CALIBO: CALIBRATING INCREMENTAL LLM ADVICE FOR BAYESIAN OPTIMIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Large language models (LLMs) have shown promise for augmenting Bayesian optimization (BO), particularly when objective evaluations are scarce. When both an LLM and a Gaussian process (GP) provide objective predictions, a central question is how their means should influence acquisition. We introduce CaliBO, which calibrates LLM guidance by estimating its incremental predictive value beyond the current GP. Using predictions recorded before each evaluation, CaliBO learns whether the disagreement between the LLM and GP means predicts the GP's subsequent residual error. A signed, shrunk coefficient produces a bounded mean correction, while GP posterior uncertainty remains unchanged for exploration. This allows CaliBO to follow, downweight, or reverse LLM advice according to its estimated incremental value, without requiring the LLM to outperform the GP as a standalone predictor. We bound acquisition perturbation and, under a zero-usefulness null and stated GP and noise-estimation assumptions, establish a sublinear advice-dependent term in the regret bound of an idealized GP-LCB variant. Across 49 hyperparameter-optimization tasks, CaliBO is competitive with ten baselines and significantly outperforms several of them. Ablation studies support the effectiveness of incremental calibration, while controlled experiments with synthetic advisors demonstrate benefits from signed correction under systematically inverted advice and reveal a failure case when uninformative advice still tracks the GP's error.

1 INTRODUCTION

Black-box optimization with expensive evaluations is common in engineering workflows such as machine learning hyperparameter optimization (Snoek et al., 2012; Falkner et al., 2018) and design-parameter tuning (Ghose et al., 2025; Chang et al., 2025). Bayesian optimization (BO) (Mockus et al., 1978; Brochu et al., 2010) addresses such problems by learning a surrogate model from limited observations and using an acquisition function to select promising evaluations (Shahriari et al., 2016). In low-data regimes, large language models (LLMs) offer a promising source of guidance through their prior knowledge and ability to learn from observations in context. Recent work has incorporated LLMs into different parts of the BO loop, including warm-starting, candidate generation, surrogate prediction, and proposal refinement (Liu et al., 2024; Chang et al., 2025; Chen et al., 2026; Zhang et al., 2023).

An important distinction, however, is between predicting objective values and estimating the uncertainty of those predictions. LLAMBO (Liu et al., 2024) demonstrates that LLMs can provide useful objective predictions in low-sample regimes, but also reports limitations in their uncertainty calibration relative to GP surrogates. These findings motivate a complementary use of the two models: exploiting the predictive information supplied by the LLM while retaining GP posterior uncertainty. Yet the GP also provides its own predictive mean, leaving a central question: *when the LLM and GP disagree, how should their means influence acquisition?*

Existing approaches make different choices about how LLM predictions enter BO (Table 1). LLAMBO uses an LLM surrogate directly for acquisition (Liu et al., 2024), whereas LLINBO uses LLMs primarily for candidate proposal while retaining GP-based scoring (Chang et al., 2025). LABO calibrates LLM predictions against realized outcomes (Chen et al., 2026), while multi-information-source methods model the relationship between auxiliary and target sources jointly (Poloczek et al.,

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.

Table 1: How representative methods incorporate LLM or auxiliary predictive guidance into acquisition. The key distinction is what determines the influence of the auxiliary signal.

Method	Mean used in acquisition	Basis of influence	GP-based uncertainty
LLAMBO (Liu et al., 2024)	LLM	Direct LLM surrogate	No
LLINBO (Chang et al., 2025)	GP	GP filter on LLM proposals	Yes
LABO (Chen et al., 2026)	Scaled LLM + discrepancy GP	LLM prediction–outcome fit	Yes[†]
MISO (Poloczek et al., 2017)	Joint multi-source mean	Joint source covariance	Yes[†]
ColaBO (Hvarfner et al., 2024)	GP reweighted by user prior	Prior specification	Yes
CaliBO	**GP + bounded correction**	**LLM–GP disagreement → GP residual**	**Yes**

[†] Includes a GP component, but not the unchanged base-GP posterior.

2017). However, when the LLM and GP are conditioned on the same observation history, the LLM's standalone agreement with outcomes does not directly measure what it contributes beyond the GP.

To answer this, we introduce CaliBO, which calibrates LLM guidance by estimating its incremental predictive value beyond the current GP. Before each objective evaluation, CaliBO records the disagreement between the LLM and GP means at the selected point. Once the outcome is observed, it pairs this disagreement with the residual of the GP prediction recorded before that evaluation. A signed, shrunk coefficient learned from these pairs produces a bounded correction to the mean used for acquisition, while GP posterior uncertainty remains unchanged for exploration. This allows CaliBO to follow, downweight, or reverse LLM advice according to its estimated incremental value, without requiring the LLM to outperform the GP as a standalone predictor.

Our contributions are as follows.

- **Incremental LLM calibration.** We formulate LLM guidance as a correction relative to the current GP and calibrate its signed influence from LLM–GP disagreement and subsequent GP residuals. This provides an online estimate of the direction and strength of the LLM's incremental predictive contribution beyond the GP.
- **Controlled influence with theoretical guarantees.** We bound the acquisition perturbation induced by the calibrated correction and, for an idealized GP-LCB variant, establish a sublinear advice-dependent term in the regret bound under a zero-usefulness null and stated GP and noise-estimation assumptions.
- **Empirical validation.** Across 49 real-world HPO tasks, CaliBO is competitive with ten baselines and significantly outperforms several of them. Ablations support incremental calibration, while controlled experiments with synthetic advisors show that signed correction can exploit systematically inverted advice and identify uninformative advice whose disagreement still tracks the GP's error as a failure regime.

2 Related Work

LLM-augmented BO. LLMs have been used in BO for warm starts and candidate generation (Zhang et al., 2023; Mahammadli & Ertekin, 2024; Yin et al., 2024; Yang et al., 2025), GP-filtered proposal mechanisms (Chang et al., 2025; Feng et al., 2025), surrogate prediction (Liu et al., 2024), and auxiliary-source calibration or evidence gating (Chen et al., 2026; Chen & Yi, 2026), and their usefulness depends on the domain and the model (Kristiadi et al., 2024). These approaches differ in where LLM guidance enters the loop and how its influence is controlled, and Table 1 summarizes the methods closest to our setting. CaliBO focuses on calibrating the incremental mean information an LLM supplies relative to the current GP.

Auxiliary sources and predictor combination. Classical BO uses auxiliary predictions through joint or discrepancy models of multiple sources (Kennedy & O'Hagan, 2001; Poloczek et al., 2017; Mikkola et al., 2023), transfer weights fitted to the current task (Feurer et al., 2018; Wistuba et al., 2018; Shilton et al., 2017), and priors with fixed or decaying influence (Hvarfner et al., 2022; 2024; Adachi et al., 2024). More broadly, forecast combination, expert advice and learning-augmented algorithms weight predictors by their realized errors (Bates & Granger, 1969; Cesa-Bianchi & Lugosi, 2006; Lykouris & Vassilvitskii, 2018; Purohit et al., 2018). CaliBO belongs to this tradition, but uses the GP as an anchor and applies a signed correction along the LLM–GP disagreement direction.

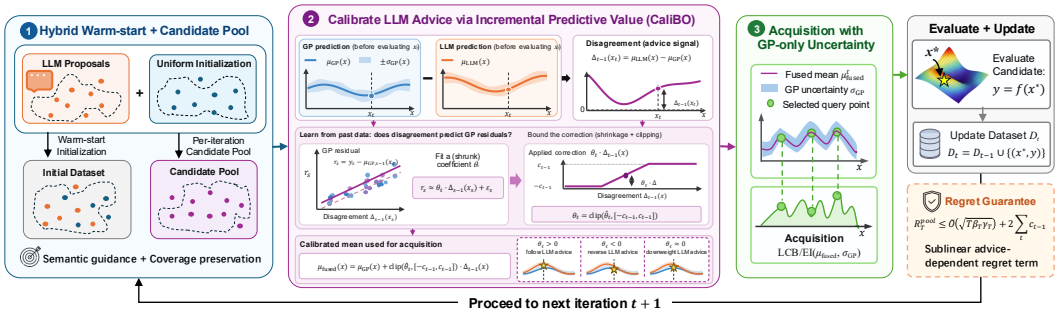

Figure 1: Illustration of CaliBO. The LLM–GP disagreement at the selected point is recorded before evaluation and paired with the GP residual observed after it. The learned coefficient produces a bounded mean correction, while GP uncertainty is unchanged by the advice layer.

What differs is the evidence and the setting. The weight is fitted to whether the disagreement predicts the GP's residual rather than to agreement with outcomes, records are fixed before their outcomes while BO decides where data are collected (hence the self-normalized analysis of Abbasi-Yadkori et al. 2011), the weight may be negative, and its effect on acquisition is bounded.

3 Method

3.1 Problem Setting and Mean Correction

We minimize an expensive black-box function $f : \mathcal{X} \to \mathbb{R}$ over a bounded domain $\mathcal{X} \subseteq \mathbb{R}^d$, observing $y_t = f(\mathbf{x}_t) + \varepsilon_t$ under a total budget of T evaluations, with history $\mathcal{D}_{t-1} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{t-1}$ when the t-th query is selected. After $t-1$ observations, a GP surrogate (Rasmussen & Williams, 2006) provides the posterior mean $\mu_{\mathrm{GP},t-1}(\mathbf{x})$ and standard deviation $\sigma_{\mathrm{GP},t-1}(\mathbf{x})$, with an observation-noise level $\hat{\sigma}^2_{\varepsilon,t-1}$ refit by marginal likelihood at every round. An LLM prompted with a task description, the search space and the most recent observations of $\mathcal{D}_{t-1}$ provides a numerical prediction $\mu_{\mathrm{LLM},t-1}(\mathbf{x})$ for every candidate. Both predictions are formed before the corresponding evaluation and are expressed in objective units. Because the two predictors draw on overlapping history, part of the LLM prediction may reproduce what the GP already captures. The advice signal is therefore the disagreement

$$\Delta_{t-1}(\mathbf{x}) \;=\; \mu_{\mathrm{LLM},t-1}(\mathbf{x}) - \mu_{\mathrm{GP},t-1}(\mathbf{x}), \tag{1}$$

which vanishes wherever the LLM agrees with the surrogate.

Calibrated mean correction. CaliBO augments GP-based BO with a single correction to the mean used for acquisition. The GP remains the reference predictor and supplies the exploration uncertainty, and the LLM enters only through Δ. At round t, for each candidate $\mathbf{x}$ of a finite pool $\mathcal{C}_t$ (Section 3.4), the acquisition mean is

$$\mu_{\mathrm{fused},t-1}(\mathbf{x}) = \mu_{\mathrm{GP},t-1}(\mathbf{x}) + q_{t-1}(\mathbf{x}), \quad q_{t-1}(\mathbf{x}) \;=\; \mathrm{clip}\big(\hat{\theta}_{t-1}\,\Delta_{t-1}(\mathbf{x}),\, [-c_{t-1}, c_{t-1}]\big), \tag{2}$$

where the coefficient $\hat{\theta}_{t-1}$ sets the direction and strength of the correction (Section 3.2) and the budget c_{t-1} limits its magnitude (Section 3.3). Neither quantity modifies the GP uncertainty, and every constant is fixed a priori (Appendix C.8).

3.2 Calibrating Disagreement Against GP Residuals

The disagreement is a candidate correction, not evidence that the LLM is more reliable. CaliBO evaluates it against the residual of the GP prediction made before the outcome was observed. We write t for the round being decided and $s < t$ for earlier rounds. When round s selects $\mathbf{x}_s$, it records, before the evaluation is issued,

$$a_s \;=\; \Delta_{s-1}(\mathbf{x}_s), \qquad b_s \;=\; \mu_{\mathrm{GP},s-1}(\mathbf{x}_s), \tag{3}$$

and once y_s is observed the record settles as the correction the GP mean turned out to need,

$$r_s \;=\; y_s - b_s. \tag{4}$$

The pair (a_s, r_s) links what the LLM proposed changing to what the observation says the GP got wrong. Because the regressor is the disagreement rather than the LLM prediction itself, advice identical to the GP has $a_s = 0$ and earns nothing, however well it matches outcomes (Remark 5). Because each record is written before y_s exists and is never recomputed afterwards, an observed outcome cannot enter its own score, and calibration reuses the evaluations BO already selects.

With m settled records, the coefficient is a shrunk least-squares slope through the origin,

$$\hat{\theta}_{t-1} \;=\; \frac{\sum_s a_s r_s}{\sum_s a_s^2} \cdot \frac{m}{m+\kappa} \;\text{ if } m \geq m_{\min} \text{ and } \textstyle\sum_s a_s^2 > \varepsilon_a, \qquad \hat{\theta}_{t-1} = 0 \;\text{ otherwise,} \tag{5}$$

with $m_{\min} = 5$, $\kappa = 4$ and a numerical tolerance $\varepsilon_a = 10^{-12}$. A positive coefficient moves the mean in the direction suggested by the LLM, a negative coefficient reverses that direction, and zero leaves the GP mean unchanged. The coefficient is not a convex-mixture weight. The shrinkage keeps a few early records from receiving the full least-squares weight, and the absence of an intercept keeps the GP's own bias from being credited to the advisor.

