [{coords, value $[0, 1]$}]); `bandwidth` (> 0, default 0.15)	exact values at known points, applied last
Meta tokens (objective properties, not location beliefs)		
`smoothness`	`level` / `per_dim` {very_smooth … very_rough}	suggests GP lengthscale
`noise`	(`level` {low … very_high}); `repeats_recommended`	suggests GP noise floor
`modality`	`expectation` {unimodal, few, many}; `n_optima`	suggests exploration appetite
Trust block (document level)		
`trust`	(`overall_confidence` {very_low, low, medium, high, very_high}); `notes`	source's stated confidence

answer. Algorithm 1 gives the procedure; each shape_b and gate_g is the closed form declared in Table 2.

The normalization extrema are computed over a fixed quasirandom reference set augmented with every belief-declared extremum candidate, so a narrow joint peak cannot be missed by the normalizer. The anchor blend uses compactly supported (Wendland-type) weights with default bandwidth 0.15 and is exact at anchored points, which is what makes the anchor basis exact and dense (App. M). Compilation is deterministic.

Compilation of α (Eq. equation 1 of the main text). α represents the overall confidence assigned to π. Choosing the most appropriate α for a given task can be a standalone research question. The current research is designed in a modular way, and we decided to opt for a simplification for the choice of assigning α. In this paper, the overall confidence maps to $c_{\text{overall}} \in \{0.20, 0.35, 0.50, 0.70, 0.90\}$ for the five levels; each belief's provenance maps to $c_{\text{provenance}} \in \{1.00, 0.90, 0.85, 0.65, 0.40\}$ for:

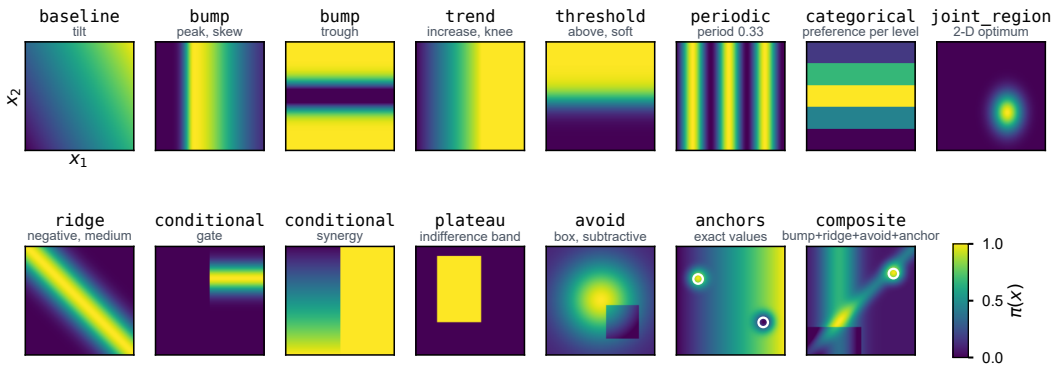


Figure 6: **PSL components as belief functions.** Each panel is a minimal PSL document on a two-dimensional unit square (x_1 horizontal, x_2 vertical), compiled by Algorithm 1 into $\pi(x) \in [0,1]$. In the `categorical` panel x_2 has five unordered levels; in the `anchors` panel the anchor values override a tilted baseline at the circled points. The last panel stacks several components in one document. Meta tokens (`smoothness`, `noise`, `modality`) and the `trust` block describe the objective rather than the optimum's location, so they have no panel.

Algorithm 1 Compilation of the belief function π from a validated PSL document.

Require: validated document $P = (\texttt{schema}, \texttt{beliefs}, \texttt{trust})$; query points x on the encoded unit cube $[0,1]^d$

Ensure: belief function $\pi : \mathcal{X} \to [0,1]$

$s(x) \leftarrow \texttt{baseline.level} + \texttt{tilt}^\top\big(x - \frac{1}{2}\big)$ ▷ base layer

for all additive beliefs $b \in \{\texttt{bump}, \texttt{trend}, \texttt{threshold}, \texttt{periodic}, \texttt{joint_region}, \texttt{ridge}, \texttt{plateau}, \texttt{avoid}\}$ **do**

  $s(x) \leftarrow s(x) \pm w_b \,\mathrm{shape}_b(x)$ ▷ `avoid` and troughs subtract

end for

$s(x) \leftarrow \max\{s(x), 0\}$ ▷ clamp

for all gates $g \in \{\texttt{categorical}$, synergy $\texttt{conditional}\}$ **do**

  $s(x) \leftarrow s(x) \cdot \mathrm{gate}_g(x)$ ▷ gates take values in $[0,1]$

end for

$R \leftarrow$ 8,192 scrambled-Halton points $\cup$ {declared modes, anchor coordinates, corners, center} ▷ reference set

$(\ell, u) \leftarrow \big(\min_{r \in R} s(r),\ \max_{r \in R} s(r)\big)$

$\pi(x) \leftarrow \mathrm{clip}\big((s(x) - \ell)/(u - \ell),\ 0,\ 1\big)$ ▷ min–max normalization

for all anchor sets with points (z_j, v_j) and weights ψ_j **do**

  $\pi(x) \leftarrow \big(1 - \sum_j \psi_j(x)\big)\,\pi(x) + \sum_j \psi_j(x)\, v_j$ ▷ applied last; $\pi(z_j) = v_j$ exactly

end for

return π

- `user_stated` (highest confidence),
- `user_confirmed`,
- `literature`,
- `llm_domain`,
- `llm_default` (lowest confidence).

$\alpha = \mathrm{clip}(c_{\text{overall}} \cdot \langle c_{\text{provenance}} \rangle_w,\ 0.1,\ 0.9)$ with the average weighted by the belief weights. In practice, this leads to a medium-confidence document built entirely on model-internal knowledge, compiling to $\alpha = 0.325$ in most of the present work; only a user-stated, very-high-confidence document raises this value.

Several different choices are possible and the HAPBO method is designed to accommodate a future α selection module.

The original α-πBO work (Guay-Hottin et al., 2025) suggested a default $\alpha = 0.5$ for all priors, which would be a reasonable choice here that would not affect the results qualitatively. The present work uses the more nuanced α of Eq. equation 1 to illustrate that the PSL expressivity can be modulated by design, by specific choice rules (our implementation), or even by LLM feedback.

Meta tokens and the surrogate. `smoothness`, `noise`, and `modality` can configure the GP (per-dimension lengthscales, noise floor, exploration). For a fixed surrogate, the compiled field π enters the noiseless EI guarantee through a multiplicative constant. The separate noisy GP-UCB bound also contains an additive misplaced-trust term, which can grow linearly under static trust (App. M). In all campaigns reported in this paper the meta-token channel is disabled. Future work can investigate the opportunity to further guide the BO approach by having the LLM condition the GP properties. The authors have considered this capacity to be beyond the scope of the current research.

A real elicited document. The prior compiled for XGBoost on the `kc1` task, as returned by the elicitor (numeric values rounded for display):

```
{ "schema": { "objective": "maximize",
              "dimensions": [ {"name": "colsample_bytree", ...},
                              {"name": "eta", ...},
                              {"name": "max_depth", ...},
                              {"name": "reg_lambda", ...} ] },
  "beliefs": [
    { "token": "baseline", "level": 0.1 },
    { "token": "joint_region",
      "dims": ["eta", "max_depth", "colsample_bytree", "reg_lambda"],
      "mode": { "eta": 0.021, "max_depth": 5, "colsample_bytree": 0.8,
                "reg_lambda": 0.46 },
      "ci":   { "eta": [0.0098, 0.099], "max_depth": [3, 8],
                "colsample_bytree": [0.6, 1.0],
                "reg_lambda": [0.099, 10.1] },
      "weight": 1.0, "provenance": "llm_domain",
      "note": "Moderate eta for stability vs speed, shallow-moderate
               depth to avoid overfit, high colsample for
               regularization, mild L2." },
    { "token": "trend", "dim": "eta", "direction": "decrease",
      "weight": 0.4, "provenance": "llm_domain" },
    { "token": "trend", "dim": "max_depth", "direction": "decrease",
      "weight": 0.3, "provenance": "llm_domain" },
    { "token": "smoothness", "level": "medium" },
    { "token": "modality", "expectation": "few" } ],
  "trust": { "overall_confidence": "medium",
             "notes": "Based on general XGBoost tuning heuristics for
                       software defect prediction tasks." } }
```

C TASK CARDS AND ELICITATION PROTOCOL

Card structure. Every source receives the same three-part card: a domain description, the typed search space, and the optimization description (objective, direction, budget, known constraints), preceded by a fixed instruction to express the belief only in the closed PSL vocabulary. Cards are de-identified, they do not name the public benchmark or dataset (the domain is described only by model class and task type), and they carry no hyperparameter guidance or expectations about the response surface. The elicitor must place the prior from its own knowledge. The card for the XGBoost defect-prediction task (parameter value lists abridged):

> *Domain.* System under study: hyperparameter optimization of an XGBoost gradient-boosted-tree classifier on a software-defect-prediction task ($\sim$2109 instances, 21 numeric code-metric features, 2 classes with $\sim$85/15 imbalance). The four knobs are the learning rate `eta` (log), `max_depth` (log), `colsample_bytree` (per-tree feature-subsampling

> fraction), and the L2 leaf regularization `reg_lambda` (log). General gradient-boosting knowledge applies: use general `xgb` hyperparameter-tuning knowledge to place the prior.
>
> *Search space.* `colsample_bytree`: ordinal, 10 linear values in $[0.1, 1.0]$; `eta`: ordinal, 10 log-spaced values in $[2^{-10}, 1]$; `max_depth`: ordinal, $\{1, 2, 3, 5, 8, 13, 20, 32, 50\}$; `reg_lambda`: ordinal, 10 log-spaced values in $[2^{-10}, 1024]$.
>
> *Optimization description.* Objective: mean validation accuracy across the fixed evaluation seeds, higher is better. Direction: maximize. Evaluation budget: about 8 to 32 evaluations (low-budget regime). Known constraints or prior observations: no leaked optimum; place the prior from general hyperparameter-tuning knowledge only.

