TRADING ADAPTIVE DEPTH FOR PARALLEL WIDTH IN PRIOR-GUIDED SEARCH

Anonymous authors
Paper under double-blind review

ABSTRACT

Automated machine-learning (AutoML) search consumes both evaluations and feedback rounds. When evaluations can run in parallel, waiting for feedback between batches can dominate elapsed time even when the total number of evaluations is unchanged. We study how much adaptive depth is needed to reach a target regret when each round can evaluate b candidates in parallel. We characterize this tradeoff through the iso-regret frontier $b \mapsto N^\star(\varepsilon, b, \delta)$, which under regularity, coverage, and calibration conditions satisfies $N^\star(\varepsilon, b, \delta) = O\Big(1 + \frac{d_{\text{eff}} \log(W/\varepsilon)}{\log(b/\nu_{\text{eff}})}\Big)$. Here d_{eff} describes how rapidly near-optimal mass vanishes under a fixed reference prior, while ν_{eff} summarizes losses in useful within-round coverage. The result gives a sufficient regime in which increasing parallel width can reduce the number of feedback rounds. Motivated by this characterization, we propose Prior-Weighted Bayesian Optimization (PWBO), which combines prior-weighted proposals, diversity-aware batch selection, and confidence-based elimination. In controlled synthetic experiments that vary prior quality and within-round coverage, the observed round counts follow the predicted reciprocal-log frontier, and the fitted difficulty parameter increases across priors as the theory predicts. We then study PWBO on a fully enumerated Avazu click-through-rate (CTR) pipeline testbed, where enumeration provides an exact reachable comparator while each search evaluation still requires model training. On this testbed, PWBO reaches the target in fewer feedback rounds than competing BO baselines.

1 INTRODUCTION

Automated machine-learning (AutoML) search is often optimized for evaluation count, but evaluation count ignores a second cost: feedback rounds. Two procedures can both train 64 candidate pipelines, yet one receives feedback 64 times while the other evaluates 16 candidates in parallel and receives feedback only four times. The total evaluations are the same, but the elapsed time differs sharply when evaluations are expensive and parallel compute is available. This distinction is especially relevant in pipeline search with expensive evaluations, where candidate generation is cheap relative to training and evaluation, and proposal strategies can be updated only after results return (Chan et al., 2025; Toledo et al., 2025; Trirat et al., 2025).

We study evaluation slots and feedback rounds as separate resources. Let N denote the number of adaptive rounds and b the number of evaluation slots per round, with total slot budget $T = Nb$. Our question is whether a larger batch can compensate for fewer opportunities to adapt: at a fixed target quality, how does the required number of feedback rounds change with b, and how does a generative prior affect that tradeoff? Classical Bayesian optimization usually measures efficiency in evaluations (Jones et al., 1998; Snoek et al., 2012; Shahriari et al., 2016), while recent agentic search work often emphasizes final performance (Toledo et al., 2025). Neither makes feedback-round complexity the main quantity of interest.

We formalize this question with the iso-regret frontier $b \mapsto N^\star(\varepsilon, b, \delta)$. Under the assumptions in Section 4,

$$N^\star(\varepsilon, b, \delta) = O\left(1 + \frac{d_{\text{eff}} \log(W/\varepsilon)}{\log(b/\nu_{\text{eff}})}\right).$$

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.

The quantity d_{eff} describes how quickly near-optimal prior mass decreases as the target becomes tighter. The factor ν_{eff} accounts for the part of nominal batch width that fails to provide useful, non-redundant coverage. Together they characterize the regime in which increasing width can reduce adaptive depth.

The same characterization suggests a practical search rule. Good proposals should inherit the reference prior's useful bias, candidates selected within a round should avoid unnecessary redundancy, and pruning should remove designs only when the surrogate provides enough evidence. Prior-Weighted Bayesian Optimization (PWBO) implements these choices with prior-weighted proposals, a repulsive batch-selection term, and confidence-based elimination.