3.3 Bounded Mean Guidance and Acquisition

The coefficient estimates how the disagreement relates to the GP residual. The budget serves a separate purpose, limiting the resulting change to acquisition scores. With the settled residuals $e_s = r_s - \hat{\theta}_{t-1} a_s$ and their mean square $V_{t-1}^{\text{res}} = \frac{1}{m}\sum_s e_s^2$,

$$c_{t-1} \;=\; \lambda_c \sqrt{\left(V_{t-1}^{\text{res}} - \hat{\sigma}^2_{\varepsilon,t-1}\right)_+} \;\text{ if } m \geq m_{\min}, \qquad c_{t-1} = 0 \;\text{ otherwise,} \qquad \lambda_c = 2, \tag{6}$$

the noise-deflated scale of the residual the calibrated correction leaves unexplained. The clip in Eq. (2) then guarantees $|q_{t-1}(\mathbf{x})| \leq c_{t-1}$ at every candidate, whatever the advisor outputs. The budget is deliberately conservative, and Remark 11 discusses its behavior under a highly accurate advisor. The next query minimizes a lower confidence bound built on the corrected mean and the GP width,

$$\mathbf{x}_t = \arg\min_{\mathbf{x}\in\mathcal{C}_t} \; \mu_{\text{fused},t-1}(\mathbf{x}) - \sqrt{\beta_t}\,\sigma_{\text{GP},t-1}(\mathbf{x}), \qquad \beta_t = \max\Big(1,\; 2\log\tfrac{|\mathcal{C}_t|\,\pi^2 t^2}{6\delta}\Big), \tag{7}$$

where β_t is the standard GP-LCB confidence parameter (Srinivas et al., 2010) at confidence level $1-\delta$, unrelated to $\hat{\theta}$. The advice layer changes only the mean term, so the influence of numerical LLM advice on acquisition is isolated in q. Section 4 bounds this perturbation without assuming that every correction is beneficial.

3.4 Hybrid Initialization and Candidate Generation

Candidate generation is complementary to mean calibration. CaliBO uses LLM proposals for task-semantic guidance together with uniform samples for coverage, both at initialization and in the per-round pool, since LLM proposals can be correlated or concentrated.

Hybrid warm start. Before a GP can be fit, CaliBO evaluates $n_0 = d+1$ points, about 30% of them uniformly sampled and the rest LLM proposals. No correction is applied during warm start. After the GP is fit, acquisition uses the GP mean alone until $m_{\min}$ records have settled.

Hybrid candidate pool. After warm start, each round selects from a pool $\mathcal{C}_t$ of $5d$ LLM proposals and $5d + \lceil 0.1\,t^2 \rceil$ uniform samples. The uniform part is a coverage channel independent of the LLM proposer, and its growth keeps coverage as LLM proposals concentrate and extends the pool-level regret bound of Section 4 to the whole domain (Corollary 13). The calibration coefficient affects only the mean correction, not pool construction.

4 Theory

The analysis uses three invariants of Section 3: records are fixed before their outcomes, $|q_{t-1}(\mathbf{x})| \leq c_{t-1}$ for every candidate and advisor, and advice never modifies the GP width. It bounds the influence of advice. It does not establish that the correction improves optimization, which Section 5 evaluates. Assumptions, proofs and further results are in Appendix B.

What the calibration targets. Let (a, r) be a record's disagreement and GP residual, as in Eqs. (3) and (4), drawn from a fixed distribution over evaluated points, and let $\mathcal{R}(\theta) = \mathbb{E}[(r - \theta a)^2]$, so that $\mathcal{R}(0)$ is the squared error of the uncorrected GP mean.

Proposition 1 (Increment risk decomposition)**.** *If $\mathbb{E}[a^2] > 0$ then, with $\theta^\star = \mathbb{E}[ar]/\mathbb{E}[a^2]$,*

$$\mathcal{R}(\theta) - \mathcal{R}(0) \;=\; \underbrace{\mathbb{E}[a^2]\,\big(\theta - \theta^\star\big)^2}_{\textit{estimation excess}} \;-\; \underbrace{\mathbb{E}[ar]^2/\mathbb{E}[a^2]}_{\textit{achievable reduction}},$$

so $\mathcal{R}(\theta) < \mathcal{R}(0)$ if and only if $|\theta - \theta^\star| < |\theta^\star|$, and $\theta^\star < 0$ if and only if $\mathbb{E}[ar] < 0$.

For this linear correction, the achievable risk reduction is determined by the advice-residual cross-moment rather than by standalone predictive accuracy. Advice identical to the GP has $a \equiv 0$ and zero value however accurate it is, which is why CaliBO scores the disagreement rather than the LLM prediction, and a negative cross-moment is why its coefficient is signed.

CaliBO estimates this coefficient online from records collected where BO chooses to evaluate. Proposition 6 (Appendix B) bounds the error of $\hat{\theta}$ relative to any fixed reference coefficient by a shrinkage bias, a self-normalized statistical term (Abbasi-Yadkori et al., 2011), and a relationship-mismatch term that more data does not remove. Because records come from selected points, it does not show that $\hat{\theta}$ converges to $\theta^\star$.

What an arbitrary advisor can cost the search.

Lemma 2 (Bounded acquisition perturbation)**.** *Let $\tilde{\sigma}_{t-1}(\mathbf{x})$ be any acquisition width shared by the corrected and uncorrected rules, and let $A_t(\mathbf{x}) = \mu_{\mathrm{GP},t-1}(\mathbf{x}) - \sqrt{\beta_t}\,\tilde{\sigma}_{t-1}(\mathbf{x})$ be the uncorrected acquisition. For every round t and every $\mathbf{x} \in \mathcal{C}_t$, $|q_{t-1}(\mathbf{x})| \le c_{t-1}$, and the point selected by the corrected acquisition satisfies $A_t(\mathbf{x}_t) \le \min_{\mathbf{x}\in\mathcal{C}_t} A_t(\mathbf{x}) + 2c_{t-1}$.*

The lemma requires no assumption on the advisor. Let $\mathcal{I}_{\mathrm{LCB}} = \{n_0 + 1, \dots, T\}$ be the acquisition rounds after the warm start, γ_T the maximal information gain of the kernel, and $R_T^{\mathrm{pool}} = \sum_{t\in\mathcal{I}_{\mathrm{LCB}}} \big(f(\mathbf{x}_t) - \min_{\mathbf{x}\in\mathcal{C}_t} f(\mathbf{x})\big)$ the cumulative pool regret.

Theorem 3 (Regret under bounded mean corrections)**.** *Consider the analysis variant obtained by replacing the deployed predictive width $\sigma_{\mathrm{GP},t-1}$ in Eq. (7) with the latent posterior standard deviation $\sigma^{\mathrm{lat}}_{t-1}$ of a fixed kernel with known noise variance. Under Assumptions 1 and 2 (Appendix B), on the confidence event $\mathcal{E}$, of probability at least $1-\delta$, and for any realized advice, with $\beta_T = \max_{t\in\mathcal{I}_{\mathrm{LCB}}} \beta_t$,*

$$R_T^{\mathrm{pool}} \;\le\; 2\sqrt{\beta_T} \sum_{t\in\mathcal{I}_{\mathrm{LCB}}} \sigma^{\mathrm{lat}}_{t-1}(\mathbf{x}_t) \;+\; 2 \sum_{t\in\mathcal{I}_{\mathrm{LCB}}} c_{t-1} \;\le\; O\big(\sqrt{T\,\beta_T\,\gamma_T}\big) \;+\; 2 \sum_{t\in\mathcal{I}_{\mathrm{LCB}}} c_{t-1}.$$

The first term is the standard GP-LCB bound, and $2\sum_t c_{t-1}$ is the entire advice-dependent addition, a logged quantity. It is sublinear under a zero-usefulness null, where advice carries no information beyond the history (Corollary 10), the uniform pool extends the bound to domain regret (Corollary 13), and Appendix B.7 states what transfers to the deployed width.

5 Experiments

Setup. We compare against ten baselines in two groups. The *LLM-augmented* group includes LLAMBO (Liu et al., 2024), LLINBO (Chang et al., 2025), LABO (Chen et al., 2026), and an adapted drifting-source multi-information-source variant (MISO) (Poloczek et al., 2017), all with gpt-4o-mini as the advisor. The *classical* non-LLM group includes GP-EI (Jones et al., 1998), GP-LCB (Srinivas et al., 2010), SMAC3 (Lindauer et al., 2022), Optuna-TPE (Akiba et al., 2019), TuRBO (Eriksson et al., 2019), and HEBO (Cowen-Rivers et al., 2022). The real-world HPO evaluation covers 25 Bayesmark (Turner et al., 2021) tasks and 24 HPOBench (Eggensperger et al., 2021) tasks. We additionally use 30 pre-specified hidden-structure synthetic tasks, consisting of 18 Gaussian-process sample paths and 12 rotated classic-function instances. We use budget $T = 25$ and five seeds, min-max normalize objectives per task (Appendix C.1), and test with paired Wilcoxon tests across tasks, Holm-corrected within each suite. Protocol details are in Appendices C–C.9. Figure 2 plots CaliBO against its GP-LCB base and the four LLM-augmented methods, which query the same advisor, so the curves compare different ways of using the same advice. All ten baselines are compared at the final budget in Table 2.

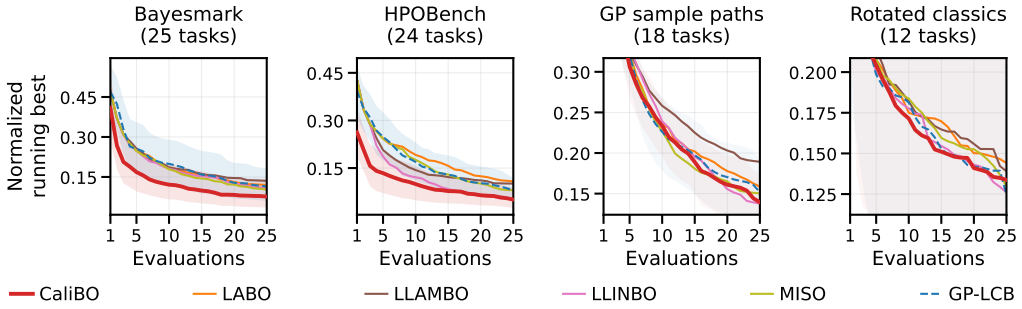

Figure 2: Anytime performance. Lower is better. Bands show 95% task-level bootstrap intervals for CaliBO and GP-LCB. Solid lines use an LLM advisor and dashed lines do not. The zero-prior panels are zoomed past the first evaluations.

Table 2: Paired comparison against CaliBO at $T = 25$ on 49 real-world tasks, five seeds. Negative favors CaliBO, p is Holm-adjusted within each suite, and bold is significant at 0.05.

	Bayesmark (25 tasks)			HPOBench (24 tasks)		
Comparison	mean diff.	tasks won	p	mean diff.	tasks won	p
LLM-augmented baselines						
LLAMBO	$\mathbf{-0.120}$	21/25	0.002	$\mathbf{-0.137}$	21/24	0.0001
LLINBO	-0.030	13/25	1.000	-0.024	13/24	0.446
LABO	$\mathbf{-0.116}$	16/25	0.041	$\mathbf{-0.205}$	24/24	$< 10^{-4}$
MISO	-0.056	17/25	0.106	$\mathbf{-0.122}$	17/24	0.014
Classical baselines						
GP-EI	$\mathbf{-0.327}$	21/25	0.0004	$\mathbf{-0.541}$	24/24	$< 10^{-4}$
GP-LCB	-0.076	15/25	0.173	$\mathbf{-0.145}$	19/24	0.004
SMAC3	$\mathbf{-0.115}$	18/25	0.041	$\mathbf{-0.214}$	24/24	$< 10^{-4}$
Optuna-TPE	-0.049	16/25	1.000	-0.039	11/24	0.690
TuRBO	-0.035	11/25	1.000	-0.023	14/24	0.690
HEBO	-0.058	14/25	0.222	-0.072	16/24	0.100

5.1 Overall Optimization Performance

CaliBO is favorable against every baseline at the final budget on real-world HPO. At $T = 25$, CaliBO has a negative mean difference against every baseline on both Bayesmark and HPOBench (Table 2). The gains are significant over LLAMBO, LABO, GP-EI, and SMAC3 on both suites, and over MISO and GP-LCB on HPOBench.

CaliBO remains robust when the LLM has no useful prior knowledge. On the hidden-structure synthetic suites, where task semantics are unavailable to the LLM advisor, CaliBO largely tracks its GP-LCB base (Figure 2), and no difference from it is significant on either family (Appendix D.3). Non-inferiority at the pre-specified 0.02 margin holds on the sample paths (Table 3). The comparisons with other LLM-augmented methods are generally favorable, while HEBO and TuRBO are better on the rotated classics, where GP-LCB itself is not competitive. This is consistent with the calibration mechanism reducing sensitivity to uninformative advice.

The real-world advantage emerges early. Figure 2 shows that the margin on Bayesmark and HPOBench opens within the first few evaluations and persists across the budget. The separation is therefore concentrated in the low-budget regime, where task-specific observations are still scarce.

5.2 Attributing the Gain

We attribute the gain with two controls run in the same pipeline, then ask whether a stronger advisor and a much higher dimension change the picture.

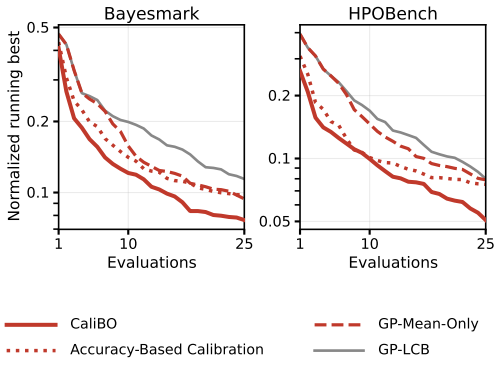

Figure 3: CaliBO and the two attribution controls under gpt-4o-mini, five seeds: normalized running best, log axis, lower is better.