Human expertise enters as text on the same card. For the cortical mixed (LLM+human) prior, the card's third section embeds the expert's statement verbatim and asks the elicitor to transcribe it:

> A domain expert who has run similar experiments on this preparation states: "Usually the top row is the most effective, and the two outer columns of the array are the most effective." Interpret this as two spatial preferences: (1) the most rostral row … is best; (2) the two edge columns … are more effective than the two middle columns. Encode this in the closed PSL vocabulary […].

The provenance tags of the resulting beliefs (`user_stated` vs. `llm_domain`) record which parts came from the human and set the compiled trust through Eq. equation 1; there is no separate numeric merge step.

For the *human-only prior*, the elicitor is given access to a summary of data from 17 subjects that are not contained in the dataset (as in (Bonizzato et al., 2023)), transcribing expert maps directly.

Adversarial priors. The matched adversarial condition replaces the compiled belief with a programmatically inverted one, which was selected to be a `joint_region` bump placed at the task's empirical anti-optimum (the worst grid configuration), same weight and provenance as the faithful prior. So the adversarial arm consumes exactly the same interface with exactly misplaced content. Baselines that consume text instead of a compiled prior receive a hostile task description through their native interface.

Elicitation protocol. The priors are elicited from a locally served open-weights model (Qwen3.5-122B) at temperature 0, one prior per task, offline, before any optimization. The first attempt uses schema-guided decoding against the PSL grammar; if validation or compilation fails, the exact validator error is fed back for up to four corrective turns (typical cost: one to two calls, $\sim$32,500 tokens total per HPO task). The elicitor-ladder study (App. I) uses the same protocol through a commercial API at temperature 1.0 with ten independent draws per task.

D Experimental details and cost

Surrogate and acquisition. All internal arms (HapBO, its ablation variants, vanilla BO) share a Matérn-5/2 GP. The acquisition is EI with $\xi = 0$, except H, which tests EI/PI/UCB; candidates are the full finite grid of each task; already-queried candidates are excluded from selection in the noiseless benchmarks. $n_{\text{initial}} = 1$ shared random point per seed are identical across HapBO, vanilla, and random search by construction. Seeds 1001–1010. Prior warm-starting is disabled everywhere except the injection-site ablation arm that studies it.

Estimands and normalizations. HPOBench: simple regret against the tabulated grid optimum; figure curves normalize per task by the grid span (optimum minus mean grid score); the adversarial figure normalizes by vanilla's mean first-query regret. Cortical: best-so-far response normalized to the subject optimum, averaged over muscles, then over ten seeds, then clustered over the six rats. Synthetic: per-function regret normalized by vanilla's mean first-query regret and aligned by fraction of each function's budget (Hartmann-6: 40, Styblinski–Tang-7: 50, Levy-10: 50, Ackley-15: 150 queries). SOLAR: best feasible efficiency, zero until the first feasible design.

Synthetic prior fidelity. For each analytic function the elicitor receives a de-named card carrying a perturbed closed-form model of the objective whose correlation with the truth is controlled ($\rho \approx 0.9$, 0.5, 0.1), and transcribes it into PSL under a maximum-fidelity instruction set. Prior quality is therefore an experimental variable; the compiled trust of Eq. equation 1 is used unchanged.

SOLAR constraints. HAPBO fits one GP per constraint and multiplies the (shaped) acquisition by the product of per-constraint feasibility probabilities $\Phi((0-\mu_i)/\sigma_i)$; simulator crashes are modeled by an additional GP whose probability-of-success factor multiplies the score, and crashed observations are excluded from the objective and constraint models. The incumbent is the best feasible, non-crashed observation.

Cost accounting. Table 3 reports per-query wall-clock and token costs on HPOBench. Wall-clock is the median budget-8 run divided by budget on the archived hardware; the external baselines' token figures are single-run measurements; hardware differs across some archived external runs, so these are order-of-magnitude costs, not a controlled benchmark. The locally served model has no marginal provider fee; its compute and energy costs are nonzero.

Table 3: **Per-query cost on HPOBench** (Qwen3.5-122B served locally), averaged across the five families. *Setup tokens* is the one-off prior elicitation, compiled once per task; *tokens/query* is the in-loop LLM cost, zero when the model is consulted only offline.

Method	wall-clock/query (s)	setup tokens (once)	tokens/query
HapBO	0.39	32,500	0
ColaBO	68.00	852	0
LLAMBO (m1)	30.60	–	89,400
LGBO	0.88	–	487
BORA	5.40	–	274
vanilla BO	0.38	–	–
random	0.36	–	–

E FIXED-BELIEF ABLATIONS: TRAJECTORIES AND ENDPOINTS

This appendix reports the two ablations summarized in Sec. 5.4. Both hold the belief fixed, so that differences can be attributed to how a prior is used rather than to what it says: every arm consumes one compiled prior (π, α) per task and shares the same GP surrogate, base EI, candidate set, and matched random initialization (ten seeds); only the stated axis differs.

Injection site (Fig. 7A–C). With the representation fixed, the prior enters through the bounded acquisition multiplier of HAPBO, the floor-free π-BO multiplier, the surrogate mean (prior-to-mean), or the initial queries (prior-warm-start), bracketed by a prior-only greedy arm and no-prior controls. Under a faithful HPO prior the sites coincide; under the matched adversarial prior they separate, with the bounded multiplier recovering best and floor-free π-BO worst.

Representation (Fig. 7D–F). With the site fixed at the bounded multiplier, only the language through which the belief becomes π varies: PSL against point-only, Gaussian-only, and direct-LLM elicitation. The representations perform similarly on the low-dimensional real benchmarks, whereas PSL's richness pays off on trustworthy high-dimensional synthetic priors and turns into a penalty under misleading ones, where the no-prior control is best. Direct-LLM elicitation is reported on the finite benchmarks only, since a fixed coarse anchor grid degenerates in 7–15 dimensions.

F PRIOR SOURCES: COMPILED FIELDS, EXAMPLE MAPS, AND TRAJECTORIES

This appendix reports the prior-source comparison on the NeuroMOSAICS Cortical panel, together with the objects the trajectories consume. The three compiled belief functions (Fig. 8A–C) show how differently the same interface renders the three knowledge sources: the LLM-only prior encodes a smooth somatotopic gradient, the LLM+human prior transcribes the expert's rostral-row and edge-column preferences, and the human-only prior reconciles data from previous subjects (rats tested under the same conditions but not included in the dataset). The example ground-truth response maps (Fig. 8D–F) illustrates the spatial structure and cross-subject variability of the underlying objective; they are shown for context and are not paired with the priors above them. Fig. 8G–I shows each prior mode against its matched vanilla BO trajectory.

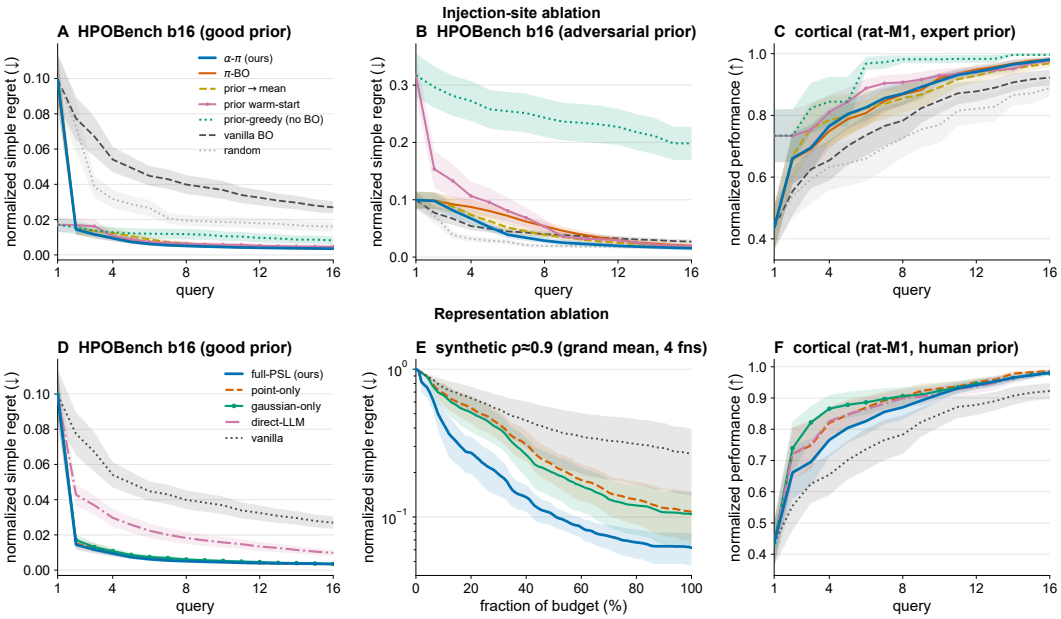


Figure 7: **Prior injection site and prior representation ablations.** One compiled prior (π, α) per task; every arm shares the GP surrogate, base EI, candidate set, and matched random start, and curves are query-wise mean $\pm$ SEM over the 40 HPOBench tasks and six rats. **Top—injection site** (representation fixed, only the site varies; seven arms): **A** faithful HPO prior, **B** matched adversarial HPO prior, **C** cortical with the expert prior. **Bottom—representation** (injection site fixed at α-π, only how the belief becomes π varies; five arms): **D** HPO good prior, grand mean, **E** synthetic $\rho \approx 0.9$, grand mean over the four analytic functions aligned by fraction of budget (direct-LLM excluded, because its fixed coarse anchor set degenerates in 7-15D), **F** cortical with the human prior. Lower is better for regret panels (A,B,D,E), higher for performance panels (C,F).

G HPOBENCH: PER-FAMILY TRAJECTORIES

Figure 9 expands the compact comparison of the main text (Fig. 2) into its five per-family panels. Matched HAPBO consumes the compiled HPO prior from the same random start as its controls, and every method shows its own curve. The low-budget gain of the aggregate is visible within every family.

Figure 10 expands the compact adversarial comparison of the main text (Fig. 5) into its five per-family panels. Each method receives the same hostile task description through its native interface, and matched HAPBO consumes the compiled adversarial prior (a bump at each task's empirical anti-optimum) from the same random start as its controls. The recovery visible in the aggregate holds in every family. HAPBO pays for the misleading belief over the first queries and converges back toward the strongest prior-free curves by query 16, so the main-text conclusion is not driven by one model class.