Our experiments separate the tradeoff itself from the performance of PWBO. Controlled problems first test whether wider batches reduce the number of feedback rounds at fixed evaluation budgets. We then vary prior quality and effective batch coverage to test when the tradeoff strengthens or disappears. On Avazu CTR pipeline search, we compare PWBO with random search, BO, πBO, LLAMBO, and LP-BO. A separate fixed-prior control tests whether GP updating improves search when further localization is needed, while component ablations examine coverage and elimination. We additionally evaluate transfer to NAS-Bench-201, a larger search space in a different domain.

Our contributions are summarized as follows:

- We formulate search efficiency as an **iso-regret frontier** over adaptive depth and parallel width, making explicit how many feedback rounds are needed at a given batch width.
- We derive a **conditional upper bound** on this frontier in which prior concentration and effective coverage enter as separate factors.
- We propose **PWBO** and test whether the mechanisms suggested by the analysis appear in controlled, CTR pipeline, and NAS-Bench-201 search settings.

2 Related Work

Automated pipeline search. Classical AutoML optimizes a fixed pipeline space with model-based or evolutionary search (Thornton et al., 2013; Feurer et al., 2015; Olson et al., 2016). Agentic systems instead use a language model to generate executable pipeline designs and are commonly evaluated on benchmarks with relatively inexpensive objective evaluations (Chan et al., 2025; Trirat et al., 2025; Guo et al., 2024; Toledo et al., 2025); LLM-as-optimizer and program-discovery methods (Yang et al., 2024; Romera-Paredes et al., 2024; Lu et al., 2024) likewise study regimes with many inexpensive evaluations. These analyses primarily index performance by evaluation count; ours explicitly indexes it by adaptive feedback rounds, and studies how prior structure and geometric coverage shape that round complexity.

Priors in Bayesian optimization. πBO (Hvarfner et al., 2022) multiplies an acquisition function by a user-specified prior over the optimizer's location and gradually decays that influence; BOPrO (Souza et al., 2021) represents an analogous belief through a pseudo-posterior, and PriorBand (Mallik et al., 2023) incorporates prior information into multi-fidelity scheduling. These methods use hand-specified densities on vectorized spaces, whereas our fixed reference is a generative distribution over structured programs, and PWBO's log-prior term arises from a KL-regularized proposal distribution rather than a multiplicative belief. Our conditional bound links the exponent d_{eff} to the reference mass of objective-dependent near-optimal sets, rather than to prior concentration alone.

Language models inside the optimizer. LLAMBO (Liu et al., 2024) uses an in-context LLM for initialization, surrogate modeling, and candidate generation, with Expected Improvement used for selection; LICO (Nguyen & Grover, 2025) places language models in other optimizer roles. These works do not derive an adaptive-round complexity of the form $N^\star(\varepsilon, b, \delta)$.

Batches, fidelity, and limited adaptivity. Batch Bayesian optimization uses joint selection or repulsion to reduce redundant evaluations within one round (Kathuria et al., 2016; González et al., 2016); we use the same device to target geometric coverage. Multi-fidelity methods (Li et al., 2018; Falkner et al., 2018; Awad et al., 2021) introduce fidelity as a third resource axis beyond slot width

and adaptive depth. Limited-adaptivity results for finite arm sets (Perchet et al., 2016; Gao et al., 2019; Agarwal et al., 2017) parameterize adaptivity by the number of arms. We instead study the iso-regret frontier over a structured, prior-guided reachable domain, separating prior-relative difficulty from geometric coverage overhead through d_{eff} and ν_{eff}.

3 Problem Formulation: The (N, b) Frontier

Let ζ denote the architecture and training choices that specify a pipeline design; we refer to ζ as its design signature. One noisy evaluation returns $y = \bar{f}(\zeta) + \xi$, where $\bar{f}(\zeta) = \mathbb{E}[y \mid \zeta]$ and $\mathbb{E}[\xi \mid \zeta] = 0$. The fixed reference prior π_θ[1] induces the reachable domain $\mathcal{R}_0 := \text{supp}(\pi_\theta)$, and all evaluated and reported signatures are restricted to $\mathcal{R}_0$. Assume $\bar{f}$ attains finite maximum and minimum values there, choose $\zeta^\star \in \arg\max_{\zeta \in \mathcal{R}_0} \bar{f}(\zeta)$, and define the finite positive range $W := \max_{\zeta \in \mathcal{R}_0} \bar{f}(\zeta) - \min_{\zeta \in \mathcal{R}_0} \bar{f}(\zeta) > 0$. The simple regret of a reported signature $\widehat{\zeta}$ is $r(\widehat{\zeta}) := \bar{f}(\zeta^\star) - \bar{f}(\widehat{\zeta})$.