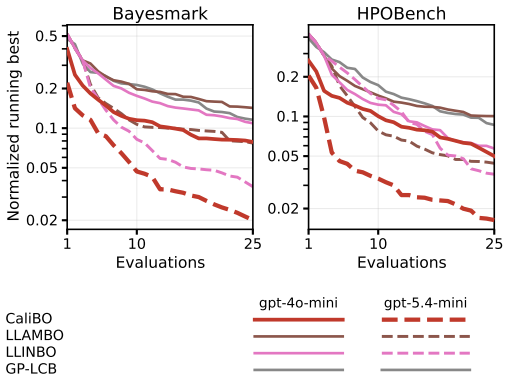

Figure 4: CaliBO, LLAMBO and LLINBO under both backbones, three seeds each: normalized running best, log axis, lower is better.

Does the advice add value beyond the candidate pool? The ***GP-Mean-Only*** control keeps CaliBO's LLM-uniform candidate pool, GP and acquisition rule and acquires with $\mu_{\mathrm{GP},t-1}$ alone, so the LLM shapes where it searches but never enters its model. It matches the protocol rather than the sampled points, since selections diverge once the two rules differ. CaliBO is better on both real-world suites, significantly on HPOBench, and no difference is significant on the zero-prior suites (Figure 3, Table 3 and Appendix D.3). The advice therefore adds value beyond the LLM candidate pool.

Should trust come from the LLM's accuracy or from its disagreement with the GP? ***Accuracy-Based Calibration*** fits the coefficient to the advisor's standalone agreement with outcomes, $y_s \approx \theta\, \mu_{\mathrm{LLM},s-1}(\mathbf{x}_s)$, instead of CaliBO's question of whether its disagreement with the GP predicts the GP's error, $r_s \approx \theta a_s$. Everything else is unchanged, including the correction applied to the disagreement Δ, the shrinkage and the budget rule. The swap pushes the coefficient above 0.8 in 90% of runs, multiplies the applied correction about fivefold, and degrades performance significantly on both suites (Figure 3). Because the LLM prediction and the outcome both contain the GP mean, their agreement credits the advisor for what the GP already knows (Remark 5). Which quantity is calibrated, not only whether advice is used, decides whether the correction helps or harms.

Does a stronger advisor widen the margin? CaliBO and the GP-Mean-Only control were rerun with gpt-5.4-mini on all 49 tasks at three seeds, every constant unchanged. The stronger backbone improves both arms, and on the same three seeds CaliBO's lead over the control grows on both real-world suites, from 0.016 to 0.122 on Bayesmark and from 0.065 to 0.092 on HPOBench, while on the zero-prior sample paths it trails the control slightly. This is descriptive. LLAMBO and LLINBO, the two LLM-augmented baselines rerun with gpt-5.4-mini, both improve with the stronger backbone as well, and CaliBO stays ahead of them under both backbones (Figure 4).

Does the gain persist at 180 dimensions? On the `dna` instance of LassoBench (Šehić et al., 2022) at $T = 100$, with the capped-pool variant of Appendix D.1, CaliBO beats all six classical baselines on every seed (Figure 5). The GP-Mean-Only control ends close to CaliBO (Appendix D.1), so at this scale the candidate pool carries the gain and the rest of the advice adds nothing detectable.

5.3 Advisor Quality and Failure Boundaries

A real advisor's quality is fixed by the model and the task, so it cannot be varied on purpose. To test how each way of assigning trust responds to advice of known quality, we replace the LLM with synthetic advisors whose prediction is a fixed function of the true objective f, on the pre-specified zero-prior tasks with all constants fixed beforehand (Appendix C.3).

Advisors. Let $\bar{f}_t$ be the mean of f over the round-t candidate pool and τ_t the range of the observed values. Figure 6 uses five advisors, and Figure 7 reports thirteen.

- **Oracle**: returns f itself, the best advice possible.

Method	Best loss	Seeds
CaliBO	$\mathbf{0.313 \pm 0.008}$	–
Optuna-TPE	0.324 ± 0.008	5/5
SMAC3	0.333 ± 0.004	5/5
HEBO	0.337 ± 0.007	5/5
GP-LCB	0.339 ± 0.002	5/5
TuRBO	0.342 ± 0.004	5/5
GP-EI	0.362 ± 0.016	5/5

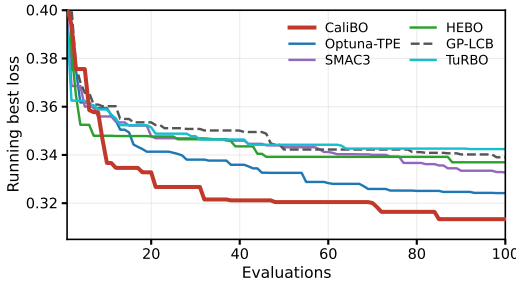

Figure 5: LassoBench `dna`, 180 dimensions, $T = 100$, five seeds. Left: final best cross-validation loss (mean $\pm$ standard deviation) and seeds on which CaliBO is better. Right: running best, mean over seeds.

- **Noisy**: returns $f + \tau_t z$ with independent $z \sim \mathcal{N}(0, 1)$ per candidate, so the noise is as large as the whole observed range.
- **Echo**: returns f at the quarter of candidates closest to evaluated points and the inverted value elsewhere, like an advisor that repeats its in-context examples but extrapolates wrongly.
- **Inverted**: returns $2\bar{f}_t - f$, the objective reflected about its pool mean. The spread is unchanged and the ranking is reversed: with $\bar{f}_t = 0.3$, a candidate with $f = 0.5$ is predicted at 0.1.
- **Pool mean**: returns $\bar{f}_t$ at every candidate, which says nothing about how candidates rank.

Mechanisms. Two single-variable controls change one choice of CaliBO each, and three references stand for the trust families of Table 1.

- **Accuracy-Based Calibration** fits the coefficient to the advisor's agreement with outcomes, as in Section 5.2. It changes only the evidence.
- **Nonnegative CaliBO** truncates $\hat{\theta}$ at zero. It changes only the sign.
- **Direct surrogate** follows LLAMBO (Liu et al., 2024): the advisor's mean replaces the GP mean, with the GP width kept, which trusts the advisor fully.
- **Gated decaying mixture** follows the decaying prior influence of πBO (Hvarfner et al., 2022): with $\mu_{\mathrm{adv},t-1}$ the advisor's prediction, $(1 - \eta_t)\,\mu_{\mathrm{GP},t-1} + \eta_t\,\mu_{\mathrm{adv},t-1}$ with $\eta_t = t^{-1/2}(1 - w_t)$, where the gate $w_t \in [0, 1]$ grows when the GP out-predicts the advisor (Appendix C.3). The weight is never negative.
- **LABO-style calibration** follows the scoring of LABO (Chen et al., 2026): a least-squares transfer coefficient from advisor predictions and outcomes at evaluated points, plus a GP on the remaining discrepancy, without LABO's fidelity gate and predictive spread.

Signed trust is what exploits wrong advice. Under inverted and echo advice CaliBO's coefficient turns negative (median -0.13 under inverted and -0.07 under echo advice) and it improves on GP-LCB. Nonnegative CaliBO, identical except for the sign, loses both gains, and the two references whose weight cannot be negative, the direct surrogate and the gated mixture, are harmed by inversion. LABO-style calibration, which can also reverse its coefficient, improves on one family and is harmed on the other.

The calibration target matters when the advice is wrong. With oracle advice Accuracy-Based Calibration gains as much as CaliBO or more, and on the sample paths it tracks CaliBO throughout. On the rotated classics it is significantly harmed by inverted and pool-mean advice, where CaliBO is not, consistent with its harm on real-world HPO in Section 5.2.

The failure regime is a confident but uninformative advisor. Across the thirteen advisors CaliBO improves on GP-LCB on average under 12 on the sample paths and 10 on the rotated classics (Figure 7). Its only significant harm is the pool-mean advisor. It carries no ranking information, yet its disagreement with the GP, $\bar{f}_t - \mu_{\mathrm{GP},t-1}$, is correlated with the GP's residual, so the coefficient becomes large (median 0.61) and the correction pulls the search toward average regions. Advice that explains the GP's error need not point to better candidates.

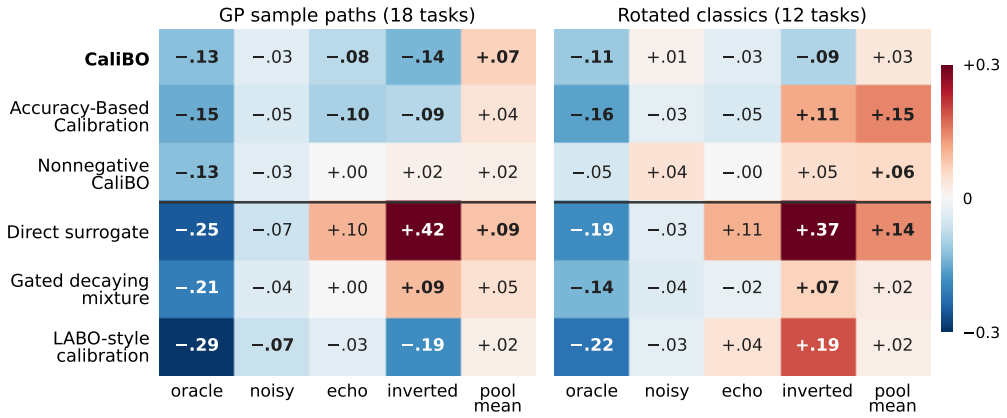

Figure 6: Final difference against plain GP-LCB under five synthetic advisors, negative favoring the mechanism, bold when $p < 0.05$. Above the line: CaliBO and its two single-variable controls. Below: reference trust families. All thirteen advisors are in Figure 7.

The price of influence control. With oracle advice CaliBO keeps three fifths to four fifths of the gated mixture's gain, the combined cost of estimating, shrinking and clipping.

6 Discussion and Limitations

When advice helped, and when it stopped mattering. Advice was most useful when its errors were structured and informative about the objective. When the advice was unstructured, we observed no measurable harm at the stated margins. When it was systematically inverted, the signed correction could still exploit it as signal. The only setting in which we observed measurable harm was an advisor with no ranking information whose disagreement with the GP nevertheless remained correlated with the GP's residual error.

Limitations. Our empirical findings are based on two LLM backbones, with the statistically significant main results established using gpt-4o-mini, and should therefore not be interpreted as claims about LLM advisors in general. The mechanism is designed primarily for the low-budget regime, where task-specific observations remain scarce. The regret analysis bounds the additional influence of advice rather than guaranteeing that every advisor improves optimization. Finally, the synthetic-advisor study exposes an important failure mode: an uninformative advisor can still receive substantial influence when its disagreement with the GP predicts the GP's residual error, because the current scalar calibration criterion measures predictive correction rather than ranking utility.

7 Conclusion

CaliBO calibrates LLM advice by asking whether the advisor's disagreement with the GP predicts the GP's remaining error, using records committed before the corresponding outcomes are observed. On real-world hyperparameter optimization tasks, it is competitive with a broad set of baselines, and the experiments indicate that calibrated advice can provide value beyond the LLM candidate pool. Calibrating trust from the LLM's standalone predictive accuracy rather than from its incremental disagreement with the GP can instead make the same correction harmful. Controlled synthetic-advisor experiments further show that signed incremental trust can exploit systematically inverted advice, while also revealing a failure mode in which an uninformative advisor receives influence because its disagreement still tracks the GP's error.

Reproducibility Statement

All constants of the mechanism are fixed a priori and listed in Appendix C.8, and the complete algorithm is given in Appendix A. Appendix C describes the evaluation protocol, seed sets, normalization,

and statistical testing conventions. The pre-specified task suite is documented in Appendix C.2, and the exact prompt templates are provided in Appendix C.7. Appendix C.9 details baseline reproduction, including the two adaptations made to the LABO construction and the shared candidate-pool specification. All assumptions and complete proofs are provided in Appendix B, together with the scope and limitations of the theoretical results. Failure handling and response-compliance rates relevant to reproducing the experiments with a different advisor are reported in Appendix C.6, while per-run LLM call counts, token usage, and costs are reported in Appendix C.5. The synthetic-advisor suite is fully specified in Appendix C.3. Source code and frozen result files are included in the anonymous supplementary material.

AI USE STATEMENT

We used generative AI tools in two distinct roles. First, an LLM is the *object of study*: the advisor whose predictions are calibrated by the proposed mechanism is an OpenAI model queried through its API. The corresponding prompts, sampling settings, token limits, and response-compliance rates are reported in Appendices C.7 and C.6. Second, as an *authoring and research aid*, we used an LLM assistant for paper writing, code implementation, drafting and checking mathematical arguments, and assistance in interpreting experimental results. All AI-assisted code, mathematical arguments, experimental analyses, and text were reviewed and verified by the authors before being included in the paper. We did not use generative AI to generate the research ideas underlying this work or to generate experimental data. The authors take full responsibility for the final manuscript, including its text, claims, code, and experimental artifacts.

REFERENCES

Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2011.

Masaki Adachi, Brady Planden, David Howey, Michael A. Osborne, Sebastian Orbell, Natalia Ares, Krikamol Muandet, and Siu Lun Chau. Looping in the human: Collaborative and explainable Bayesian optimization. In *International Conference on Artificial Intelligence and Statistics*, 2024.

Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 2623–2631, 2019.

J. M. Bates and C. W. J. Granger. The combination of forecasts. *Operational Research Quarterly*, 20 (4):451–468, 1969. doi: 10.1057/jors.1969.103.

Eric Brochu, Vlad M. Cora, and Nando de Freitas. A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *arXiv preprint arXiv:1012.2599*, 2010.

Richard H. Byrd, Peihuang Lu, Jorge Nocedal, and Ciyou Zhu. A limited memory algorithm for bound constrained optimization. *SIAM Journal on Scientific Computing*, 16(5):1190–1208, 1995. doi: 10.1137/0916069.

Nicolò Cesa-Bianchi and Gábor Lugosi. *Prediction, Learning, and Games*. Cambridge University Press, 2006. doi: 10.1017/CBO9780511546921.