H HAPBO ACROSS BO VARIANTS

The bounded multiplier attaches to the acquisition landscape (it fundamentally re-sorts its ranking) and nothing else, so it should be indifferent to the BO machinery around it: which acquisition is shaped, and whether queries are issued one at a time or in batches. This appendix verifies both on matched pairs; Figure 11 collects the evidence (panels A–C: acquisition functions; D–E: batch selection).

Acquisition functions (Fig. 11A–C). The bounded multiplier applies to any nonnegative base acquisition. Panels A–C repeat the matched prior-on/prior-off contrast under EI, PI, and UCB on

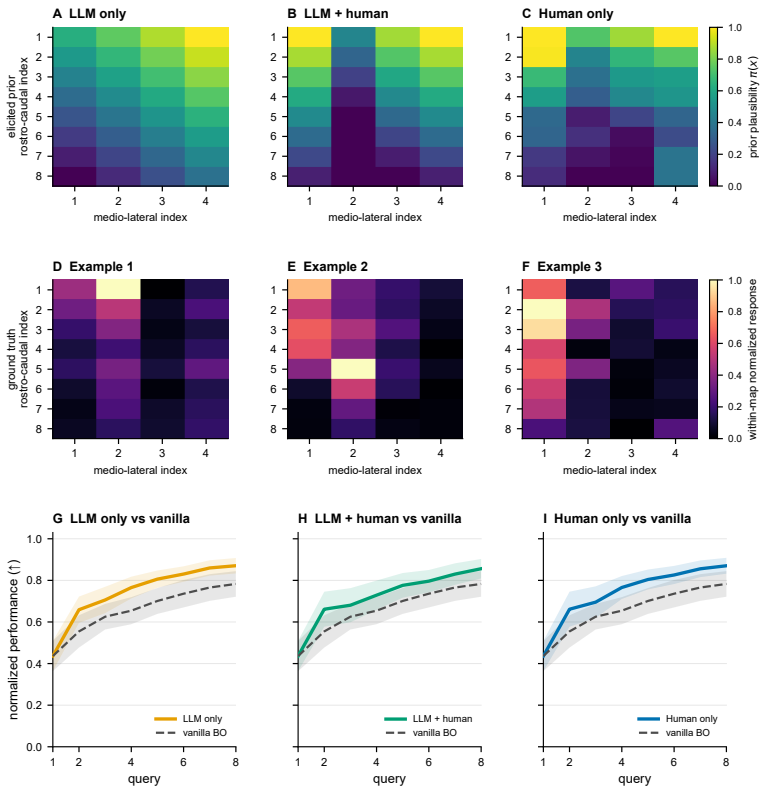


Figure 8: **Prior source and cortical optimization, complete panel. A–C:** compiled belief functions for LLM-only, LLM+human, and human-only knowledge. **D–F:** three example ground-truth response maps from the NeuroMOSAICS dataset, each normalized independently to expose spatial structure; they are not paired with the priors above. **G–I:** each prior mode against matched vanilla BO over eight queries; mean ± SEM clustered over six rats after averaging muscles and ten seeds.

HPOBench and the cortical tasks. Every arm shares the GP surrogate, candidate set, and matched seeded initialization ($n_{\text{initial}} = 1$); only the prior differs. Raw UCB is an absolute score that can fall below the incumbent, so its α-π arm shapes the incumbent-relative confidence bound, keeping its behavior consistent with the nonnegative EI and PI improvements. Because the paired arms share the same seeded first query, their query-1 values coincide exactly (normalized regret 0.078 on HPOBench, normalized performance 0.436 on cortical), so every subsequent divergence is attributable to shaping alone. Across all three acquisitions the shaped arm improves low-budget sample efficiency: on HPOBench the final normalized regret drops from $0.023/0.031/0.017$ (vanilla EI/PI/UCB) to $0.002/0.003/0.002$, a 9–12$\times$ reduction; on cortical the advantage is largest at low budget (query-3 normalized performance $0.70/0.82/0.73$ vs. $0.63/0.52/0.63$) and narrows toward the budget end.

Batch selection (Fig. 11D–E). The multiplier is likewise indifferent to whether queries are issued one at a time or in parallel batches (an often likely setting when several electrodes or configurations are evaluated together). We keep the matched EI contrast of panels A–C and replace sequential selection with a Kriging-Believer batch of size $q = 3$: each round takes the shaped-EI argmax, fantasizes its outcome as the posterior mean, refits, and repeats until the batch is full, then evaluates all three points. The first member of a batch sees exactly the sequential arm's data, so divergence appears only inside a batch, where the surrogate is frozen across picks. The shaped arm's advantage is preserved in both settings.

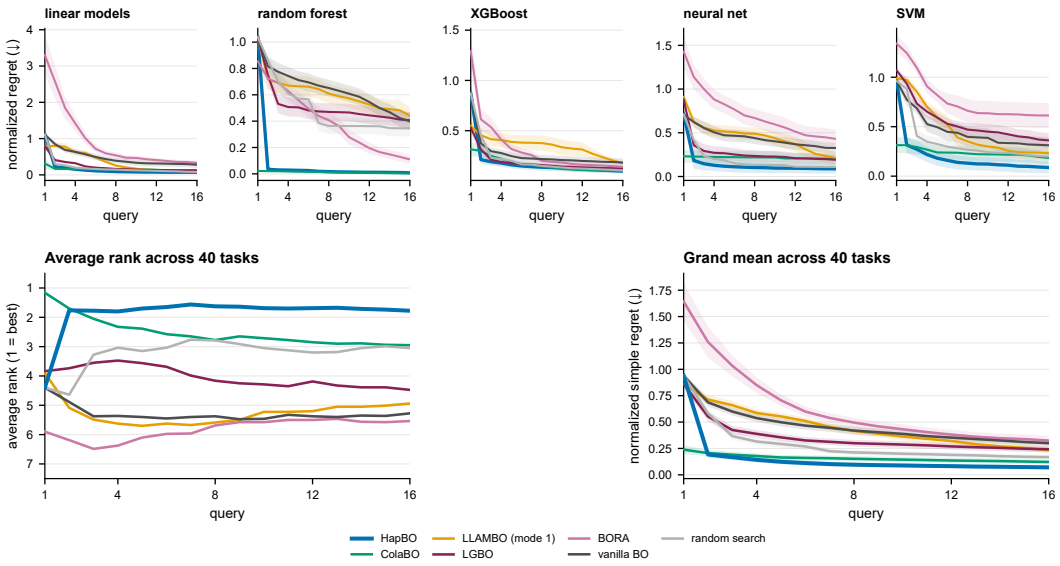


Figure 9: **HPOBench, per model family.** Normalized simple regret for the five families (top), average task-wise rank and grand mean over 40 tasks (bottom), using matched HAPBO and every method's native budget-16 curve. Regret is normalized per task by the full-grid span (optimum minus mean grid score); bands are ±SEM across tasks after averaging ten seeds within each task. LLAMBO omits ∼32 transient run failures. The family panels show the aggregate gain of the main text is not driven by one model class.

I ELICITATION LADDER: ELICITOR QUALITY VS. DOWNSTREAM PERFORMANCE

Does a stronger elicitor produce a more useful prior? Figure 12 runs identical α-πBO with priors elicited by different LLMs on one HPOBench task per family (`kc1`). Three models of one commercial family form a capability ladder (Claude Opus 4.8 (Anthropic, 2026a) > Claude Sonnet 4.6 (Anthropic, 2026b)> Claude Haiku 4.5 (Anthropic, 2025)), each re-elicited over ten independent draws at temperature 1.0; the open-weights prior (Qwen3.5-122B) and the no-prior vanilla baseline are shown for reference from the main figure. The final regrets are presented: the frontier models match or edge the Qwen3.5-122B prior, whereas the weakest sits close to no prior. The performance and cost per run are presented in Table 4. A caveat is that the weakest tested elicitor (Haiku 4.5) is least reliable, and failed to produce a usable prior on 22% of elicitations. No failure was encountered by using Qwen3.5-122B throughout the paper (nor with the other Claude-family elicitors).

Table 4: **Elicitor ladder: downstream regret and elicitation cost** (HPOBench `kc1`, per family). Columns lr–svm: final simple regret (lower is better), mean over ten elicitation draws. *fail%*: fraction of the fifty attempted elicitations that yielded no usable prior. *Elicit. tokens*: mean one-off token total per task; *elicit. cost*: its dollar cost at the provider's prices (the local model has no marginal fee). Bold marks the best value per regret column.