One *evaluation slot* pays for one complete noisy evaluation. A search has b such slots per round and N adaptive rounds, with every assignment in round n fixed before any outcome from that round is observed. We call N the adaptive depth, b the *slot width*, and $T = Nb$ the total number of slots. Let $m_n \leq b$ be the number of distinct signatures assigned to round n. Re-evaluating one signature consumes another slot but does not increase m_n; if each of the m_n signatures receives at least ℓ_n evaluations, then $m_n \ell_n \leq b$. The attainable covering radius depends on both m_n and where those signatures are placed in the design metric. Replication does not change the observation-noise distribution, but averaging ℓ_n replicates reduces uncertainty in the mean at each repeated signature.

Definition 1 (Round complexity and frontier)**.** $N^\star(\varepsilon, b, \delta) \in \{1, 2, \ldots\}$ is the minimum number of rounds for which there exists an adaptive policy using at most b slots per round and guaranteeing $\Pr(r(\widehat{\zeta}) > \varepsilon) \leq \delta$. The probability is over evaluation noise and policy randomization. For fixed (ε, δ), the map $b \mapsto N^\star(\varepsilon, b, \delta)$ is the *iso-regret slot-width frontier*.

At a theoretical frontier point, the adaptive policy uses a total slot budget $T = bN^\star(\varepsilon, b, \delta)$. We call this an *adaptivity-beneficial regime* when T independent draws from the fixed reference cannot attain regret at most ε with probability at least $1 - \delta$.

4 Conditions for the Depth-Width Exchange

Two properties determine whether a wider batch can reduce the number of adaptive rounds. Good designs must receive enough probability under the reference prior, and additional evaluation slots must provide non-redundant coverage of the plausible set. We formalize these two effects below.

4.1 Two bottlenecks

Prior quality. For tolerance ε, let

$$\mathcal{Z}_\varepsilon := \{\zeta \in \mathcal{R}_0 : r(\zeta) \leq \varepsilon\}, \qquad p_\varepsilon := \pi_\theta(\mathcal{Z}_\varepsilon).$$

We summarize how quickly this mass vanishes as the target tightens by

Definition 2 (Prior-relative near-optimality dimension)**.**

$$d_{\text{eff}} := \sup_{\varepsilon \in [\varepsilon_{\min}, \varepsilon_{\max}]} \frac{\log(1/p_\varepsilon)}{\log(W/\varepsilon)}. \tag{1}$$

Equivalently, $p_\varepsilon \geq (\varepsilon/W)^{d_{\text{eff}}}$ on the resolved tolerance range. A smaller d_{eff} means that the prior makes near-optimal designs easier to reach.

[1]We use $\pi_\theta(\zeta)$ for the probability mass at a signature and $\pi_\theta(\mathcal{S}) := \Pr_{\zeta \sim \pi_\theta}(\zeta \in \mathcal{S})$ for the probability assigned to a set of signatures $\mathcal{S}$.

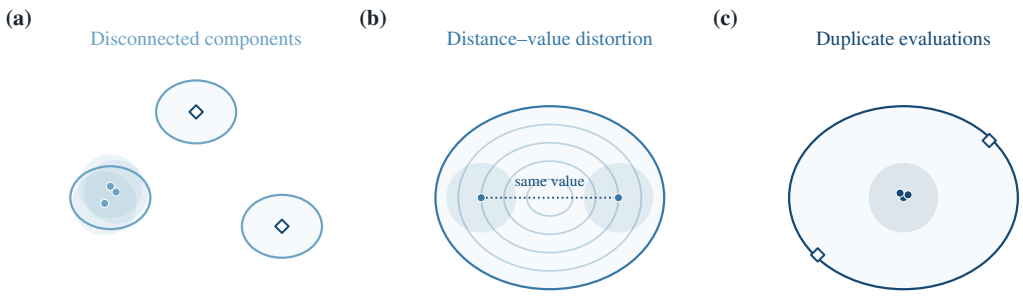

Figure 1: **Three ways a batch can lose effective coverage. (a)** Disconnected regions: evaluations cluster in one region and leave others uncovered. **(b)** Distance–value distortion: designs far apart in the representation have similar objective values, so geometric spread does not translate to value diversity. **(c)** Duplicate or clustered evaluations: several batch slots evaluate the same or nearby designs, reducing the effective number of distinct evaluations. All three raise ν_{eff} and weaken the depth–width exchange.