Chih-Yu Chang, Milad Azvar, Chinedum Okwudire, and Raed Al Kontar. Llinbo: Trustworthy llm-in-the-loop bayesian optimization. *arXiv preprint arXiv:2505.14756*, 2025.

Jiangyu Chen and Ban Yi. Evidence-gated LLM priors for multi-objective Bayesian optimization. *arXiv preprint arXiv:2606.01730*, 2026.

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. In *International Conference on Machine Learning*, 2026. arXiv:2605.22054.

Alexander I. Cowen-Rivers, Wenlong Lyu, Rasul Tutunov, Zhi Wang, Antoine Grosnit, Ryan-Rhys Griffiths, Alexandre Max Maraval, Hao Jianye, Jun Wang, Jan Peters, and Haitham Bou-Ammar. HEBO: Pushing the limits of sample-efficient hyper-parameter optimisation. *Journal of Artificial Intelligence Research*, 74:1269–1349, 2022.

Katharina Eggensperger, Philipp Müller, Neeratyoy Mallik, Matthias Feurer, Rene Sass, Aaron Klein, Noor Awad, Marius Lindauer, and Frank Hutter. HPOBench: A collection of reproducible multi-fidelity benchmark problems for HPO. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021. URL https://openreview.net/forum?id=1k4rJYEwda-.

David Eriksson, Michael Pearce, Jacob Gardner, Ryan D Turner, and Matthias Poloczek. Scalable global optimization via local bayesian optimization. *Advances in neural information processing systems*, 32, 2019.

Stefan Falkner, Aaron Klein, and Frank Hutter. BOHB: Robust and efficient hyperparameter optimization at scale. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, pp. 1437–1446, 2018.

Qiaozhu Feng, Zhibo Huang, Yang Wei, and Wei W Xing. BOPRO: Towards new style Bayesian optimization with large language models. In *Advanced Intelligent Computing Technology and Applications (ICIC 2025)*, Lecture Notes in Computer Science. Springer, 2025.

Matthias Feurer, Benjamin Letham, Frank Hutter, and Eytan Bakshy. Practical transfer learning for Bayesian optimization. *arXiv preprint arXiv:1802.02219*, 2018.

Amur Ghose, Andrew B Kahng, Sayak Kundu, and Zhiang Wang. Orfs-agent: Tool-using agents for chip design optimization. In *2025 ACM/IEEE 7th Symposium on Machine Learning for CAD (MLCAD)*, pp. 1–13. IEEE, 2025.

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.

Carl Hvarfner, Frank Hutter, and Luigi Nardi. A general framework for user-guided Bayesian optimization. In *International Conference on Learning Representations*, 2024.

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.

Marc C. Kennedy and Anthony O'Hagan. Bayesian calibration of computer models. *Journal of the Royal Statistical Society: Series B*, 63(3):425–464, 2001.

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 *Proceedings of the 41st International Conference on Machine Learning*, pp. 25603–25622. PMLR, 2024.

Marius Lindauer, Katharina Eggensperger, Matthias Feurer, André Biedenkapp, Difan Deng, Carolin Benjamins, Tim Ruhkopf, René Sass, and Frank Hutter. SMAC3: A versatile Bayesian optimization package for hyperparameter optimization. *Journal of Machine Learning Research*, 23(54):1–9, 2022.

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

Thodoris Lykouris and Sergei Vassilvitskii. Competitive caching with machine learned advice. In *International Conference on Machine Learning*, 2018.

Kanan Mahammadli and Seyda Ertekin. Sequential large language model-based hyper-parameter optimization. *arXiv preprint arXiv:2410.20302*, 2024.

M. D. McKay, R. J. Beckman, and W. J. Conover. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 21(2): 239–245, 1979. doi: 10.1080/00401706.1979.10489755.

Petrus Mikkola, Julien Martinelli, Louis Filstroff, and Samuel Kaski. Multi-fidelity bayesian optimization with unreliable information sources. In *International Conference on Artificial Intelligence and Statistics*, 2023.

Jonas Mockus, Vytautas Tiesis, and Antanas Zilinskas. The application of Bayesian methods for seeking the extremum. *Towards Global Optimization*, 2(117-129):2, 1978.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Matthias Poloczek, Jialei Wang, and Peter I. Frazier. Multi-information source optimization. In *Advances in Neural Information Processing Systems*, 2017.

Manish Purohit, Zoya Svitkina, and Ravi Kumar. Improving online algorithms via ML predictions. In *Advances in Neural Information Processing Systems*, 2018.

Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning*. Adaptive Computation and Machine Learning. MIT Press, Cambridge, MA, 2006.

Kenan Šehić, Alexandre Gramfort, Joseph Salmon, and Luigi Nardi. LassoBench: A high-dimensional hyperparameter optimization benchmark suite for Lasso. In *International Conference on Automated Machine Learning (AutoML)*, 2022.

Bobak Shahriari, Kevin Swersky, Ziyu Wang, Ryan P Adams, and Nando De Freitas. Taking the human out of the loop: A review of bayesian optimization. *Proceedings of the IEEE*, 104(1): 148–175, 2016.

Alistair Shilton, Sunil Gupta, Santu Rana, and Svetha Venkatesh. Regret bounds for transfer learning in bayesian optimisation. In *International Conference on Artificial Intelligence and Statistics*, 2017.

Jasper Snoek, Hugo Larochelle, and Ryan P. Adams. Practical Bayesian optimization of machine learning algorithms. In *Advances in Neural Information Processing Systems*, volume 25, pp. 2951–2959, 2012.

Niranjan Srinivas, Andreas Krause, Sham M Kakade, and Matthias Seeger. Gaussian process optimization in the bandit setting: No regret and experimental design. In *Proceedings of the 27th International Conference on Machine Learning (ICML)*, pp. 1015–1022, 2010.

Ryan Turner, David Eriksson, Michael McCourt, Juha Kiili, Eero Laaksonen, Zhen Xu, and Isabelle Guyon. Bayesian optimization is superior to random search for machine learning hyperparameter tuning: Analysis of the black-box optimization challenge 2020. In *Proceedings of the NeurIPS 2020 Competition and Demonstration Track*, volume 133 of *Proceedings of Machine Learning Research*, pp. 3–26. PMLR, 2021. URL `https://proceedings.mlr.press/v133/turner21a.html`.

Sattar Vakili, Kia Khezeli, and Victor Picheny. On information gain and regret bounds in Gaussian process bandits. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2021.

Martin Wistuba, Nicolas Schilling, and Lars Schmidt-Thieme. Scalable Gaussian process-based transfer surrogates for hyperparameter optimization. *Machine Learning*, 107:43–78, 2018.

Zhuo Yang, Daolang Wang, Lingli Ge, Beilun Wang, Tianfan Fu, and Yuqiang Li. Reasoning bo: Enhancing bayesian optimization with long-context reasoning power of llms. *arXiv preprint arXiv:2505.12833*, 2025.

Yuxuan Yin, Yu Wang, Boxun Xu, and Peng Li. Ado-llm: Analog design bayesian optimization with in-context learning of large language models. In *Proceedings of the 43rd IEEE/ACM International Conference on Computer-Aided Design*, pp. 1–9, 2024.

Michael R Zhang, Nishkrit Desai, Juhan Bae, Jonathan Lorraine, and Jimmy Ba. Using large language models for hyperparameter optimization. *arXiv preprint arXiv:2312.04528*, 2023.

A Algorithm Details

Algorithm 1 CaliBO

Hybrid warm start, fit GP on $\mathcal{D}$, record set $\mathcal{B} \leftarrow \emptyset$
for $t = n_0 + 1, \ldots, T$ **do**
  If a pending record exists, settle it: $r \leftarrow y_{t-1} - b$, append (a, r) to $\mathcal{B}$
  Build hybrid candidate pool $\mathcal{C}_t$, query GP and LLM means on $\mathcal{C}_t$
  $\hat{\theta} \leftarrow \left(\sum a_s r_s / \sum a_s^2\right) \cdot m/(m+\kappa)$ if $m \geq m_{\min}$ and $\sum a_s^2 > \varepsilon_a$, else 0
  $c \leftarrow \lambda_c \sqrt{\max(V^{\text{res}} - \hat{\sigma}_\varepsilon^2, 0)}$ from settled residuals
  $\mu_{\text{fused}} \leftarrow \mu_{\text{GP}} + \text{clip}(\hat{\theta}\,\Delta, \pm c)$
  Select $\mathbf{x}_t$ by Eq. (7), record (a, b) at $\mathbf{x}_t$ before evaluating
  Evaluate $y_t = f(\mathbf{x}_t) + \varepsilon_t$, refit GP
end for

B Theoretical Analysis

Throughout this appendix, μ_{t-1} abbreviates the GP posterior mean $\mu_{\text{GP},t-1}$ after $t-1$ observations and σ_{t-1} abbreviates the latent posterior standard deviation $\sigma_{t-1}^{\text{lat}}$ of Theorem 3, the standard deviation of $f(\mathbf{x})$ under the posterior of Assumption 1. It differs from the deployed predictive width $\sigma_{\text{GP},t-1}$ of Eq. (7), which includes the fitted noise level (Appendix B.7). Further, $\mathcal{I}_{\text{LCB}} = \{n_0 + 1, \ldots, T\}$ denotes the acquisition rounds and $\beta_T = \max_{t \in \mathcal{I}_{\text{LCB}}} \beta_t$, $\mathcal{F}_{t-1}$ denotes the σ-algebra generated by the first $t-1$ queries, observations, and LLM responses, and the quantities a_s, r_s, $\hat{\theta}_{t-1}$, c_{t-1} are as defined in Section 3. The settled records available at round t are relabeled $1, \ldots, m$ in order of settlement, with $m = m_t$, and $\hat{\theta}_m$ denotes the coefficient $\hat{\theta}_{t-1}$ computed from them. A quantity such as $\sigma_{s-1}(\mathbf{x}_s)$ attached to record s refers to the round at which that record was made. Before activation ($m < m_{\min}$) the fused mean equals the GP mean and every bound below holds trivially with $c_{t-1} = 0$.

B.1 Assumptions

Assumption 1 (GP model)**.** *The kernel k and the noise variance σ_ε^2 are fixed and known, posterior updates are exact under them, and conditionally on $\mathcal{F}_{t-1}$ the objective value at any $\mathbf{x}$ satisfies $f(\mathbf{x}) \sim \mathcal{N}\big(\mu_{t-1}(\mathbf{x}), \sigma_{t-1}^2(\mathbf{x})\big)$, and observations are $y_t = f(\mathbf{x}_t) + \varepsilon_t$ with $\varepsilon_t \sim \mathcal{N}(0, \sigma_\varepsilon^2)$ independent of everything else.*

Assumption 2 (Confidence event)**.** *With β_t as in Eq. (7), the event $\mathcal{E} = \big\{|f(\mathbf{x}) - \mu_{t-1}(\mathbf{x})| \leq \sqrt{\beta_t}\,\sigma_{t-1}(\mathbf{x}) \;\; \forall t \leq T, \; \forall \mathbf{x} \in \mathcal{C}_t\big\}$ holds with probability at least $1 - \delta$. For an uninformative advisor it follows from the union bound over the finite pools encoded in β_t (Srinivas et al., 2010). For informative advice it is assumed.*

Assumption 3 (Zero-usefulness null)**.** *Conditionally on $\mathcal{F}_{t-1}$ and on the advice function $\Delta_{t-1}(\cdot)$, the distribution of y_t is the GP posterior predictive of Assumption 1. Equivalently, the advice carries no information about f beyond $\mathcal{D}_{t-1}$.*

Assumption 4 (Lipschitz objective)**.** *The domain is the unit box $\mathcal{X} = [0,1]^d$ after coordinate normalization, and f is L-Lipschitz on $\mathcal{X}$ in the Euclidean norm.*

Assumption 5 (Noise estimate)**.** *At every round t, the fitted noise level satisfies $\hat{\sigma}_{\varepsilon,t-1}^2 \geq \sigma_\varepsilon^2 - \zeta_t$ for a deterministic slack sequence $\zeta_t \geq 0$ carried through the bounds. The fitted level is the deflator inside c_{t-1} in Eq. (6), distinct from the known variance used by the analyzed posterior.*

The next three items support the result on the coefficient estimate and do not presume a correctly specified GP.

Assumption 6 (Pre-evaluation filtration)**.** *Let $\mathcal{G}_s$ be the σ-algebra generated by everything the run has produced at the instant the round-s evaluation is issued: the advice functions $\Delta_0, \ldots, \Delta_{s-1}$, the candidate pools $\mathcal{C}_1, \ldots, \mathcal{C}_s$, the selected points $\mathbf{x}_1, \ldots, \mathbf{x}_s$, all internal randomization, and the observations $y_1, \ldots, y_{s-1}$. The family is increasing by construction, $\mathcal{G}_s \subseteq \mathcal{G}_{s+1}$, because $\mathcal{G}_{s+1}$ adds*

y_s and the round-$(s+1)$ quantities to it. Under this definition $a_s = \Delta_{s-1}(\mathbf{x}_s)$, $b_s = \mu_{s-1}(\mathbf{x}_s)$, every settled record $(a_{s'}, r_{s'})$ with $s' < s$, and therefore $\hat{\theta}_{s-1}$ and c_{s-1}, are all $\mathcal{G}_s$-measurable, while y_s is realized only after $\mathcal{G}_s$ is determined.

This holds by construction, since each record is written before its evaluation is issued. The selected point may depend on earlier records.