Elicitor	lr	rf	xgb	nn	svm	fail%	elicit. tokens	elicit. cost
Opus 4.8	0.0029	**0.0000**	0.0019	0.0068	**0.0000**	0.0	34,000	$0.20
Sonnet 4.6	0.0021	0.0003	**0.0007**	0.0077	**0.0000**	0.0	32,000	$0.12
Haiku 4.5	**0.0019**	0.0015	0.0013	0.0068	0.0084	22.0	44,000	$0.05
Qwen (local)	0.0020	0.0009	0.0020	**0.0048**	0.0012	0.0	29,000	$0 (local)
vanilla (no prior)	0.0023	0.0209	0.0014	0.0324	0.0056	–	–	–

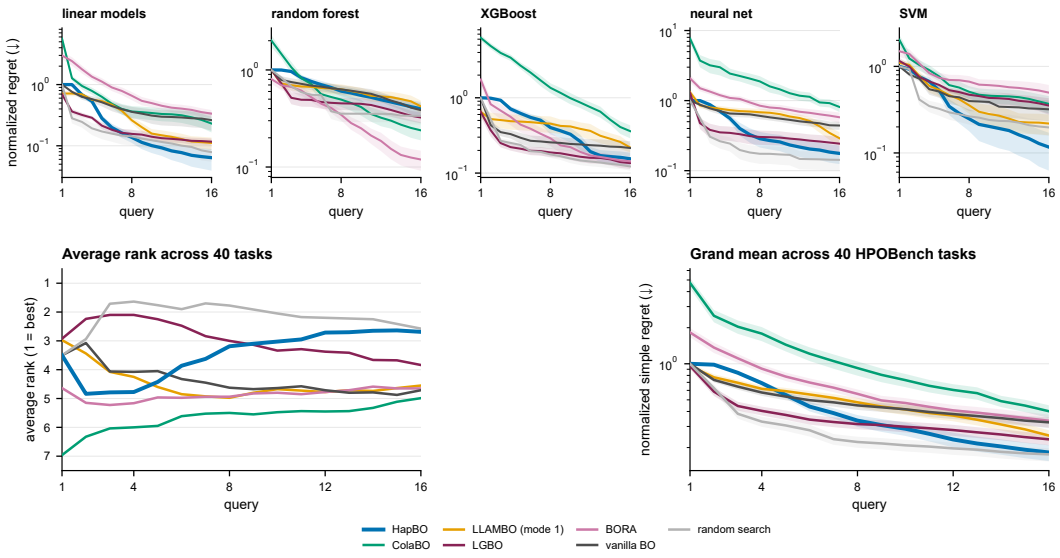


Figure 10: **Adversarial HPOBench, per model family.** Normalized simple regret for the five families (top), average task-wise rank and grand mean over 40 tasks (bottom), using matched HAPBO and the same hostile description through each method's native interface. The rank panel uses the task grid complete for every method; regret is normalized by vanilla's mean first-query regret; bands are ±SEM across tasks after averaging ten seeds. BORA completes 400/400, LLAMBO 370/400, and ColaBO 400/400 task–seed endpoints. The family panels show the aggregate recovery of the main text is not driven by one model class.

J TARGETED CHOICES IN BASELINE IMPLEMENTATIONS

Prior-guided and in-loop baselines admit more than one faithful implementation, and the choice can move a curve as much as a change of method. Methods like LLAMBO and LGBO are designed to be modular and modified in future work. This is the case also for the current HAPBO, as discussed above, since the choice of α and metatokens for the surrogate is subject to welcome variations from future work. For the two baselines LLAMBO and LGBO where we experienced the most open ambiguity at the time of integration, we performed a direct comparison in a restricted HPOBench setting, holding the harness identical (same local model, task cards, GP where used, candidate set, initialization, seeds, budget). Then, we adopted and reported in the main text the variant that performed best, so every headline comparison is against the strongest competitor we could construct (Fig. 13).

LLAMBO: initialization mode. The original paper presents random initialization (mode 1) and zero-shot warm-starting with five LLM-proposed points (mode 2). Mode 1 is markedly more sample-efficient through the low-budget regime (mode 2's warm-start proposals begin below even a random draw); we therefore report mode 1.

LGBO: confidence operator. No public code reproduces the LGBO (as defined in the original paper) on these benchmarks, so we reconstruct it, and two readings exist: the paper's stated construction lifts the GP mean by exactly c posterior standard deviations (nonnegative, capped), whereas the public release maps confidence through $\delta = 3\,\Phi^{-1}(c)$, which is negative for $c < 0.5$. This suppresses the very region the model prefers, and unbounded as $c \to 1$. We port both into the identical harness; the paper-equation version is stronger on both the standard prior and the adversarial prior; we therefore report our paper-equation version.

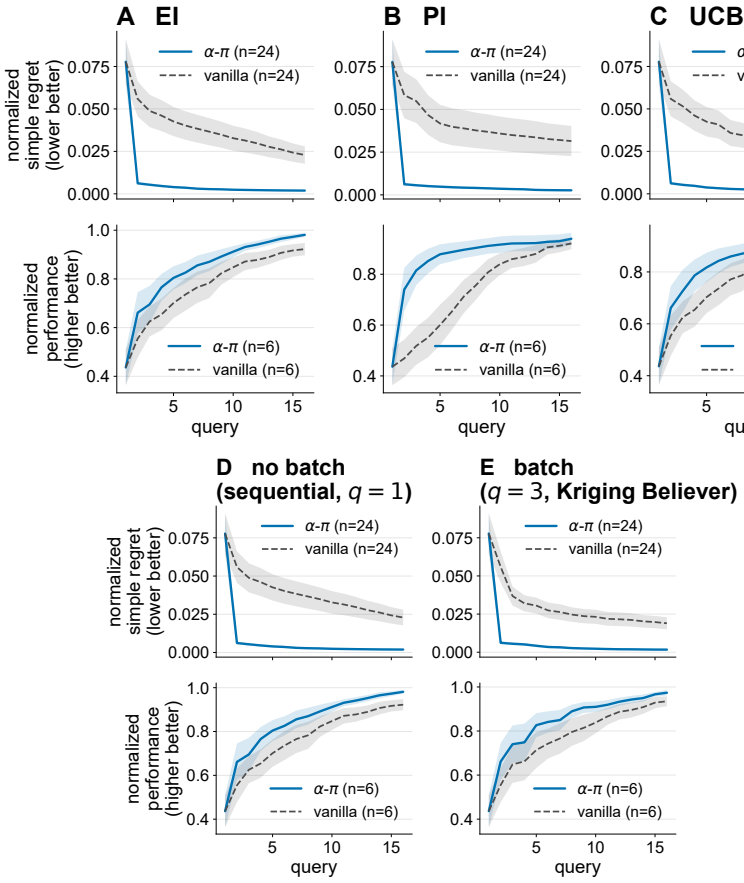


Figure 11: **Bounded shaping across BO variants** (matched arms; only the stated ingredient varies; details and conditions in the text of this appendix). **A–C, acquisition functions:** matched α-π vs. vanilla BO under EI, PI, and UCB; top rows HPOBench (normalized simple regret, lower is better; 24 lr/rf/xgb tasks), bottom rows cortical (normalized performance, higher is better; six rats); ten seeds. **D–E, batch selection:** the same matched EI contrast under sequential (q=1) and batch (q=3, Kriging Believer) selection; D reproduces the EI panels of A.

K LEAK CONTROL

The objective of this paper is to determine, when LLM knowledge about a given problem exists, whether this can be integrated, by proper harnessing, in BO. However, there is a type of existing knowledge that is less desirable when testing any LLM-prior method on public benchmarks, which is contamination: a strong prior might reflect recall of the final benchmark answer rather than transferable expertise. We address this with two complementary controls: a direct probe of what the elicitor natively knows, and an additional objective whose optimum cannot have been memorized.

K.1 DOES THE ELICITOR NATIVELY KNOW THE OPTIMUM?

For every task we present the task card and ask the model, from prior knowledge alone, for its single best guess of the optimum's location. We score the guess with the metric *recovered* $\% = (v - \bar{v})/(v^\star - \bar{v})$, the fraction of the head-room between the grid mean and the optimum that the blind guess captures. We read the reasoning trace and flag recall only when the model both names the underlying benchmark and states its optimum. For the analytic functions we probe the de-named perturbed-model cards at all three fidelities, score against the *original* optimum, and add a positive control that names the function outright.

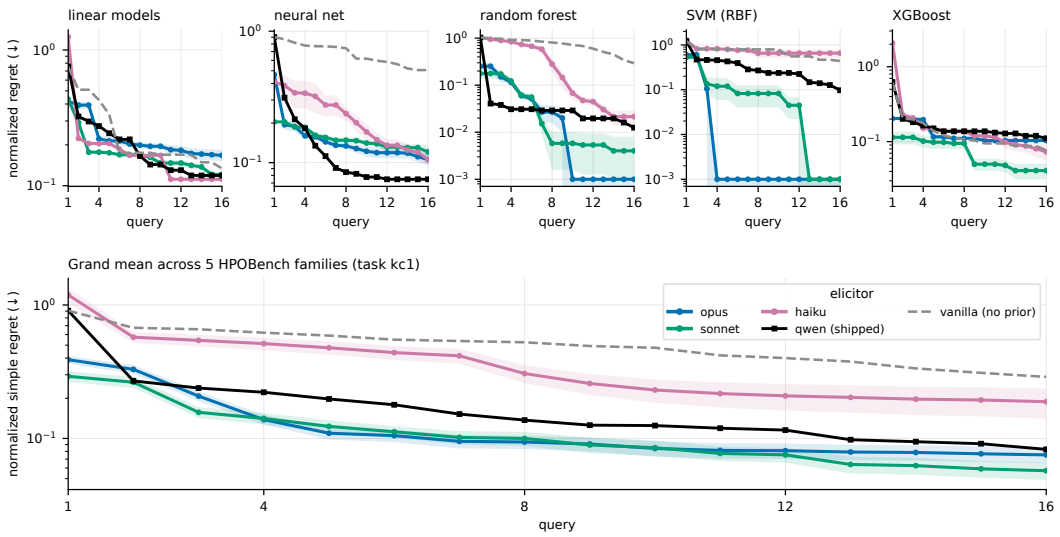


Figure 12: **Downstream performance and elicitor quality.** Normalized simple regret (regret over grid span, log axis floored at 10^{-3}) vs. query on HPOBench task `kc1`, per family (top) and pooled over the five families (bottom). Colored curves: ladder elicitors, query-wise mean $\pm$ SEM over ten elicitation draws (temperature 1.0). Black: the main-text Qwen3.5-122B prior; dashed grey: no prior; both are query-wise means over ten seeds, shown band-less. The prior is the only difference between arms

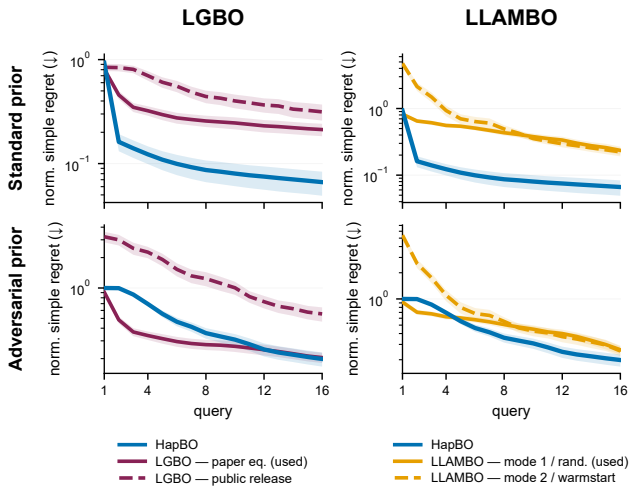


Figure 13: **Baseline-implementation sensitivity.** LGBO and LLAMBO, in two possible implementations, each shown with the matched HAPBO curve. **Left, LGBO:** the paper's mean-shift equation (solid, used in the main text) vs. the public release's confidence operator (dashed). **Right, LLAMBO:** random initialization, mode 1 (solid, main text) vs. zero-shot warm-start, mode 2 (dashed). **Top:** standard prior, grid-span-normalized regret (linear). **Bottom:** matched adversarial prior, vanilla-query-1-normalized regret (log). Bands are $\pm$SEM across tasks after averaging ten seeds.