Effective coverage. A batch of m distinct designs is useful only insofar as those designs cover different parts of the current plausible set. We measure coverage in a representation $z(\zeta)$ and require nearby designs in this representation to have similar objective values. Concretely, the covering radius is the largest representation-space distance from a plausible design to its nearest batch member; a smaller radius means that no part of the plausible set is far from the batch. Duplicate or clustered evaluations, disconnected plausible regions, and a poor match between representation distance and objective variation all reduce the benefit of a wider batch (Figure 1). We collect these losses in $\nu_{\text{eff}} \geq 1$: heuristically, m distinct evaluations behave like m/ν_{eff} useful covering locations. Let $w_n := \sup_{\zeta \in \mathcal{R}_n} r(\zeta)$ denote the largest remaining regret, or value slack, after round n. Under the regularity and coverage conditions, one round contracts this slack as

$$w_n \lesssim \left(\frac{\nu_{\text{eff}}}{m}\right)^{1/d_{\text{eff}}} w_{n-1} + \text{statistical error}. \tag{2}$$

Thus, increasing m reduces the required depth only when the additional evaluations provide useful coverage. Geometric dispersion alone is insufficient if distance in the representation does not correspond to objective variation.

4.2 A CONDITIONAL EXCHANGE LAW

The exchange result uses four conditions. Near-optimal sets must retain the prior mass summarized by d_{eff}. The m evaluated designs in each round must cover the current plausible set up to the overhead ν_{eff}. Differences in the mean objective $\bar{f}$ must be controlled by distance in the representation used to define that coverage. Finally, the confidence bounds used for elimination must hold simultaneously over the search horizon.

Iterating Equation 2 gives the following sufficient-round envelope.

Theorem 1 (Conditional depth–width exchange)**.** *Fix the tolerance range in Equation 1 and suppose the four conditions above hold. If each round evaluates m distinct designs, the statistical floor is below a fixed fraction of ε, and $m > \nu_{\text{eff}}$, then*

$$N^\star(\varepsilon, b, \delta) = O\left(1 + \frac{d_{\text{eff}} \log(W/\varepsilon)}{\log(m/\nu_{\text{eff}})}\right). \tag{3}$$

When every slot evaluates a distinct design, $m = b$.

The bound characterizes a sufficient regime in which greater near-optimal prior mass reduces the required depth, while poor effective coverage reduces the benefit of increasing width. When coverage does not translate into value progress, the exchange can disappear and performance can revert toward a total-budget regime. The formal assumptions are given in Appendix A.2.1 and the proof in Appendix A.2.2.

5 FROM CHARACTERIZATION TO SEARCH DESIGN

The conditions required by the bound motivate three practical requirements for shallow-adaptation search:

1. **Exploit favorable prior mass** so that expensive evaluations start in promising parts of the reachable space.
2. **Avoid redundant evaluations within a round** so that additional width produces additional coverage.
3. **Prune conservatively** so that adaptation shrinks the plausible set without discarding the best reachable design.

PWBO maps each requirement to a concrete mechanism. Requirement (1) uses a prior-weighted proposal mixture that favors high-probability signatures. Requirement (2) adds a pairwise repulsion term so that batch selections spread in the embedding space rather than duplicate the same design. Requirement (3) uses confidence-based elimination that removes a signature only when its upper confidence bound falls below the best surviving lower bound.

While our implementation uses a frozen language model as the generative source for both the fixed reference prior and the history-conditioned proposal, the analysis in Section 4 applies to any generative model that supplies a reference distribution over structured candidates.