Assumption 7 (Conditionally sub-Gaussian settlement)**.** *There is $R_\xi > 0$ such that $\xi_s := r_s - \mathbb{E}[r_s \mid \mathcal{G}_s]$ satisfies $\mathbb{E}\big[e^{\lambda \xi_s} \mid \mathcal{G}_s\big] \le e^{\lambda^2 R_\xi^2/2}$ for all $\lambda \in \mathbb{R}$ and all s.*

It holds whenever f is bounded and the observation noise is sub-Gaussian, which covers every suite in this paper, and it makes no demand on the advisor.

Definition 4 (Reference coefficient and mismatch)**.** *Fix $\theta^{\mathrm{ref}} \in \mathbb{R}$, not depending on the realized run, and set*

$$\phi_s \;:=\; \mathbb{E}[r_s \mid \mathcal{G}_s] - \theta^{\mathrm{ref}} a_s, \qquad \xi_s \;:=\; r_s - \mathbb{E}[r_s \mid \mathcal{G}_s], \qquad \textit{so} \qquad r_s \;=\; \theta^{\mathrm{ref}} a_s + \phi_s + \xi_s$$

with ϕ_s $\mathcal{G}_s$-measurable and $\mathbb{E}[\xi_s \mid \mathcal{G}_s] = 0$.

The decomposition holds for every choice of θ^{ref} and posits no linear relationship. Its content is the size of the mismatch sequence $\{\phi_s\}$.

B.2 What the Calibration Targets

Proposition 1 is stated under a fixed distribution Q on the pair (a, r), which we take square integrable with $\alpha := \mathbb{E}_Q[a^2] \in (0, \infty)$. Write $\mathcal{R}(\theta) := \mathbb{E}_Q[(r - \theta a)^2]$ and $\theta^\star := \mathbb{E}_Q[ar]/\alpha$. Two consequences of the identity are used below: $\mathcal{R}$ is minimized uniquely at $\theta^\star$, with $\mathcal{R}(\theta^\star) = \mathcal{R}(0) - \mathbb{E}_Q[ar]^2/\alpha$, and the non-strict form $\mathcal{R}(\theta) \le \mathcal{R}(0)$ holds if and only if $|\theta - \theta^\star| \le |\theta^\star|$.

Proof of Proposition 1. Expanding, $\mathcal{R}(\theta) = \mathbb{E}_Q[r^2] - 2\theta\,\mathbb{E}_Q[ar] + \theta^2\alpha$, so $\mathcal{R}(\theta) - \mathcal{R}(0) = \theta^2\alpha - 2\theta\alpha\theta^\star = \alpha\big((\theta - \theta^\star)^2 - (\theta^\star)^2\big)$, which is the displayed identity since $\alpha(\theta^\star)^2 = \mathbb{E}_Q[ar]^2/\alpha$. The right side is negative exactly when $(\theta - \theta^\star)^2 < (\theta^\star)^2$, which is the strict equivalence, and the sign statement is the sign of $\mathbb{E}_Q[ar]/\alpha$ with $\alpha > 0$. The two consequences above follow from the identity since $\alpha > 0$. □

Remark 5 (Why the LLM prediction is the wrong regressand)**.** *Let $\ell := a + b$ be the LLM prediction and $Y := r + b$ the settled outcome, where b is the surrogate mean at the same point, so that Accuracy-Based Calibration regresses Y on ℓ from the identical pre-evaluation records. Both variables then contain b, and the shared term contributes to the accuracy fit whatever the advisor does. A predictor that returns $\mu_{\mathrm{LLM}} = \mu_{\mathrm{GP}}$ exactly therefore has $a \equiv 0$, so the disagreement fit has nothing to win, while the accuracy fit regresses $Y = b + r$ on $\ell = b$ and scores well whenever the spread of b across the scored points is large next to the surrogate's own error. That fit is not perfect, since r stays unexplained, but what it scores is bought entirely with the shared baseline signal. The achievable reduction of Proposition 1 is stated for the increment pair because that is the pair whose gain is zero when the advisor adds nothing. The same principle is why the regression of Eq.* (5) *has no intercept: an intercept would absorb the surrogate's own systematic bias, which is not information the advisor supplied. The single-variable experiment separating the two is Figure 3, with the tests in Appendix D.3.*

B.3 What the Online Coefficient Estimates

Proposition 6 (Online coefficient error)**.** *Adopt Assumptions 6 and 7 and Definition 4. Let $S_m := \sum_{s \le m} a_s^2$, $h_m := m/(m + \kappa)$, and let $\hat{\theta}_m := h_m \sum_{s \le m} a_s r_s / S_m$ on $\{S_m > 0\}$ and $\hat{\theta}_m := 0$ otherwise, as in Eq.* (5)*. The mechanism consults this estimate only once $m \ge m_{\min}$ and holds $\hat{\theta} = 0$ before then. Items (i) and (iii) below describe the activated regime, which is the only one in which the coefficient is used. Item (ii) holds for every prefix, activated or not. Then*

(i) on $\{m \ge m_{\min},\, S_m > 0\}$, exactly,

$$\hat{\theta}_m - \theta^{\mathrm{ref}} \;=\; -(1 - h_m)\,\theta^{\mathrm{ref}} \;+\; h_m \,\frac{\sum_{s \le m} a_s \xi_s}{S_m} \;+\; h_m \,\frac{\sum_{s \le m} a_s \phi_s}{S_m},$$

(ii) for any fixed $\lambda_0 > 0$ and any $\delta \in (0,1)$, with probability at least $1-\delta$, simultaneously for every $m \geq 1$,

$$\Big|\sum_{s\leq m} a_s\xi_s\Big| \;\leq\; B_m(\delta) \;:=\; R_\xi\sqrt{2\,(S_m+\lambda_0)\,\log\frac{\sqrt{1+S_m/\lambda_0}}{\delta}}\,,$$

(iii) on that event, simultaneously for every $m \geq m_{\min}$ with $S_m > 0$,

$$\big|\hat\theta_m - \theta^{\mathrm{ref}}\big| \;\leq\; \frac{\kappa}{m+\kappa}\big|\theta^{\mathrm{ref}}\big| \;+\; h_m\frac{B_m(\delta)}{S_m} \;+\; h_m\frac{\sum_{s\leq m}|a_s\phi_s|}{S_m},$$

and the middle term is at most $R_\xi\sqrt{2\big(1+\lambda_0/S_m\big)\log\big(\sqrt{1+S_m/\lambda_0}/\delta\big)/S_m}$, hence of order $R_\xi\sqrt{\log(S_m/\delta)/S_m}$ once $S_m \geq \lambda_0$.

Proof. (i) For $m \geq m_{\min}$ the estimate is the displayed ratio. By Definition 4, $\sum_{s\leq m} a_s r_s = \theta^{\mathrm{ref}} S_m + \sum_{s\leq m} a_s\phi_s + \sum_{s\leq m} a_s\xi_s$. Dividing by S_m, multiplying by h_m and subtracting θ^{ref} gives the display, the first term being $(h_m-1)\theta^{\mathrm{ref}}$.

(ii) Set $\mathcal{F}^{\mathrm{rec}}_{s-1} := \mathcal{G}_s$, which is increasing by Assumption 6. With respect to it the scalar regressor a_s is $\mathcal{F}^{\mathrm{rec}}_{s-1}$-measurable by Assumption 6, the noise ξ_s is $\mathcal{F}^{\mathrm{rec}}_{s}$-measurable since $y_s \in \mathcal{G}_{s+1}$, and ξ_s is conditionally R_ξ-sub-Gaussian with conditional mean zero given $\mathcal{F}^{\mathrm{rec}}_{s-1}$ by Assumption 7, so $\{(a_s,\xi_s)\}$ is a predictable-regressor, martingale-noise sequence. Theorem 1 of Abbasi-Yadkori et al. (2011), specialized to dimension one with regularizer λ_0 so that $\bar V_m = \lambda_0 + S_m$, gives, uniformly over m with probability at least $1-\delta$,

$$\frac{\big(\sum_{s\leq m} a_s\xi_s\big)^2}{\lambda_0+S_m} \;\leq\; 2R_\xi^2\log\frac{(\lambda_0+S_m)^{1/2}\lambda_0^{-1/2}}{\delta},$$

which rearranges to the stated $B_m(\delta)$.

(iii) Apply the triangle inequality to (i), using $1-h_m = \kappa/(m+\kappa)$ and $|\sum a_s\phi_s| \leq \sum|a_s\phi_s|$. For the final display, $h_m \leq 1$ and $B_m(\delta)/S_m = R_\xi\sqrt{2(S_m+\lambda_0)\log(\cdot)}/S_m = R_\xi\sqrt{2(1+\lambda_0/S_m)\log(\cdot)/S_m}$. □

B.4 Bounded Acquisition Perturbation

Lemma 2 holds for every round t, every realization and every shared width $\tilde\sigma_{t-1}$, with no assumption on the advisor or on the GP. In the regret proofs below the uncorrected acquisition with the latent width, A_t at $\tilde\sigma_{t-1} = \sigma_{t-1}$, is written LCB_t.

Proof of Lemma 2. The bound $|q_{t-1}(\mathbf{x})| \leq c_{t-1}$ is immediate from the clip in Eq. (2), so $|\mu_{\mathrm{fused},t-1}(\mathbf{x}) - \mu_{t-1}(\mathbf{x})| \leq c_{t-1}$ for all $\mathbf{x} \in \mathcal{C}_t$. Write $A^{\mathrm{f}}_t(\mathbf{x}) = \mu_{\mathrm{fused},t-1}(\mathbf{x}) - \sqrt{\beta_t}\,\tilde\sigma_{t-1}(\mathbf{x})$ for the corrected acquisition. Then $|A^{\mathrm{f}}_t(\mathbf{x}) - A_t(\mathbf{x})| \leq c_{t-1}$ pointwise since the width is shared. For any $\mathbf{x} \in \mathcal{C}_t$, $A_t(\mathbf{x}_t) \leq A^{\mathrm{f}}_t(\mathbf{x}_t) + c_{t-1} \leq A^{\mathrm{f}}_t(\mathbf{x}) + c_{t-1} \leq A_t(\mathbf{x}) + 2c_{t-1}$, using that $\mathbf{x}_t$ minimizes A^{f}_t. □

B.5 Regret Under Bounded Mean Corrections

We first record two facts about the records under the null.

Lemma 7 (Shrinkage never inflates the residuals)**.** *For any set of records, V^{res}_{t-1} evaluated at $\hat\theta_{t-1}$ satisfies $V^{\mathrm{res}}_{t-1} \leq \frac{1}{m}\sum_s r_s^2$. When $\sum_s a_s^2 = 0$ the mechanism sets $\hat\theta_{t-1} = 0$ and the claim is an identity.*

Proof. Write $\hat\theta = h_m\,\theta_{\mathrm{LS}}$ with $h_m = m/(m+\kappa) \in (0,1]$ and $\theta_{\mathrm{LS}} = \sum a_s r_s / \sum a_s^2$. Expanding, $\frac{1}{m}\sum_s(r_s - \hat\theta a_s)^2 = \frac{1}{m}\sum_s r_s^2 - \frac{1}{m}\,\theta^2_{\mathrm{LS}}\big(2h_m - h_m^2\big)\sum_s a_s^2$, and $2h_m - h_m^2 \geq 0$. □

Lemma 8 (Deterministic variance budget)**.** *For every selection sequence, $\sum_{t\le T}\sigma_{t-1}^2(\mathbf{x}_t)\le C_k\,\gamma_T$ with $C_k=2\bar{k}/\log(1+\bar{k}/\sigma_\varepsilon^2)$, $\bar{k}=\sup_{\mathbf{x}}k(\mathbf{x},\mathbf{x})$. Consequently $W_m:=\sum_{s\le m}\big(\sigma_{s-1}^2(\mathbf{x}_s)+\sigma_\varepsilon^2\big)$ satisfies $W_m\le W_T\le H_T:=C_k\gamma_T+T\sigma_\varepsilon^2$, and H_T is deterministic: it depends on the kernel and the horizon, not on the realized trajectory.*

Proof. By concavity of the logarithm, $u\le\frac{U}{\log(1+U)}\log(1+u)$ for $0\le u\le U$. Apply this with $u=\sigma_{t-1}^2(\mathbf{x}_t)/\sigma_\varepsilon^2$ and $U=\bar{k}/\sigma_\varepsilon^2$, sum over t, and bound $\frac{1}{2}\sum_t\log\big(1+\sigma_\varepsilon^{-2}\sigma_{t-1}^2(\mathbf{x}_t)\big)$ by γ_T, the maximum information gain over T-point sets (Srinivas et al., 2010). □

Lemma 9 (Residual scale under the null)**.** *With $\mathcal{G}_s$ as in Assumption 6, $\mathbf{x}_s$ and $v_s:=\sigma_{s-1}^2(\mathbf{x}_s)+\sigma_\varepsilon^2$ are $\mathcal{G}_s$-measurable. Under Assumptions 1 and 3, conditionally on $\mathcal{G}_s$ the settled outcome satisfies $r_s=y_s-\mu_{s-1}(\mathbf{x}_s)\sim\mathcal{N}(0,v_s)$. Write $\bar{v}:=\bar{k}+\sigma_\varepsilon^2$, and let H_T be the deterministic variance budget of Lemma 8. Then for any $\delta'\in(0,1)$, with probability at least $1-\delta'$, simultaneously for every prefix of $m\le T$ settled records,*

$$\frac{1}{m}\sum_{s\le m}r_s^2\;\le\;\frac{1}{m}\sum_{s\le m}\sigma_{s-1}^2(\mathbf{x}_s)\;+\;\sigma_\varepsilon^2\;+\;\frac{D_T(\delta')}{m},$$

$$D_T(\delta'):=2\sqrt{2\,\bar{v}H_T\log(1/\delta')}+8\bar{v}\log(1/\delta').$$

Proof. Conditional normality is Assumption 1 combined with Assumption 3: the advice carries no information about f beyond $\mathcal{D}_{s-1}$, so conditioning on it, and on the point it helped select, leaves the posterior predictive law of y_s at $\mathbf{x}_s$ unchanged. Write $r_s=\sqrt{v_s}\,Z_s$ with $Z_s\mid\mathcal{G}_s\sim\mathcal{N}(0,1)$.