No elicitor natively knows the optima from the cards alone (Tables 5–7). On HPOBench the blind guess recovers 49–77% of the head-room; on the cortical grid it lands *below* the map average (-20%); no trace on any real task shows recall. On the analytic functions the guess tracks the fidelity of the supplied model and the de-named cards are not recognized, with one exception: the elicitor recognizes the Ackley family at every fidelity yet defers to the perturbed card rather than to a

memorized optimum, so its guess still degrades with ρ. The named-function control recovers 100% in every case, confirming the knowledge exists but surfaces only when explicitly requested. Across the elicitor ladder the picture holds on all five HPO families: a stronger model guesses marginally better blind (grand-mean recovered 66% to 77%) but none of the 400 elicitations recalls an optimum, so the downstream elicitor-quality trend of App. I does not reflect memorization.

Table 5: **Probing native recall of benchmark optima.** Recovered and raw % of the blind best-guess optimum per benchmark (local elicitor, greedy decoding), with reasoning-trace recall evidence. HPO values are means over the eight tasks per family ($\pm$SEM); cortical over the sixteen rat–muscle maps; SOLAR is a single problem scored against the best feasible pool design.

Problem (family)	recovered %	recall evidence
HPOBench lr	64$\pm$8	no recall evidence (0/8)
HPOBench rf	49$\pm$11	no recall evidence (0/8)
HPOBench xgb	71$\pm$7	no recall evidence (0/8)
HPOBench nn	77$\pm$8	no recall evidence (0/8)
HPOBench svm	70$\pm$10	no recall evidence (0/8)
NeuroMOSAICS cortical	-20$\pm$4	no recall evidence (0/16)
SOLAR	93	no recall evidence (0/1)

Table 6: **Probing native recall of benchmark optima, analytic functions.** De-named cards supply a perturbed model at three fidelities ρ; the guess is scored against the *original* optimum (recovered % and normalized RMS distance to $x^\star$; $0 =$ exact). The named control names the function outright.

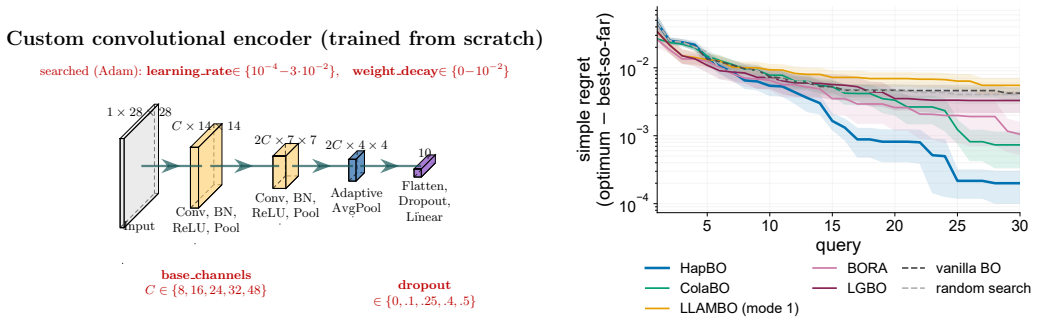
Function	card	recovered %	dist. to $x^\star$	recall evidence
Hartmann-6	$\rho\approx0.9$	70	0.08	follows provided model, no family recognition
	$\rho\approx0.5$	30	0.24	follows provided model, no family recognition
	$\rho\approx0.1$	-6	0.35	follows provided model, no family recognition
	named control	100	0.00	names Hartmann, states $x^\star$
Styblinski–Tang-7	$\rho\approx0.9$	58	0.55	follows provided model, no family recognition
	$\rho\approx0.5$	42	0.52	follows provided model, no family recognition
	$\rho\approx0.1$	-175	0.65	follows provided model, no family recognition
	named control	100	0.00	names Styblinski–Tang, states $x^\star$
Levy-10	$\rho\approx0.9$	99	0.02	follows provided model, no family recognition
	$\rho\approx0.5$	90	0.10	follows provided model, no family recognition
	$\rho\approx0.1$	-157	0.41	follows provided model, no family recognition
	named control	100	0.00	names Levy, states $x^\star$
Ackley-15	$\rho\approx0.9$	96	0.00	recognizes Ackley family, drops it to follow directives
	$\rho\approx0.5$	12	0.15	recognizes Ackley family, drops it to follow directives
	$\rho\approx0.1$	-2	0.45	recognizes Ackley family, drops it to follow directives
	named control	100	0.00	names Ackley, states $x^\star$

Table 7: **Probing native recall of benchmark optima across models.** Recovered % of the blind best-guess per HPOBench family (ladder elicitors at temperature 1.0, three draws per task; local elicitor greedy). No elicitation on any family recalls an optimum.

Elicitor	lr	rf	xgb	nn	svm	grand mean	recall evidence
Qwen3.5-122B	64±8	49±11	71±7	77±8	70±10	66±4	no recall evidence (0/40)
Claude Opus	75±4	70±7	72±3	82±3	84±4	77±2	no recall evidence (0/120)
Claude Sonnet	61±6	70±7	72±3	84±3	75±5	72±2	no recall evidence (0/120)
Claude Haiku	66±4	48±6	76±2	84±3	71±6	69±2	no recall evidence (0/120)

K.2 An un-memorizable CNN benchmark

We define a small bespoke convolutional encoder (Fig. 14A) and compute its hyperparameter-response surface ourselves: a 4-D grid of $6 \times 5 \times 5 \times 5 = 750$ configurations over learning rate, base channel width, dropout, and weight decay, each network trained from scratch and scored by seed-averaged validation accuracy on a fixed small image subsample. This exact architecture-and-grid response map exists in no public corpus, so the optimum is not recoverable from memory; a competent elicitor can still place a reasonable prior from general machine-learning intuition, which is the ability we mean to isolate. On this objective the ordering of the main text is preserved (Fig. 14B): matched-initialization HapBO reaches the lowest simple regret, ahead of ColaBO, BORA, and LGBO; the prior-free controls trail, and LLAMBO's in-loop guidance does not improve on them here.



(A) (B)

Figure 14: **Leak-control benchmark. (A)** the bespoke convolutional encoder optimized as an un-memorizable objective, with the four searched hyperparameters highlighted. **(B)** query-wise simple regret (log axis, mean ± SEM over ten seeds) on the 750-configuration grid. The optimum is a property of our network, grid, and training protocol and appears in no public corpus, so no method's prior can reflect memorization.

L Two further baselines: CAKE and CALIPER

Two recent LLM-driven methods keep the model in the loop in ways orthogonal to the prior-weighting baselines of the main text; we run both in our harness on the HPOBench suite (five families, ten seeds), under the faithful and the matched adversarial prior. **CAKE** (Suwandi et al., 2025) uses the LLM as a crossover/mutation operator that evolves the GP kernel expression at each query; it consumes no compiled belief, so its curve is identical in both prior conditions. **CALIPER** (Chen et al., 2026a) elicits per-role priors once and admits them through a discrete counterfactual evidence gate; we adapt it to single-objective HPO (three objective personas; EI in place of the hypervolume acquisition) and, absent public code, reimplement it from the paper. HapBO attains the lowest grand-mean regret under both conditions (Fig. 15); CAKE and CALIPER are competitive under a faithful prior but do not overtake it, and CALIPER, whose gate trusts the elicited roles, degrades most when the prior is adversarial, whereas the bounded multiplier keeps HapBO robust.

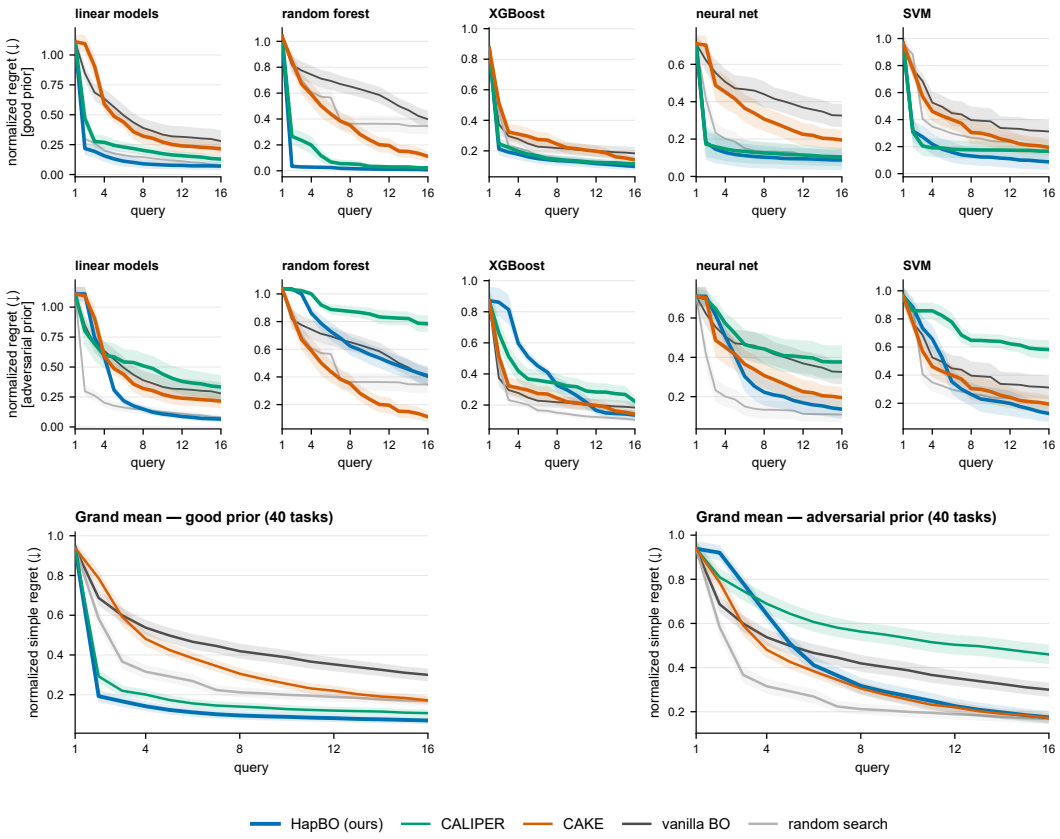


Figure 15: **CAKE and CALIPER vs. HAPBO on HPOBench.** Normalized simple regret (lower is better) per model family under a faithful prior (top) and a matched adversarial prior (middle), with grand means over the 40 tasks (bottom; faithful left, adversarial right). All arms share the GP surrogate, candidate set, and matched random initialization (ten seeds); CAKE is prior-agnostic, so the same curve appears in both conditions.