5.1 PWBO IN ONE ROUND

At round n, PWBO first draws a cheap candidate pool from a mixture of the history-conditioned generative proposal and the fixed reference prior,

$$q_n(\zeta) = (1-\eta)\pi_\theta(\zeta \mid H_n) + \eta\pi_\theta(\zeta), \qquad \eta \in (0,1]. \tag{4}$$

In our implementation, $\pi_\theta(\cdot \mid H_n)$ is realized by re-prompting the frozen language model with the observed history table (Appendix D.5); no fine-tuning is performed. After deduplication and filtering by the current plausible set, the eligible signatures form Σ_n. Let $S \subseteq \Sigma_n$ denote the signatures already selected in the current batch, initialized as $S = \varnothing$ and updated after each greedy step. PWBO then selects the next distinct signature using

$$\zeta_{\text{next}} = \arg\max_{\zeta \in \Sigma_n \setminus S} \left\{ \log \widetilde{q}_n(\zeta) + \beta_t \alpha(a_n(\zeta); m_t) - \frac{2c_t}{\max\{1, |S|\}} \sum_{\zeta' \in S} \exp\left(-\frac{\|z(\zeta) - z(\zeta')\|^2}{2h^2} \right) \right\}. \tag{5}$$

Here $\widetilde{q}_n$ is the restriction of q_n to the eligible set Σ_n, renormalized to sum to one; $a_n(\zeta)$ is the executable representative of signature ζ; $\alpha(\cdot; m_t)$ is the acquisition function of the GP surrogate trained on all observations so far; β_t weights acquisition against the prior and increases from a small initial value, so selection is prior-dominated (broad exploration) early and acquisition-guided (exploitation) later; and c_t controls the repulsion strength, held fixed in our experiments. The terms favor high proposal probability, high acquisition value, and separation from designs already selected in the batch. Implementation details and schedules are given in Appendix D.5. After evaluating the batch, PWBO updates confidence bounds and removes a signature only when its upper bound is below the best surviving lower bound. Let $L_n(\zeta)$ and $U_n(\zeta)$ denote the resulting lower and upper confidence bounds on $\bar{f}(\zeta)$ after round n, and recall that $\mathcal{R}_n \subseteq \mathcal{R}_0$ is the plausible set surviving elimination after round n. The relevant simultaneous calibration event is

$$\mathcal{E}_{\text{cal}} := \bigcap_{n=1}^{N} \bigcap_{\zeta \in \mathcal{R}_{n-1}} \{L_n(\zeta) \le \bar{f}(\zeta) \le U_n(\zeta)\}, \qquad \Pr(\mathcal{E}_{\text{cal}}) \ge 1 - \delta. \tag{6}$$

On this event the elimination rule preserves the best reachable signature.

Appendix D.5 tabulates PWBO's per-step selection rule alongside standard BO and the prior-augmented baselines used in the experiments: the log-prior term favors candidates supported by the reference distribution, while the repulsion term reduces redundancy within each batch.

The experiments measure the two corresponding behaviors: within-batch coverage and the calibration and survival behavior of confidence-based elimination. Implementation details and complete pseudocode are given in Appendix B.

6 EXPERIMENTS

We organize the experiments around three questions. **Q1** asks whether the reciprocal-log depth–width frontier appears in a controlled synthetic setting. **Q2** tests how prior quality and effective coverage shape the frontier on the Avazu CTR pipeline search space. **Q3** compares PWBO with existing search methods, tests the contribution of GP posterior updating beyond the fixed prior, and validates transfer to NAS-Bench-201.

6.1 EXPERIMENTAL SETUP

We use two complementary environments.

Synthetic benchmark. The objective is the power-cusp family

$$f_k(x) = -W\left(\frac{\|x - x^\star\|_2}{R}\right)^k$$

on a two-dimensional disk of radius R for $k \in \{1, 2, 3\}$. For $k = 1$, the regret gap is proportional to embedding distance, so reducing the covering radius translates one-to-one into regret reduction. For larger k, the same covering-radius reduction yields smaller regret reduction near the optimum, weakening the value–distance coupling that Theorem 1 requires. Three reference priors isolate the role of prior mass: an informed prior centered near $x^\star$, a uniform prior over the disk, and an adversarial prior with the same radial distribution as the informed prior but displaced toward the boundary. The objective and embedding geometry are held fixed, so any frontier difference isolates prior quality. A localized-flat objective removes the value–distance coupling by making all points outside a small optimum neighborhood yield the same value; it serves as a negative control in which the exchange should fail.