Step 1 (a scalar moment bound). For $0\le u\le 1/4$, $\log\mathbb{E}\,e^{u(Z^2-1)}=-u-\frac{1}{2}\log(1-2u)=\sum_{j\ge2}\frac{(2u)^j}{2j}\le\frac{1}{4}\sum_{j\ge2}(2u)^j=\frac{u^2}{1-2u}\le 2u^2$.

Step 2 (supermartingale). Fix $\lambda\in(0,\frac{1}{4\bar{v}}]$ and set $U_m=\sum_{s\le m}v_s(Z_s^2-1)$, $Q_m=\sum_{s\le m}v_s^2$, $M_m=\exp(\lambda U_m-2\lambda^2Q_m)$. Since v_s is $\mathcal{G}_s$-measurable and $\lambda v_s\le 1/4$, Step 1 gives $\mathbb{E}[M_m\mid\mathcal{G}_m]\le M_{m-1}$ with $M_0=1$, so M is a supermartingale and Ville's inequality yields $\mathbb{P}\big(\sup_{m\le T}M_m\ge 1/\delta'\big)\le\delta'$. On the complement, for every $m\le T$, $U_m\le 2\lambda Q_m+\lambda^{-1}\log(1/\delta')\le 2\lambda\bar{v}H_T+\lambda^{-1}\log(1/\delta')$, using $Q_m\le\bar{v}W_m\le\bar{v}H_T$ by Lemma 8.

Step 3 (choice of λ). Because H_T is deterministic, λ can be fixed before any data are seen: take $\lambda=\min\big\{\frac{1}{4\bar{v}},\sqrt{\log(1/\delta')/(2\bar{v}H_T)}\big\}$, so no data-dependent parameter choice enters the supermartingale argument. If the second term attains the minimum, the bound is $2\sqrt{2\bar{v}H_T\log(1/\delta')}$. Otherwise $H_T\le 8\bar{v}\log(1/\delta')$ and the bound is $H_T/2+4\bar{v}\log(1/\delta')\le 8\bar{v}\log(1/\delta')$. Both cases are covered by $D_T(\delta')$, and dividing by m gives the display. □

Theorem 3 is stated for the acquisition evaluated in the latent posterior width σ_{t-1} of the fixed kernel of Assumption 1, with γ_T the maximum information gain of that kernel. All sums below run over $\mathcal{I}_{\mathrm{LCB}}$, and the warm-start rounds enter only in Corollary 13.

Proof of Theorem 3. On $\mathcal{E}$, for the pool minimizer $\mathbf{x}_t^\star$, $f(\mathbf{x}_t^\star)\ge\mathrm{LCB}_t(\mathbf{x}_t^\star)\ge\mathrm{LCB}_t(\mathbf{x}_t)-2c_{t-1}\ge f(\mathbf{x}_t)-2\sqrt{\beta_t}\,\sigma_{t-1}(\mathbf{x}_t)-2c_{t-1}$ by Lemma 2 and two applications of the confidence bound, with $\beta_t\le\beta_T$. Summing over $\mathcal{I}_{\mathrm{LCB}}$, bounding $\sum_t\sigma_{t-1}(\mathbf{x}_t)\le\sqrt{T\sum_t\sigma_{t-1}^2(\mathbf{x}_t)}$ by Cauchy–Schwarz, and applying Lemma 8 gives the display. □

The size of the advice-dependent term $2\sum_t c_{t-1}$ is bounded under the zero-usefulness null.

Corollary 10 (Cost of uninformative advice)**.** *If in addition Assumptions 3 and 5 hold, then with probability at least $1-\delta-\delta'$,*

$$\sum_{t\in\mathcal{I}_{\mathrm{LCB}}}c_{t-1}\;=\;O\big(\sqrt{T\,\gamma_T}\big)\;+\;O\Big(\sum_{t\in\mathcal{I}_{\mathrm{LCB}}}\sqrt{\zeta_t}\Big)\;+\;\tilde{O}\big(T^{3/4}\big),$$

which is sublinear whenever $\gamma_T=o(T)$ and $\sum_t\sqrt{\zeta_t}=o(T)$.

The explicit form behind Corollary 10 is

$$\sum_{t\in\mathcal{I}_{\mathrm{LCB}}} c_{t-1} \;\le\; 4\sqrt{C_k\,\gamma_T\,T} \;+\; 2\sum_{t\in\mathcal{I}_{\mathrm{LCB}}} \sqrt{\zeta_t} \;+\; 4\sqrt{D_T(\delta')\,T},$$

with C_k and H_T from Lemma 8 and $D_T(\delta') = \tilde{O}(\sqrt{H_T})$ from Lemma 9. Beyond the sublinearity stated in the corollary, up to logarithmic factors $\sum_t c_{t-1}$ stays within the order of $\sqrt{T\gamma_T}$ if γ_T grows polynomially faster than $\sqrt{T}$ and $\sum_t \sqrt{\zeta_t} = o\big(\sqrt{T\gamma_T}\big)$. Neither growth condition is implied by Assumption 5 alone.

Proof of Corollary 10. Let $m = m_t$ be the number of settled records at round t, so that $c_{t-1} = 0$ while $m < m_{\min}$. Chaining Lemma 7, Lemma 9 (on its probability-$(1-\delta')$ event, simultaneous over prefixes), and Assumption 5 inside the cap definition, with $\lambda_c = 2$,

$$c_{t-1}^2 \;=\; \lambda_c^2\big(V_{t-1}^{\mathrm{res}} - \hat{\sigma}^2_{\varepsilon,t-1}\big)_+ \;\le\; 4\Big(\tfrac{1}{m}\textstyle\sum_{s\le m}\sigma^2_{s-1}(\mathbf{x}_s) + \zeta_t + D_T(\delta')/m\Big),$$

so by $\sqrt{x+y+z} \le \sqrt{x} + \sqrt{y} + \sqrt{z}$, $c_{t-1} \le 2\sqrt{\frac{1}{m}\sum_{s\le m}\sigma^2_{s-1}(\mathbf{x}_s)} + 2\sqrt{\zeta_t} + 2\sqrt{D_T(\delta')/m}$. The first term is at most $2\sqrt{C_k\gamma_T/m}$ by Lemma 8. Since the settled count satisfies $m_t \ge t - t_0$ for a constant warm-start and settlement lag t_0, $\sum_t m_t^{-1/2} \le \sum_{m=1}^{T} m^{-1/2} \le 2\sqrt{T}$. Collecting terms, $\sum_t c_{t-1} \le 4\sqrt{C_k\gamma_T T} + 2\sum_t\sqrt{\zeta_t} + 4\sqrt{D_T(\delta')\,T}$. Finally $H_T \le C_k\gamma_T + T\sigma_\varepsilon^2$ inside $D_T(\delta')$, so $\sqrt{D_T\,T} = \tilde{O}(T^{3/4})$ whenever $\gamma_T = O(T)$, and the sublinearity claim follows term by term. The two events ($\mathcal{E}$ and the lemma's) fail with probability at most $\delta + \delta'$. □

For the Matérn-$5/2$ kernel $\gamma_T = \tilde{\Theta}\big(T^{d/(d+5)}\big)$ (Vakili et al., 2021), so the $\tilde{O}(T^{3/4})$ term is of lower order than the GP-LCB term only for $d > 5$. For $d \le 5$ the corollary guarantees a sublinear addition rather than preservation of the GP-LCB rate. The comparison is with the GP-LCB bound on the same pools, not with the realized regret of a separate GP-LCB run.

Remark 11 (Why the budget uses the post-correction residual)**.** *The budget of Eq.* (6) *is the scale of the residual the correction leaves unexplained, so it is not monotone in the advisor's usefulness. For an advisor that predicts f exactly, $a_s = r_s$ up to observation noise, the shrunk coefficient approaches $h_m = m/(m+\kappa)$, and the residual it leaves is $(1-h_m)\,r_s = \kappa\,r_s/(m+\kappa)$. After noise deflation, c_{t-1} is then about $2\kappa/m$ times the correction $\hat{\theta}_{t-1}a$ at a typical point, so the cap tightens around a perfect advisor at rate κ/m and would remove its influence as $m \to \infty$. Two facts limit what this costs. First, the guarantees do not depend on this choice. The proof of Corollary 10 uses the budget only through Lemma 7, $V_{t-1}^{\mathrm{res}} \le \frac{1}{m}\sum_s r_s^2$, and Theorem 3 holds for any logged budget, so the budget built from the uncorrected residuals, $\lambda_c\sqrt{\big(\frac{1}{m}\sum_s r_s^2 - \hat{\sigma}^2_{\varepsilon,t-1}\big)_+}$, which is monotone in usefulness, carries the same bounds. Second, at the budgets studied here the cap rarely binds on a real advisor. Over the 3,555 rounds with a nonzero correction in the gpt-4o-mini runs on Bayesmark and HPOBench, the clip is active at the selected point in 0.9% of rounds, and the median budget there is 2.2 times the GP standard deviation. We keep the post-correction residual because it is the more conservative of the two admissible choices. Its cost shows with oracle advice, where CaliBO keeps three fifths to four fifths of the gated mixture's gain (Section 5.3).*

B.6 From Pool Regret to Domain Regret

Theorem 3 controls regret against the best point in each round's finite pool. The uniform component of the pool closes the remaining gap to the domain optimum.

Lemma 12 (Uniform-pool covering)**.** *Let the round-t pool contain $N_t \ge \nu\,t^p$ points drawn uniformly and independently on $[0,1]^d$ (the frozen construction has $\nu = 0.1$, $p = 2$). For any $\delta'' \in (0,1)$, with probability at least $1-\delta''$, simultaneously for all $t \le T$, every $\mathbf{x} \in [0,1]^d$ has a uniform pool point within Euclidean distance*

$$\rho_t \;=\; 2\sqrt{d}\,\Big(\tfrac{1}{N_t}\log\tfrac{\pi^2 t^2 N_t}{6\,\delta''}\Big)^{1/d}.$$

Proof. Set $g_t^d = \frac{1}{N_t}\log\frac{\pi^2 t^2 N_t}{6\delta''}$. If $g_t \ge 1/2$ the claim is immediate: then $\rho_t \ge \sqrt{d}$, the diameter of the box, and the pool is nonempty. Assume $g_t < 1/2$ and partition $[0,1]^d$ into J_t^d axis-aligned cubes

of side $1/J_t$ with $J_t = \lceil 1/(2g_t) \rceil$, so each side is at most $2g_t$ and at least g_t (the logarithm exceeds one, hence $g_t^d \geq 1/N_t$ and $g_t \leq 1$). Each cube has volume at least g_t^d, so a fixed cube receives no uniform point with probability at most $(1 - g_t^d)^{N_t} \leq e^{-N_t g_t^d} = \frac{6\delta''}{\pi^2 t^2 N_t}$, and the number of cubes satisfies $J_t^d \leq g_t^{-d} \leq N_t$. A union bound over the cubes and then over t (via $\sum_t t^{-2} = \pi^2/6$) leaves probability at least $1 - \delta''$ that every cube at every round is occupied. A point in the same cube as $\mathbf{x}$ is within $\sqrt{d} \cdot 2g_t = \rho_t$. □

Corollary 13 (Domain regret)**.** *Let $R_T^{\mathrm{domain}} = \sum_{t=1}^{T} \big(f(\mathbf{x}_t) - \min_{\mathbf{x} \in \mathcal{X}} f(\mathbf{x})\big)$ over all T evaluations, warm start included. Under the assumptions of Theorem 3 together with Assumption 4, with probability at least $1 - \delta - \delta''$,*

$$R_T^{\mathrm{domain}} \;\leq\; n_0 L\sqrt{d} \;+\; R_T^{\mathrm{pool}} \;+\; L \sum_{t \in \mathcal{I}_{\mathrm{LCB}}} \rho_t,$$

with R_T^{pool} bounded by Theorem 3 on the same event and $\sum_t c_{t-1}$ by Corollary 10 at a further probability cost of δ', and with $N_t = \Theta(t^2)$ the covering sum is $L\,\tilde{O}(\log T)$ at $d = 2$ and $L\,\tilde{O}\big(T^{1-2/d}\big)$ at $d > 2$, which stays within the $\tilde{O}(T^{3/4})$ deviation order for every $d \leq 8$, the largest dimension on the suites this analysis is applied to. The 180-dimensional study of Appendix D.1 runs a capped, fixed-size pool and is outside this corollary.

Proof. Let $\mathbf{x}^\star$ minimize f on $\mathcal{X}$ and $\mathbf{u}_t$ be the uniform pool point nearest to it. Each warm-start round contributes at most $L\sqrt{d}$ under Assumption 4, since both points lie in the unit box. For $t \in \mathcal{I}_{\mathrm{LCB}}$, on the event of Lemma 12, $\min_{\mathbf{x} \in \mathcal{C}_t} f(\mathbf{x}) \leq f(\mathbf{u}_t) \leq f(\mathbf{x}^\star) + L\rho_t$ by Assumption 4. Sum over $\mathcal{I}_{\mathrm{LCB}}$ and add R_T^{pool}. With $N_t = \Theta(t^2)$, $\rho_t = \tilde{O}(t^{-2/d})$, and $\sum_t t^{-2/d}$ is $O(\log T)$ at $d = 2$ and $O\big(T^{1-2/d}/(1 - 2/d)\big)$ at $d > 2$, and $1 - 2/d \leq 3/4$ exactly when $d \leq 8$. □

B.7 Scope and the Theory-Implementation Gap

The analysis concerns an idealized variant of the deployed mechanism, and the two differ in two respects. Theorem 3 and Lemma 8 assume a fixed kernel with known noise and use the latent posterior width $\sigma_{t-1}^{\mathrm{lat}}$, while the deployed rule refits the GP online and acquires with the predictive width $\sigma_{\mathrm{GP},t-1}$, which includes the fitted noise. The records are also collected adaptively at selected points while both predictors are refit. The regret statements should therefore be read as describing the idealized variant, not as guarantees for the deployed rule. Lemma 2 does apply to the deployed rule, with $\tilde{\sigma}_{t-1} = \sigma_{\mathrm{GP},t-1}$. As an informal check only, the empirical coverage of the GP confidence intervals at the selected points is 0.954 to 0.975 against a nominal 0.90 on our suites, which suggests the intervals are not too narrow there but does not verify the assumptions.