M THEORY

This appendix contains the formal analysis behind HAPBO: the expressivity and organization of the Prior Specification Language (PSL), the acquisition-level guarantee of the bounded multiplier, and two separate convergence analyses: 1) noiseless fixed-kernel EI and 2) noisy finite-domain GP-UCB. The two convergence regimes are analyzed independently.

Notation. The search space $\mathcal{X} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_d$ is a typed product: each coordinate is a compact interval after the schema's scale transform (e.g. log), a finite ordinal set, or a finite categorical set; $\mathcal{X}$ is compact, and a finite-candidate implementation acts on a grid $D = \{x_1, \ldots, x_N\} \subseteq \mathcal{X}$. The objective is f, maximized unless stated otherwise; $X^\star = \arg\max_x f(x)$ (nonempty on compact $\mathcal{X}$ for continuous f, always on finite D).

Three regret quantities. We distinguish:

- the *instantaneous regret* $r_t = f(x^\star) - f(x_t)$ of the point queried at round t;
- the *best-sampled latent simple regret* $S_T = \min_{t \le T} r_t$, the regret of the best point *queried* within the budget, evaluated on the latent objective (under noise, $f(x_t)$ is not observed exactly, so S_T is not directly observable);
- the *recommendation regret* $f(x^\star) - f(\hat{x}_T)$ of a point $\hat{x}_T$ recommended from a noisy history.

Each result below controls exactly one of these: the EI theorem controls best-sampled simple regret in a noiseless setting (where it is observable); the UCB theorem controls cumulative instantaneous regret $R_T = \sum_{t \le T} r_t$, and through $S_T \le R_T/T$ the best-sampled latent simple regret.

M.1 PSL COMPILER: EXPRESSIVITY AND VOCABULARY STRUCTURE

The results of this subsection characterize the compiler independently of any optimizer.

M.1.1 TYPED DOMAIN AND COMPILER CONTRACT

A PSL document P (typed schema; primitives: anchors, bumps, joint regions, trends, thresholds, periodic, ridges, conditionals, avoid regions, plateaus, categorical weights; trust metadata) compiles as $\mathcal{C}(P) = (\pi_P, \alpha_P, F_P)$, where $\pi_P : \mathcal{X} \to [0,1]$ is the prior belief function, $\alpha_P \le \alpha_{\max} < 1$ is the compiled trust, and $F_P \subseteq \mathcal{X}$ is the feasible set. The semantics clauses used by the theory:

- **(S1)** *Finite anchors.* On finite D, an anchor at x_i with value $p_i \in [0,1]$ sets the compiled prior at x_i to exactly p_i, with no renormalization or clipping; candidates not covered by any anchor take the baseline value.
- **(S2)** *Continuous anchors.* On continuous coordinates, for any finite anchor set $z_1, \dots, z_K$ forming a δ-cover, the compiler realizes a continuous partition of unity subordinate to that cover: $\pi_P(x) = \sum_j v_j \psi_j(x)$ with $\psi_j \ge 0$, $\sum_j \psi_j = 1$, and $\psi_j(x) > 0$ only when $\mathrm{dist}(x, z_j) < \delta$.
- **(S3–S6)** *Macro semantics.* The baseline compiles to an affine function; a trend to a piecewise-linear saturating ramp; a periodic token to a sinusoidal envelope mapped into $[0,1]$; a bump or avoid to a signed Gaussian $\pm w \exp(-(x-\mu)^2/(2\sigma^2))$ with weight $w \in [0,1]$; a joint region to the corresponding multi-dimensional product.
- **(S7)** *Feasibility.* No belief primitive removes a candidate. Hard limits are part of *defining* the search space: they enter through the schema's typed per-dimension ranges (and, for a residual joint limit, a caller-supplied feasibility filter), fixing F_P before any belief is compiled. The soft `avoid` primitive subtracts from the score, which is then clamped to $[0,1]$; it can lower a region's plausibility but never removes it from F_P.

Theorem 1 (PSL expressivity)**.** *Let the compiler satisfy (S1)–(S2).*

- *(a)* ***Finite exactness.*** *For every $p \in [0,1]^N$ on a finite candidate set $D = \{x_1, \dots, x_N\}$ there is an anchors-only PSL document P with $\pi_P(x_i) = p_i$ for all i.*
- *(b)* ***Continuous density.*** *For every continuous $g : \mathcal{X} \to [0,1]$ (categorical dimensions as finite strata) and every $\varepsilon > 0$ there is an anchors-only document P with $\sup_{x \in \mathcal{X}} |\pi_P(x) - g(x)| < \varepsilon$.*
- *(c)* ***Quantitative rate.*** *If the continuous block of a stratum satisfies $S \subseteq \prod_{i=1}^{d_c} [0, D_i]$ and g is L-Lipschitz there, accuracy ε is achieved with $K \le \prod_i \lceil D_i \sqrt{d_c}\, L/\varepsilon \rceil$ anchors per stratum; equivalently, the sup-error in the anchor budget K is $O(K^{-1/d_c})$.*

Proof. (a) One anchor per candidate with value p_i; clause (S1) returns the table exactly, and each $p_i \in [0,1]$ needs no clipping.

(b) Fix a continuous stratum S (compact). Uniform continuity gives $\delta > 0$ with $\mathrm{dist}(x,y) < \delta \Rightarrow |g(x) - g(y)| < \varepsilon$. Take a finite δ-cover $z_1, \dots, z_K$ of S and the partition of unity of clause (S2); set anchor values $g(z_j)$. Then

$$|\pi_P(x) - g(x)| = \Big| \sum_j \psi_j(x) \big(g(z_j) - g(x) \big) \Big| \le \sum_j \psi_j(x)\, |g(z_j) - g(x)| < \varepsilon,$$

since $\psi_j(x) > 0$ forces $\mathrm{dist}(x, z_j) < \delta$ and the ψ_j sum to one. Repeat per categorical stratum, gating anchor sets by the categorical assignment. The interpolant is a convex combination of values in $[0,1]$, hence a valid prior field.

(c) Partition the bounding box into axis-aligned cells of side $\delta/\sqrt{d_c}$ with $\delta = \varepsilon/L$; each cell has diameter δ, so anchors placed one per cell meeting S form a δ-cover of size at most the stated product. Running the argument of (b) with this cover bounds the sup-error by $L\delta = \varepsilon$; inverting the cell count gives the $O(K^{-1/d_c})$ form. □

Proposition 1 (vocabulary organization)**.** *Let $(\mathcal{X}, \mu)$ be a product probability space and $g \in L^2(\mathcal{X}, \mu)$. Define recursively $g_\emptyset = \mathbb{E}[g]$ and $g_S(x_S) = \mathbb{E}[g \mid x_S] - \sum_{T \subsetneq S} g_T(x_T)$. Then $g = \sum_{S \subseteq \{1,\dots,d\}} g_S$ in L^2 with $\langle g_S, g_T \rangle = 0$ for $S \neq T$, and the order-2 truncation $g_{\leq 2} = g_\emptyset + \sum_i g_i + \sum_{i<j} g_{ij}$ satisfies*

$$\|g - g_{\leq 2}\|_2^2 = \sum_{|S|>2} \|g_S\|_2^2.$$

PSL's layers mirror this decomposition: the baseline is the order-0 component, the marginal primitives the order-1 components, the interaction primitives the order-2 components, and anchors the residual complete basis.

Proof. The recursion gives $\mathbb{E}[g_S \mid x_T] = 0$ for every strict subset $T \subsetneq S$; the product structure of μ then yields orthogonality across distinct index sets, and the recursive sum telescopes to g. For the truncation, $g - g_{\leq 2} = \sum_{|S|>2} g_S$ and orthogonality gives the Pythagorean identity. □

Limitation. Proposition 1 explains the vocabulary's organization; it does not show that low-order structure dominates any particular elicited prior.