C Experimental Protocol and Reproducibility

This appendix consolidates the experimental setup: evaluation protocol and metric definitions, the pre-specified zero-prior suite, the contamination-control prompt policy, the synthetic-advisor suite, software and LLM configuration, LLM cost, failure handling, prompt templates, CaliBO hyperparameters, baseline reproduction details including our implementations of the two published two-source constructions, the shared candidate pool, and the GP surrogate.

C.1 Evaluation Protocol and Metric Definitions

Normalization. For each task we collect the best objective value reached within the budget for every method and seed. Per-task min-max normalization rescales each task to $[0, 1]$ following Liu et al. (2024) so that tasks with different raw scales can share one axis. The *reference scale* of a task is its empirical minimum and maximum over twelve methods (CaliBO, the ten baselines of Table 2 and the GP-Mean-Only control), computed once and used for every normalized number reported on that task in Table 2, the design-control comparisons of Appendix D.3, the non-inferiority tests below and the inline differences of Section 5.2. The running-best curves of Figures 2, 3 and 4 share a second scale, the range of every objective value the same twelve methods evaluated on the task, so that each gpt-4o-mini curve lies in $[0, 1]$. The two scales have the same minimum and differ only

in the maximum. Both are computed from the gpt-4o-mini runs alone, and gpt-5.4-mini results are placed on the same scales, so they can fall below 0 on tasks where the stronger backbone finds a better value. A comparison that appears more than once on these suites appears on one scale throughout. The synthetic-advisor suite cannot use the reference scale, because the reference methods were never run in that setting, and uses its own scale (Appendix C.3). The 180-dimensional study reports raw objective values and is not normalized.

Paired head-to-head protocol. All comparisons in Table 2 are paired: for each task, CaliBO and the opponent are compared on the identical seed set $\{42, 123, 456, 1, 2\}$, the five per-seed normalized values are averaged into one number per task per arm, and the per-task differences are tested across tasks with a two-sided Wilcoxon signed-rank test. The unit of inference is the task. Seeds only stabilize each task's mean. A task on which both methods reach the same value counts as a tie, not a win. Reported p values are Holm-adjusted within each suite, over the ten baseline comparisons of Table 2 and separately over the four design controls of Appendix D.3.

Non-inferiority testing. Comparisons where no significant difference is detected are additionally tested for non-inferiority: a one-sided Wilcoxon signed-rank test of the per-task differences shifted by a tolerance margin δ_{NI}, whose null hypothesis is that CaliBO is worse than the opponent by δ_{NI} or more. Rejecting that null is what supports non-inferiority at margin δ_{NI}. The Wilcoxon form is a location test on the distribution of per-task differences, a median-type statement rather than a claim about the mean. The primary margin $\delta_{\mathrm{NI}} = 0.02$ normalized objective was fixed before any of these comparisons ran, and it is applied only on the reference scale. The margin $\delta_{\mathrm{NI}} = 0.05$ is reported as a secondary, looser margin. On the real-world HPO suites the 0.02 test rejects a deficit of that size against every classical baseline except TuRBO, where it does not reject on either suite ($p = 0.21$ on Bayesmark, $p = 0.06$ on HPOBench), so TuRBO is reported as a comparison where no difference is detected rather than as non-inferiority.

Table 3: One-sided non-inferiority p values of CaliBO at margins 0.02 (pre-specified) and 0.05. Smaller is stronger evidence of no deficit.

	vs GP-LCB		vs GP-Mean-Only	
Suite	$\delta_{\mathrm{NI}} = 0.02$	$\delta_{\mathrm{NI}} = 0.05$	$\delta_{\mathrm{NI}} = 0.02$	$\delta_{\mathrm{NI}} = 0.05$
Bayesmark	0.002	0.0002	0.005	0.001
HPOBench	$< 10^{-4}$	$< 10^{-4}$	$< 10^{-4}$	$< 10^{-4}$
GP sample paths	0.027	0.002	0.059	0.008
Rotated classics	0.190	0.055	0.117	0.005

Anytime curves. Anytime plots aggregate task-first, the per-task mean over seeds and then the mean over tasks, matching the tables' unit of inference. Bands are 95% bootstrap intervals over tasks (2,000 resamples), so seeds of one task are never treated as independent evidence.

C.2 Pre-Specified Zero-Prior Suite

The zero-prior suite consists of 30 tasks whose construction seeds were fixed before any cell was run, with a standing rule that no design change may be based on their results.

GP sample-path tasks (18). Each task is a draw from an approximate GP prior via random Fourier features with $D = 256$ features: $f(\mathbf{x}) = \sqrt{2/D}\sum_j w_j \cos(\boldsymbol{\omega}_j^\top \mathbf{x} + b_j)$ with $w_j \sim \mathcal{N}(0, 1)$ and $b_j \sim \mathrm{Uniform}[0, 2\pi]$. Spectral frequencies are Gaussian for the RBF kernel and multivariate Student-t with five degrees of freedom for Matérn-5/2, scaled by the inverse lengthscale. The fixed grid is kernel $\in$ {RBF, Matérn-5/2} $\times$ $d \in \{2, 4, 8\}$ $\times$ lengthscale $\in \{0.10, 0.25, 0.50\}$. The task seed is the registered base seed plus the grid index.

Rotated classic instances (12). Six function-dimension pairs (sphere, Rosenbrock and Rastrigin at $d = 3$, Ackley, Griewank and Rosenbrock at $d = 6$), two instances each. Every instance draws an optimum location $\mathbf{x}_{\mathrm{opt}} \sim \mathrm{Uniform}[0.2, 0.8]^d$ and a rotation matrix from the QR decomposition

of a standard Gaussian matrix, and evaluates the classic function on $\mathbf{v} = R\left((\mathbf{u} - \mathbf{x}_{\text{opt}}) \cdot 10\right)$, so memorized optimum locations of the named functions are uninformative. The domain is $[0, 1]^d$ throughout, all coordinates continuous, minimization.

Contamination-control prompt policy. No prompt in the main experiments contains dataset names, model names, task names, or function names. The LLM sees a generic minimization framing, anonymized parameter names, box bounds, and the numeric observation history, and nothing else. On the zero-prior suite this makes the advisor's prior knowledge unusable by construction. On the real HPO suites it removes the direct memorization channel.

C.3 Synthetic Advisor Suite

The experiments of Section 5.3 run on the 30 zero-prior tasks of Appendix C.2 with seeds $\{42, 123, 456, 7, 13\}$, budget $T = 25$, the candidate pool of Section 3.4 with uniform candidates only, and the acquisition rule of Eq. (7). No LLM is called. Each advisor returns a value at every candidate of the round-t pool, computed from the true objective f at that candidate. Write $\bar{f}_t$ for the mean of f over the pool and τ_t for the range of the observed values.

Advisors.

- **Oracle**: f.
- **Noisy**: $f + \tau_t z$, with $z \sim \mathcal{N}(0, 1)$ drawn once per candidate location.
- **Oracle, noisy observations** and **noisy, noisy observations**: the first two advisors while each observed value carries Gaussian noise with standard deviation 5% of the running observed range. Their reference is plain GP-LCB under the same observation noise.
- **Echo**: f at the quarter of candidates closest to evaluated points, measured in standardized coordinates, and $2\bar{f}_t - f$ elsewhere.
- **Inverted (linear)**: $2\bar{f}_t - f$.
- **Inverted (sign)**: $2\bar{f}_t - f$ on one side of a random hyperplane through the centroid of the warm-start points, and f on the other.
- **Warped**: a monotone convex transform of f that preserves its ranking, $\min f + (\max f - \min f)\,(e^{3u} - 1)/(e^3 - 1)$ with u the min-max scaled value over the pool.
- **Local corruption**: $f - 2\tau_t \exp\!\big(-\|\mathbf{x} - \mathbf{x}_h\|^2/(2\varrho^2)\big)$, a false basin around a point $\mathbf{x}_h$ far from the warm start, with ϱ equal to 15% of the domain diagonal.
- **Drifting**: $f + \tau_t z_t$ with the noise redrawn at every round.
- **Shrunk toward the pool mean**: $f + 2\lambda(\bar{f}_t - f)$, which is half-strength advice at $\lambda = 25\%$, the constant pool-mean advisor at $\lambda = 50\%$, and half-strength inverted advice at $\lambda = 75\%$.

Mechanisms. All mechanisms share the GP, the pool and the acquisition rule, and differ only in the mean used for acquisition.

- **CaliBO**: Eqs. (2) to (6) with the constants of Table 5.
- **Accuracy-Based Calibration**: the same records, with the coefficient fitted by shrunk least squares of $Y_s = r_s + b_s$ on $\ell_s = a_s + b_s$, applied to the disagreement, and the budget computed from the residuals of that fit.
- **Nonnegative CaliBO**: CaliBO with $\hat{\theta}_{t-1}$ replaced by $\max(\hat{\theta}_{t-1}, 0)$.
- **Direct surrogate**: the advisor's value replaces the GP mean, with the GP width kept.
- **Gated decaying mixture**: $(1 - \eta_t)\,\mu_{\text{GP},t-1} + \eta_t\,\mu_{\text{adv},t-1}$ with $\eta_t = t^{-1/2}(1 - w_t)$. Every four evaluations the gate compares, at each evaluated point, the error of a leave-one-out GP prediction with that of the advisor. With n points of which the GP wins k, it sets $w_t = k/n$ when a one-sided binomial test of k against one half is significant at level $0.5/K$, with K the number of refreshes in the run, and $w_t = 0$ otherwise.
- **LABO-style calibration**: at every round, a least-squares coefficient ρ between the advisor's values and the outcomes at evaluated points, a GP on the residuals $y - \rho\,\mu_{\text{adv}}$, and the mean $\rho\,\mu_{\text{adv}} + \mu_{\text{disc}}$ with the discrepancy GP's width.

Scale and testing. Differences are against plain GP-LCB, per task min-max normalized over all cells of the original synthetic-advisor run, averaged over seeds within a task and then over tasks, with a two-sided Wilcoxon test across tasks. CaliBO, the gated mixture and LABO-style calibration on six advisors come from that original run. Accuracy-Based Calibration, Nonnegative CaliBO, the direct surrogate, and LABO-style calibration on the remaining seven advisors were added later on the same tasks, seeds and scale, with the GP optimizer's random restarts seeded per cell.

C.4 Software, Seeds, and LLM Configuration

All experiments run on a single Linux CPU workstation. Two seed sets are in use and they are not interchangeable: every arm on the real-advisor suites, Bayesmark, HPOBench and the zero-prior families, uses $\{42, 123, 456, 1, 2\}$, while the synthetic-advisor suite of Section 5.3 uses $\{42, 123, 456, 7, 13\}$, fixed when that suite was registered. The main-result advisor is `gpt-4o-mini`. The second backbone is `gpt-5.4-mini`, run on the first three seeds of the real-advisor set. Sampling uses temperature 0.8 for candidate generation (completion cap 2000 tokens) and 0.3 for surrogate prediction (completion cap 1000 tokens, at most 25 candidates per call), and candidate-generation calls are chunked so that a response never carries more configurations than fit the completion cap. Both limits exist because truncated responses are unparseable and silently discard the whole call otherwise.

C.5 LLM Cost Accounting

Table 4 reports per-run means over the 245 runs behind Table 2 for every method that queries the advisor, priced at gpt-4o-mini list rates. At gpt-4o-mini rates CaliBO spends about $0.03 per run, and about $0.19 at gpt-5.4-mini rates for the same usage, with the same ordering across methods. It spends about a quarter less than LLAMBO and forty percent less than LABO, because the calibration adds no calls of its own: records reuse the predictions the acquisition already needed, so the whole cost sits in the shared prediction and generation calls.

Table 4: Per-run advisor usage on the real-world HPO suites at $T = 25$ under gpt-4o-mini: mean LLM calls, tokens, wall-clock time, and cost at the list rates of gpt-4o-mini ($0.15 and $0.60 per million prompt and completion tokens) and gpt-5.4-mini ($0.75 and $4.50) for the same token counts. Classical baselines issue no calls.

				$ per run		
Method	calls	prompt tokens	completion tokens	gpt-4o-mini	gpt-5.4-mini	seconds per run
CaliBO	75	103,000	24,500	0.030	0.188	537
LLAMBO	65	91,700	44,900	0.041	0.271	627
LLINBO	22	16,100	4,600	0.005	0.033	98
LABO	156	261,500	19,200	0.051	0.283	796
MISO	156	260,900	19,200	0.051	0.282	666
GP-Mean-Only	20	15,200	17,500	0.013	0.090	265

C.6 Failure Handling and Response Compliance

Surrogate-prediction calls are chunked at 25 candidates. Measured over the 13,375 prediction calls behind the real-world HPO main table, responses use 124 completion tokens on average (95th percentile 143, maximum 153) against a 1000-token cap, so truncation is not a factor: only 3 responses in 13,375 failed to parse. In 6.1% of calls the response is a well-formed numeric array of the wrong length, an instruction-compliance failure rather than a truncation. Such a chunk is discarded and the affected candidates keep the fallback value used before any advice arrives, the mean of the observed objectives, so a failed chunk contributes an uninformative constant rather than a spurious prediction. The same policy applies to every arm that queries the advisor, and its effect on the fused mean is bounded by the influence budget like any other advice. Candidate-generation calls are chunked by dimension so that a response never carries more configurations than fit the cap. No generation call failed to parse in the main runs, and 0.41% of the 100,275 generated configurations required clipping to the box bounds.

Degenerate cases are defined explicitly. Before $m_{\min}$ settled records, and whenever $\sum_s a_s^2$ is numerically zero, the coefficient is $\hat{\theta} = 0$ and the fused mean is exactly the GP mean. The failed-chunk fallback is the mean of observed objectives rather than $\Delta = 0$: a $\Delta = 0$ fallback would present a failed candidate as one the advisor endorses at exactly the GP's value, while the constant fallback yields a mean-reversion signal that the same coefficient prices and the same budget bounds. Neither choice can move the fused mean by more than c_{t-1}.

C.7 Prompt Templates

CaliBO issues two kinds of LLM call: candidate generation (the LLM share of the warm start and of the per-iteration pool) and surrogate prediction ($\mu_{\mathrm{LLM},t-1}$ on the candidate pool). The same two templates are used for every task and suite. Only the parameter bounds, the shuffled last-20 history, and the candidate list change per call.

Candidate generation (temperature 0.8, max 2000 tokens).

```
You are an expert black-box optimisation assistant.
We are minimising an objective function with these parameters:
{parameter_bounds}

Previous evaluations (most recent shuffled):
{shuffled_last_20_history}

Current best objective: {best_obj}
Target objective: {0.9 * best_obj}
  # 1.1 * best_obj when best_obj <= 0

Propose {n_candidates} diverse candidate parameter sets that are likely
to achieve the target or better. Return ONLY a JSON array of objects,
each with keys matching the parameter names.

Example:
[{"x0": 1.5, "x1": 2.3}, ...]
```

Surrogate prediction (temperature 0.3, max 1000 tokens, at most 25 candidates per call).

```
You are an expert black-box optimisation surrogate.
Parameters:
{parameter_bounds}

Training data:
{last_20_history}

Predict the objective value for each candidate below.
  Candidate 0: (x0=..., x1=..., ...)
  Candidate 1: (x0=..., x1=..., ...)
  ...

Return ONLY a JSON array of numbers (predicted objective values),
one per candidate, in order.

Example: [1.23, 4.56, ...]
```

C.8 CaliBO Hyperparameters

Table 5 lists the full configuration. Every constant was chosen on a separate design task set, frozen before the main and synthetic-advisor experiments, and is shared across all suites and seeds. There

is no per-task tuning. The use of $\sigma_{\mathrm{GP},t-1}$ for exploration is an architectural invariant, not a tunable choice.

Table 5: CaliBO configuration used in every run of this paper. No constant is tuned per task or per suite.

Parameter	Value	Notes
Calibration activation $m_{\min}$	5	Fused mean equals the GP mean until five records are settled
Shrinkage constant κ	4	$\hat{\theta}$ multiplies the least-squares slope by $m/(m+\kappa)$
Budget multiplier λ_c	2	$c_{t-1} = \lambda_c \sqrt{(V_{t-1}^{\mathrm{res}} - \hat{\sigma}^2_{\varepsilon,t-1})_+}$
Coefficient tolerance ε_a	10^{-12}	$\hat{\theta}_{t-1} = 0$ whenever $\sum_s a_s^2 \le \varepsilon_a$, so advice indistinguishable from the GP receives no influence
Noise deflation $\hat{\sigma}^2_\varepsilon$	fitted	Fitted observation-noise level of the GP, rescaled to the raw objective units
Pool constants $(n_{\mathrm{L}}, n_{\mathrm{U}}, \nu, p)$	$(5, 5, 0.1, 2)$	Per-step pool: $n_{\mathrm{L}} d$ LLM candidates plus $n_{\mathrm{U}} d + \lceil \nu\, t^p \rceil$ uniform candidates, Section 3.4
Acquisition	LCB	Eq. (7) with GP confidence parameter $\delta = 0.1$
Warm-start length n_0	$d+1$	Uniform-LLM hybrid, correction inactive during warm start
Warm-start uniform ratio	0.3	30% uniform, 70% LLM-generated warm-start points
Prediction batch size	25	Maximum candidates per surrogate-prediction call

C.9 Baseline Reproduction Details

Canonical library implementations with default hyperparameters are used except where noted. All LLM-based arms share the backbone, the per-task parameter spaces, the seeds, and the prompt policy above.

GP-EI (Jones et al., 1998). Matérn-5/2 GP with ARD length scales (scikit-learn (Pedregosa et al., 2011)), EI optimized by L-BFGS-B (Byrd et al., 1995) from 20 restarts seeded by 200 Latin-hypercube candidates (McKay et al., 1979). Warm start $n_0 = d + 1$ uniform points.

GP-LCB (Srinivas et al., 2010). Two variants. The plain variant issues no LLM calls at all: uniform warm start, uniform candidate pool of the same size schedule, GP mean and uncertainty, LCB with the same β_t. The GP-Mean-Only control keeps CaliBO's hybrid LLM-uniform candidate pool and uses no mean correction, so the LLM enters it only through the candidates.

SMAC3 (Lindauer et al., 2022), **Optuna-TPE** (Akiba et al., 2019), **HEBO** (Cowen-Rivers et al., 2022), **TuRBO-1** (Eriksson et al., 2019). Default facades and samplers of the respective releases, single-fidelity, identical budgets.

LLAMBO (Liu et al., 2024). The published three-component design, run as specified: LLM candidate generation with target-score interpolation, LLM surrogate prediction with uncertainty from three shuffled-history orderings, EI over LLM-generated candidates, per-step candidate count $10d + \lceil 0.1t^2 \rceil$.

LLINBO (Chang et al., 2025). LLM proposer filtered by a GP-UCB trust region under the published schedule and shrinking radius.

LABO (Chen et al., 2026). No public implementation was available at the time of writing. The scoring construction follows the paper: ρ is estimated by least squares over pairs of LLM predictions and realized outcomes at evaluated points, and the LLM fidelity enters the posterior variance as $\sigma_R^2 = \rho^2 \sigma_L^2 + \sigma_\delta^2$ with a discrepancy GP δ. Prediction timing is faithful to the published algorithm: the prediction paired with an outcome is the one cached when the point was selected, before its outcome existed, and the advisor is never re-queried at evaluated points. Two adaptations are implementation choices we made when porting the construction into the shared experimental framework, and are not properties of the original. First, the published gate decides whether a candidate receives a real evaluation at all, with the LLM prediction standing in as a low-fidelity observation. In a protocol where every method spends exactly T real evaluations, the gate instead falls back to the plain GP surrogate at that candidate ($p_\Delta = \sigma_\delta^2/(\rho^2\sigma_L^2 + \sigma_\delta^2) > 0.75$). Second, only selected points receive evaluations here, so the ρ pairs accumulate over selected points and the uniform initial design

contributes none. Remaining operational choices (LLM predictive spread from three perturbed chunked calls, the shared candidate pool below, LCB with the shared β_t) are fixed a priori and identical across all runs.

MISO, drifting source (Poloczek et al., 2017). A joint-covariance multi-information-source model with the LLM as a drifting source: the source discrepancy kernel is $\sigma^2_{\text{disc}} \cdot \text{Matérn}(\mathbf{x}) \cdot \text{RBF}(t)$, with time lengthscale 5, over an augmented source-time state, so the model can learn both the scale and the drift rate of the advisor's error. Source weights emerge from the joint posterior. Predictions require at least three settled source-target pairs before the source enters the posterior.

Shared candidate pool. CaliBO, the GP-Mean-Only control, LABO, and MISO all draw candidates under the identical pool construction ($n_{\text{L}}d$ LLM plus $n_{\text{U}}d + \lceil \nu t^p \rceil$ uniform candidates with $(n_{\text{L}}, n_{\text{U}}, \nu, p) = (5, 5, 0.1, 2)$, Section 3.4), so candidate coverage after the initial design is matched in distribution. LABO and MISO initialize by uniform sampling of $d + 1$ points, following their pipeline convention. LLAMBO and LLINBO retain their published pool conventions, reported above.

C.10 GP Surrogate

All GP-based arms share one surrogate configuration: Matérn-5/2 kernel with ARD per-dimension length scales plus a WhiteKernel observation-noise term initialized at 0.1, all hyperparameters refit by marginal likelihood after every evaluation (three optimizer restarts, jitter 10^{-6}), outputs standardized to zero mean and unit variance. The fitted WhiteKernel level, mapped back to raw objective units through the output scaler, is the $\hat{\sigma}^2_\varepsilon$ used by the noise-deflated influence budget.

D Additional Results

D.1 High-Dimensional Behavior on LassoBench

LassoBench (Šehić et al., 2022) tunes one regularization weight per feature of a weighted Lasso, so the dimension of the search space is the number of features and the objective is the cross-validation loss of the model fitted at that weight vector. The `dna` instance has 180 parameters on $[-1, 1]^{180}$. Coordinates carry no semantics an advisor could recall, each being the log regularization weight of an anonymous feature index, so this suite is zero-prior by construction in the same sense as the zero-prior suite of Appendix C.2.

Three settings differ from the main protocol and apply identically to every arm that issues LLM calls. The budget is $T = 100$ rather than 25, because 25 evaluations in 180 dimensions leaves every arm at its random-search value. The warm start is capped at 20% of the budget, since $d + 1$ points would exceed the whole budget. The advisor's share of the candidate pool is capped at 20 configurations and the pool at 300, because $5d$ would ask one call to emit 900 configurations, and proposal requests are split into chunks of at most 250 numbers, which at this dimension is one configuration per call. Results therefore describe a capped-pool variant. The GP-Mean-Only control, run under the same settings, reaches a best loss of 0.317 ± 0.003 against 0.313 ± 0.008 for CaliBO, which is better on two of the five seeds.

D.2 Synthetic Advisors: Full Matrix

Figure 7 reports all thirteen advisors of Appendix C.3 for CaliBO and the two reference mechanisms run on the full matrix.

D.3 Design Controls

On the reference scale with five seeds, CaliBO's mean difference against GP-Mean-Only is -0.035 on Bayesmark (17 of 25 tasks won, $p = 0.109$) and -0.090 on HPOBench (21 of 24, $p < 10^{-4}$), and against Accuracy-Based Calibration it is -0.056 (21 of 25, $p = 0.003$) and -0.068 (19 of 24, $p = 0.001$). Negative favors CaliBO. The p values are Holm-adjusted within each suite over four design controls, these two and two alternative influence rules that use the same records, neither of which differs significantly from CaliBO. On the zero-prior suites, where these tests are not Holm-adjusted, the two-sided p values are 0.246 on the sample paths and 1.000 on the rotated classics against

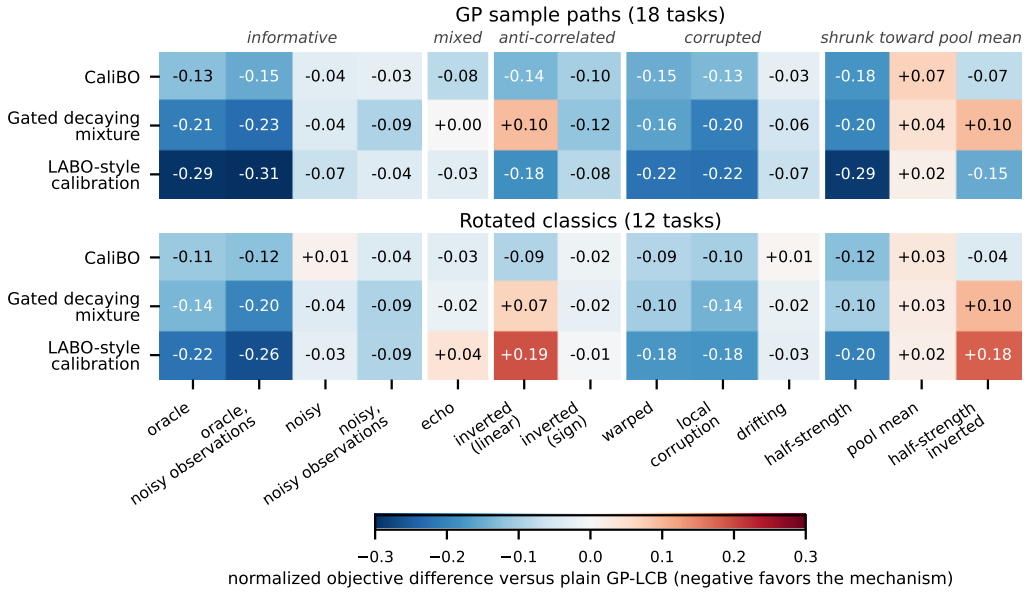

Figure 7: All thirteen synthetic advisors: final difference against plain GP-LCB, negative favoring the mechanism, with advisors grouped by type.

GP-LCB, and 0.468 and 0.677 against GP-Mean-Only. The *GP-confidence cap* keeps CaliBO's coefficient but replaces the residual budget by the per-candidate limit $\sqrt{\beta_t}\,\sigma_{\mathrm{GP},t-1}(\mathbf{x})$. *Error-weighted mixing* replaces the signed correction by the convex mean $(1-\eta)\,\mu_{\mathrm{GP},t-1} + \eta\,\mu_{\mathrm{LLM},t-1}$, with $\eta = e_{\mathrm{GP}}/(e_{\mathrm{GP}} + e_{\mathrm{LLM}})$ built from the exponentially weighted (decay 0.9) mean absolute errors $|r_s|$ and $|r_s - a_s|$ of the two predictors at evaluated points.