Remark 1 (macros cannot reduce expressivity)**.** *Enlarging the vocabulary with macro tokens cannot reduce what is expressible: any document written in a sub-vocabulary is a document of the full vocabulary, so exactness and density (Theorem 1) are retained. The macro tokens exist for economy of expression, not for additional expressive power.*

M.2 Common bounded-shaping results

Definition 1 (raw bounded shaping)**.** *Let $A_t : F_P \to \mathbb{R}$ be a base acquisition at round t. The shaped acquisition is*

$$A_t^M(x) = A_t(x)\, M_t(x), \qquad M_t(x) = 1 - \alpha_t + \alpha_t\, \pi_t(x), \qquad m := 1 - \alpha_{\max} > 0, \tag{4}$$

with $\pi_t(x) \in [0,1]$ and $\alpha_t \in [0, \alpha_{\max}]$. The default is a run-fixed field π and scalar static trust α compiled before the run; time-varying trust α_t is introduced only for the decay results below. The implementation clamps $\alpha_{\max} = 0.9$, so $1/m \leq 10$. Note that the multiplier is applied to the raw *acquisition: the argmax of a multiplicatively shaped acquisition is not invariant to per-step affine rescaling, so selection rules that min–max normalize the acquisition before multiplying (the form in which Eq. 10 of Guay-Hottin et al., 2025 is stated) are different rules and are not covered by the results below.*

Proposition 2 (bounded influence on acquisition selection)**.** *For M_t as in Eq. equation 4: $m \leq M_t(x) \leq 1$ for all x and t. Consequently, if $A_t \geq 0$ on the feasible set and $x_t \in \arg\max_x A_t(x) M_t(x)$, then*

$$A_t(x_t) \;\geq\; m \max_x A_t(x).$$

Proof. $M_t = 1 - \alpha_t(1 - \pi_t)$ with $\alpha_t(1 - \pi_t) \in [0, \alpha_{\max}]$ gives the bounds. For the selection statement, $A_t(x_t) \geq A_t(x_t) M_t(x_t) \geq A_t(x) M_t(x) \geq m\, A_t(x)$ for every x, using $M_t(x_t) \leq 1$, optimality of x_t, and $M_t(x) \geq m$. □

M.3 Regime I: noiseless fixed-kernel expected improvement

This subsection follows the minimization convention of the classical EI analysis (Bull, 2011): f is minimized (equivalently $-f$ is maximized); the statements are orientation-free. L_n denotes the nonnegative best-sampled simple regret after n evaluations, observable here because observations are noiseless.

Assumption 1 (EI regime)**.** *(i) Observations are deterministic and noiseless; (ii) the domain is compact; (iii) the kernel, lengthscale, and scale are fixed during the run, with smoothness $\nu < \infty$; (iv) the objective lies in the RKHS ball $\|f\|_{\mathcal{H}_\ell(\mathcal{X})} \le R$; (v) the acquisition is raw EI, shaped as in Definition 1 with a run-fixed multiplier $M \in [m, 1]$, continuous in x so the shaped argmax exists; (vi) no sequential GP hyperparameter refitting. In addition, the two selection-dependent ingredients of the classical machinery hold for* any *selection sequence: the covering argument of Bull (2011, Lemma 7) bounds the posterior standard deviation at all but at most p rounds by $C\, p^{-(\nu\wedge 1)/d} \log^{\kappa} p$, and the running-optimum telescoping bounds the incumbent decrement $z_n^* - f(x_{n+1}) > 2Rp^{-1}$ at no more than p rounds; a pigeonhole over the window $[p, 3p]$ yields an index $n_p \in [p, 3p]$ satisfying both bounds simultaneously. Finally, EI satisfies the two-sided sandwich (Bull, 2011, Lemma 8): with $I = (f(x_n^*) - f(x))^+$, $s = s_n(x; \ell)$, and $\tau(z) = z\Phi(z) + \phi(z)$,*

$$\max\Big(I - Rs,\ \tfrac{\tau(-R/\rho)}{\tau(R/\rho)}\, I\Big) \ \le\ \mathrm{EI}_n(x) \ \le\ I + (R+\rho)\, s. \tag{5}$$

For a run-fixed multiplier M define

$$M(\mathrm{opt}) := \max_{x^\star \in X^\star} M(x^\star), \qquad \Lambda(M) := \frac{\sup_x M(x)}{M(\mathrm{opt})} \ \le\ \frac{1}{m}, \tag{6}$$

using $\sup_x M \le 1$ and $M(\mathrm{opt}) \ge m$; note $\Lambda(M) \ge 1$ always.

Theorem 2 (shaped-EI rate preservation)**.** *Under Assumption 1, the best-sampled simple regret of shaped EI satisfies, for $3p \le n < 3(p+1)$ with p past the initial design,*

$$L_n(\mathrm{EI}^M) \ \le\ \Lambda(M)\, \frac{\tau(R/\rho)}{\tau(-R/\rho)} \Big(2Rp^{-1} + C(R+\rho)\, p^{-(\nu\wedge 1)/d} \log^{\kappa} p\Big) \ =\ \Lambda(M)\, B_n(\mathrm{EI}), \tag{7}$$

where $B_n(\mathrm{EI}) = O\big(n^{-(\nu\wedge 1)/d} \log^{\kappa} n\big)$ denotes the Bull-style theoretical upper bound on the loss of unshaped EI—an upper bound, not the realized loss of a separately executed vanilla run. In particular $L_n(\mathrm{EI}^M) \le (1/m)\, B_n(\mathrm{EI})$: bounded shaping preserves the classical rate, with a constant determined by the multiplier at an optimizer.

Proof. Fix p past the initial design and let $n_p \in [p, 3p]$ be the pigeonhole index of Assumption 1. Let $x^\star$ be a global minimizer maximizing M over $X^\star$. By the lower branch of equation 5 at $x^\star$, with $I_{n_p}(x^\star) = z^*_{n_p} - z^*$,

$$\mathrm{EI}_{n_p}(x^\star) \ \ge\ \frac{\tau(-R/\rho)}{\tau(R/\rho)}\, (z^*_{n_p} - z^*).$$

By optimality of the shaped selection at step n_p, $\mathrm{EI}_{n_p}(x^\star) M(x^\star) \le \mathrm{EI}_{n_p}(x_{n_p+1}) M(x_{n_p+1})$, hence

$$\mathrm{EI}_{n_p}(x^\star) \ \le\ \mathrm{EI}_{n_p}(x_{n_p+1})\, \frac{M(x_{n_p+1})}{M(x^\star)} \ \le\ \Lambda(M)\, \mathrm{EI}_{n_p}(x_{n_p+1}).$$

By the upper branch of equation 5 at x_{n_p+1} and the two pigeonhole bounds,

$$\mathrm{EI}_{n_p}(x_{n_p+1}) \ \le\ \big(z^*_{n_p} - f(x_{n_p+1})\big)^+ + (R+\rho)\, s_{n_p}(x_{n_p+1}) \ \le\ 2Rp^{-1} + C(R+\rho)\, p^{-(\nu\wedge 1)/d} \log^{\kappa} p.$$

Chaining the three displays and using monotonicity $z_n^* \le z^*_{n_p}$ for $n \ge n_p$ (in particular for $3p \le n < 3(p+1)$, since $n_p \le 3p$) gives equation 7. □

Corollary 1 (scalar prior-quality constant)**.** *For scalar trust α and run-fixed prior π with miss $\beta_\star = 1 - \max_{x^\star \in X^\star} \pi(x^\star)$ at the optimizer,*

$$L_n(\mathrm{EI}^M) \ \le\ \frac{1}{1 - \alpha\beta_\star}\, B_n(\mathrm{EI}).$$

Proof. $M(\mathrm{opt}) = 1 - \alpha(1 - \pi(x^\star)) = 1 - \alpha\beta_\star$ and $\sup_x M \le 1$ in equation 6. □

Corollary 2 (correct at an optimizer)**.** *If $\pi(x^\star) = 1$ at some global optimizer, then $\Lambda(M) = 1$: the shaped run satisfies the* same *theoretical upper bound as unshaped EI, at any trust level, regardless of the belief elsewhere. This is a statement about the bound; it does not assert that the realized shaped trajectory dominates a realized vanilla-EI trajectory.*

Proof. $M(\mathrm{opt}) = 1$ and $\sup_x M \le 1$ give $\Lambda = 1$ in equation 6. □

Corollary 3 (elicitation–compilation error)**.** *Let $g : \mathcal{X} \to [0,1]$ be the source's latent belief landscape with miss $\beta_g = 1 - g(x^\star)$ at a global optimizer, and suppose the elicitation-plus-compilation pipeline produces π with $\epsilon = \|\pi - g\|_\infty$. Then for scalar trust α,*

$$\Lambda(M) \;\le\; \frac{1}{1 - \alpha\,\min(1,\ \beta_g + \epsilon)}.$$

The pipeline enters the constant through two scalars added together—the source's miss at the optimizer and the sup-norm compilation error—and the penalty never exceeds the ceiling $(1-\alpha_{\max})^{-1}$.

Proof. $\pi(x^\star) \ge g(x^\star) - \epsilon$ gives $1 - \pi(x^\star) \le \beta_g + \epsilon$, and $1 - \pi(x^\star) \le 1$ always; insert into equation 6 via $M(\mathrm{opt}) = 1 - \alpha(1 - \pi(x^\star))$. □

Remark 2 (time-varying trust)**.** *If the trust schedule satisfies $\sup_x \alpha_t(x) \to 0$, the same argument applied at the pigeonhole iteration shows that the effective constant approaches one asymptotically for any prior. Decay is not required for convergence in this noiseless EI regime, the rate is preserved at any fixed $\alpha \le \alpha_{\max}$.*

M.4 Regime II: noisy finite-domain raw GP-UCB

Back to the maximization convention.

Assumption 2 (UCB regime)**.** *(i) Finite candidate set D with $|D| = N$; kernel k with $k(x,x) \le 1$, fixed during the run, noise model fixed; (ii) observations $y_t = f(x_t) + \varepsilon_t$ with noise variance $\sigma_n^2 > 0$ in the GP update; (iii) the acquisition is raw GP-UCB, $U_t(x) = \mu_{t-1}(x) + \sqrt{\beta_t}\,\sigma_{t-1}(x)$ with β_t nondecreasing—no incumbent-relative and no min–max-normalized variant; (iv) shaped selection $x_t \in \arg\max_{x \in D} U_t(x) M_t(x)$ with history-measurable $M_t \in [m, 1]$; (v)* re-querying of candidates is allowed *(no exclusion mask), so the optimum remains selectable at every step; (vi) $f(x^\star) \ge 0$, a normalization treated as part of the algorithm (satisfied when responses are nonnegative, or after subtracting a known lower bound; the multiplicative rule is not shift-invariant, so the shift is part of the method's definition). Two standard facts are imported.* Confidence event*: there are standard choices of β_t such that with probability at least $1 - \delta$,*

$$|f(x) - \mu_{t-1}(x)| \;\le\; \sqrt{\beta_t}\,\sigma_{t-1}(x) \qquad \text{for all } x \in D \text{ and all } t \ge 1 \tag{8}$$

(for f sampled from the GP, $\beta_t = 2\log(N\pi^2 t^2/(6\delta))$, Srinivas et al., 2010; for f in the kernel's RKHS with sub-Gaussian noise, the self-normalized construction of Chowdhury & Gopalan, 2017, valid for any adapted selection sequence). Call the event in equation 8 E. Information-gain bound*: for any selection sequence, with β_t nondecreasing and $k \le 1$,*

$$\sum_{t=1}^{T} 4\beta_t \sigma_{t-1}^2(x_t) \;\le\; C_1\,\beta_T\,\gamma_T, \qquad C_1 = \frac{8}{\log(1 + \sigma_n^{-2})}, \tag{9}$$

where γ_T is the maximum information gain after T rounds (Srinivas et al., 2009, Lemma 5.4).

Lemma 1 (finite-domain information gain)**.** *On finite D with $k \le 1$: $\gamma_T \le \frac{N}{2}\log\bigl(1 + T\sigma_n^{-2}/N\bigr) = O(N \log T)$.*

Proof. The information gain of any T observations is $\frac{1}{2}\log\det(I + \sigma_n^{-2} K_T)$ with K_T the $T \times T$ kernel matrix of the queried points. K_T is positive semidefinite of rank at most N (all columns lie in the span of the N candidate feature vectors), so it has at most N nonzero eigenvalues $\lambda_i \ge 0$ with $\sum_i \lambda_i = \mathrm{tr}(K_T) \le T$. Then $\log\det = \sum_{i \le N} \log(1 + \sigma_n^{-2}\lambda_i) \le N \log\bigl(1 + \sigma_n^{-2}(\sum_i \lambda_i)/N\bigr)$ by concavity of $\log(1+u)$ (Jensen), which is at most $N \log(1 + \sigma_n^{-2} T/N)$. □

Define the per-step *misplaced-trust penalty at the optimum*

$$\theta_t \;:=\; \min_{x^\star \in X^\star} \frac{1 - M_t(x^\star)}{M_t(x^\star)} \;=\; \frac{1}{M_t(\mathrm{opt})} - 1, \qquad 0 \le \theta_t \le \frac{1-m}{m},$$

and $\bar\theta_T = \max_{t \le T} \theta_t$.

Theorem 3 (noisy shaped GP-UCB regret)**.** *Under Assumption 2, on the event E, the cumulative regret $R_T = \sum_{t\le T}\big(f(x^\star) - f(x_t)\big)$ of shaped GP-UCB satisfies*

$$R_T \;\le\; f(x^\star)\sum_{t=1}^{T}\theta_t \;+\; (1+\bar\theta_T)\,\sqrt{C_1 T\beta_T\gamma_T}. \tag{10}$$

The second term is the usual information-gain term of unshaped GP-UCB, inflated by at most $1+\bar\theta_T \le 1/m$; the first, additive term is the cumulative price of misplaced trust and depends on the belief only through its value at the optimum*: confident but wrong beliefs about regions that do not contain the optimum contribute nothing to the bound.*

Proof. Fix t and work on E. Let $x^\star$ attain the minimum in θ_t.

Step 1 (optimism through the multiplier). By equation 8, $U_t(x^\star) \ge f(x^\star) \ge 0$. By shaped optimality and $M_t > 0$,

$$U_t(x_t)M_t(x_t) \;\ge\; U_t(x^\star)M_t(x^\star) \;\ge\; f(x^\star)M_t(x^\star) \;\ge\; 0,$$

so $U_t(x_t) \ge 0$, and $f(x^\star) \le U_t(x_t)\,M_t(x_t)/M_t(x^\star) \le U_t(x_t)(1+\theta_t)$, using $M_t(x_t) \le 1$ and $1/M_t(x^\star) = 1+\theta_t$.

Step 2 (value at the query). By equation 8, $f(x_t) \ge \mu_{t-1}(x_t) - \sqrt{\beta_t}\sigma_{t-1}(x_t) = U_t(x_t) - 2\sqrt{\beta_t}\sigma_{t-1}(x_t)$.

Step 3 (per-step regret). Subtracting, $r_t \le \theta_t U_t(x_t) + 2\sqrt{\beta_t}\sigma_{t-1}(x_t)$. By equation 8 again, $U_t(x_t) \le f(x_t) + 2\sqrt{\beta_t}\sigma_{t-1}(x_t) \le f(x^\star) + 2\sqrt{\beta_t}\sigma_{t-1}(x_t)$, so

$$r_t \;\le\; \theta_t f(x^\star) + (1+\theta_t)\,2\sqrt{\beta_t}\,\sigma_{t-1}(x_t).$$

Step 4 (sum). Sum over $t \le T$, bound $1+\theta_t \le 1+\bar\theta_T$, and apply Cauchy–Schwarz with equation 9: $\sum_t 2\sqrt{\beta_t}\sigma_{t-1}(x_t) \le \sqrt{T\sum_t 4\beta_t\sigma_{t-1}^2(x_t)} \le \sqrt{C_1 T\beta_T\gamma_T}$. □

Corollary 4 (correct at an optimizer)**.** *If $\pi_t(x^\star) = 1$ for all t at some optimizer, then $M_t(\mathrm{opt}) = 1$, $\theta_t = 0$, and shaped GP-UCB satisfies the* vanilla *GP-UCB bound $R_T \le \sqrt{C_1 T\beta_T\gamma_T}$ on E—at any trust level, however wrong the belief is elsewhere.*

Proof. $\theta_t = 0$ collapses equation 10. □

Corollary 5 (decaying trust)**.** *If $\alpha_t \le \min(\alpha_{\max},\, c/t)$, then $\theta_t \le (c/m)/t$ and, on E,*

$$R_T \;\le\; \frac{c\,f(x^\star)}{m}\,(1+\log T) \;+\; \frac{1}{m}\,\sqrt{C_1 T\beta_T\gamma_T}\,:$$

the additional cumulative term is $O(\log T)$ for any prior, however adversarial; cumulative regret is sublinear under the usual information-gain conditions (on finite D, $\beta_T\gamma_T = O(N\log^2 T)$ by Lemma 1); and the best-sampled latent simple regret vanishes, since $S_T \le R_T/T \to 0$.

Proof. $\theta_t = (1 - M_t(\mathrm{opt}))/M_t(\mathrm{opt}) \le \alpha_t(1-\pi_t(x^\star))/m \le (c/t)/m$; $\sum_{t\le T} 1/t \le 1+\log T$; and $1+\bar\theta_T \le 1 + (1-m)/m = 1/m$. The simple-regret statement is $\min_{t\le T} r_t \le R_T/T$. □

Proposition 3 (static misplaced trust can cost linear cumulative regret)**.** *For every $m \in (0,1)$, there is a two-point problem in the noisy regime of Assumption 2 and a valid run-fixed multiplier $M \in [m,1]$ such that, on the confidence event E, shaped GP-UCB has*

$$R_T = (1-c)T - O(\log T), \qquad m < c < 1.$$

Thus static shaping can incur linear cumulative regret even with positive observation noise.

Proof. Let $D = \{a,b\}$, $f(a) = 1$, and $f(b) = c$, where $m < c < 1$. Use the fixed kernel $k(x,x') = \mathbf{1}\{x = x'\}$, a zero GP prior mean, and independent Gaussian observation noise of variance $\sigma_n^2 > 0$, with the same variance in the posterior update. Set $M(a) = m$ and $M(b) = 1$. These multipliers are realized by $\alpha = 1-m$, $\pi(a) = 0$, and $\pi(b) = 1$. Use the shaped selection rule from the first round and allow re-querying.

The objective belongs to the kernel's RKHS, with $\|f\|_{\mathcal{H}_k(D)} = \sqrt{1+c^2}$. Choose the nondecreasing RKHS confidence schedule from Assumption 2; on this fixed two-point domain it can be chosen with $\beta_t = O(\log(t+1))$ and $\beta_t \to \infty$, since $\gamma_t = O(\log(t+1))$. Work on its confidence event E.

Write $N_a(t-1)$ for the number of observations of a before round t. The independent-arm kernel gives

$$\sigma^2_{t-1}(a) = \frac{\sigma_n^2}{\sigma_n^2 + N_a(t-1)}.$$

Whenever the shaped rule selects a, optimality and confidence imply

$$mU_t(a) \geq U_t(b) \geq c, \qquad U_t(a) \leq 1 + 2\sqrt{\beta_t}\,\sigma_{t-1}(a).$$

Consequently,

$$N_a(t-1) \leq \frac{4m^2\sigma_n^2\beta_t}{(c-m)^2} - \sigma_n^2.$$

Using monotonicity of β_t and accounting for the last selection of a (with $N_a(T) = 0$ if it is never selected), we obtain

$$N_a(T) \leq 1 + \frac{4m^2\sigma_n^2\beta_T}{(c-m)^2} = O(\log T).$$

Each query of b incurs regret $1-c$, whereas a query of a incurs zero regret. Therefore

$$R_T = (1-c)\big(T - N_a(T)\big) = (1-c)T - O(\log T).$$

□

Three comments on Proposition 3. First, it establishes that linear cumulative regret can occur within the positive-noise regime of Theorem 3. The linear dependence on T is therefore a real worst-case possibility, although the construction does not establish sharpness of the coefficient in the upper bound. Second, Corollaries 4 and 5 give sufficient conditions for recovering sublinear cumulative regret: correctness at an optimizer or decaying trust. They are not necessary conditions; for example, a positive multiplier that is constant across candidates leaves the vanilla GP-UCB selection rule unchanged. The default HAPBO configuration keeps trust static within a run and accepts the resulting worst-case trade-off, because deployment campaigns of 8–30 queries are judged by the best configuration found rather than by cumulative regret. Third, the proposition does not imply a failure of discovery. In this construction, a must eventually be queried on E: otherwise $U_t(a) = \sqrt{\beta_t} \to \infty$, while repeated observations of b give $c \leq U_t(b) \leq c + 2\sqrt{\beta_t}\sigma_n/\sqrt{\sigma_n^2 + t - 1} \to c$, contradicting the shaped selection rule. Hence the best-sampled latent simple regret is zero after the first query of a, despite linear cumulative regret. This does not establish a recommendation guarantee from noisy observations.