CTR benchmark. The real benchmark searches 562 valid CTR pipeline signatures on Avazu (drawn from 576 architecturally distinct configurations after removing degenerate cases), formed by crossing interaction family, embedding dimension, layer count, learning rate, hidden dimension, and dropout. One expensive slot trains one executable realization and returns validation AUC. The headline endpoint is rounds-to-target at $\varepsilon{=}10^{-4}$ relative to the best reachable signature by held-out mean validation AUC; simple regret at a fixed evaluation budget is reported as a secondary metric. Test AUC is not used for search or method selection.

Unless stated otherwise, T is a nominal evaluation cap and the batch width b determines $N_{\max} = \lfloor T/b \rfloor$ complete adaptive rounds and $T_{\text{eff}} = bN_{\max}$ executed evaluations; no incomplete final batch is launched. Reported rounds count completed batches from one, so the first evaluated batch is round 1. We report means $\pm$ s.e.m. over 10 recorded search runs. A non-hit is cap-coded at $N_{\max}$ and is always distinguished through its hit rate. The primary resource views are (i) executed evaluations T_{eff} and (ii) completed adaptive rounds: they distinguish total search work from opportunities to revise selection after feedback. PWBO is compared with random search, standard BO (Jones et al., 1998; Snoek et al., 2012), πBO (Hvarfner et al., 2022), LLAMBO (Liu et al., 2024), and LP-BO (González et al., 2016). The complete experimental protocol, including the candidate space, seeds, hyperparameters, and statistical procedures, is given in Appendix C.

6.2 Q1: DOES THE DEPTH–WIDTH FRONTIER APPEAR IN A CONTROLLED SETTING?

Figure 2 reports the synthetic validation. Panel (a) fixes the objective to the $k = 1$ power-cusp and shows the empirical iso-regret frontier $\widehat{N}(\varepsilon, b)$ for the three reference priors, with reciprocal-log fits overlaid. The fitted scaling

$$N(\varepsilon, b) = c_0 + d_{\text{fit}} \frac{\log(W/\varepsilon)}{\log(b/\nu_{\text{fit}})}$$

has d_{fit} increasing from informed (2.68) through uniform (3.31) to adversarial (3.79), matching the prior-difficulty knob d_{eff} predicted by Theorem 1.

Panel (b) fixes the prior and varies the growth exponent $k \in \{1, 2, 3\}$. Larger k flattens the objective near the optimum and weakens the coupling between embedding distance and regret, so the same

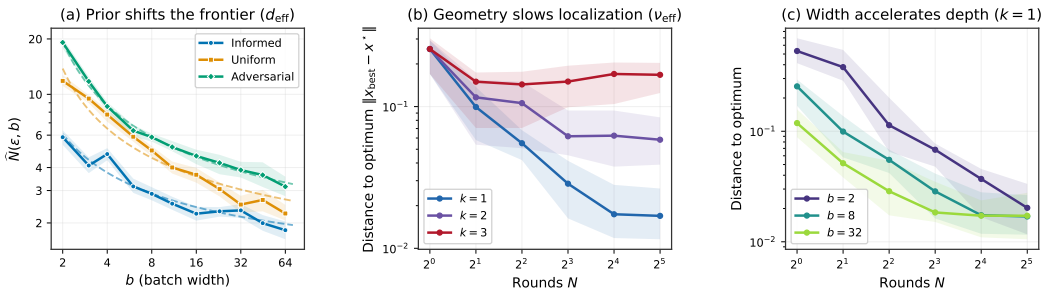

Figure 2: **Synthetic validation of the depth–width exchange.** **(a)** Prior-difficulty knob (d_{eff}): empirical iso-regret frontier $\widehat{N}(\varepsilon, b)$ for the $k = 1$ power-cusp under three reference priors, with reciprocal-log fits overlaid. **(b)** Geometric-overhead knob (ν_{eff}): distance to optimum versus rounds for $k \in \{1, 2, 3\}$ under a fixed prior; larger k weakens value–distance coupling and slows localization. **(c)** Width acceleration: distance to optimum versus rounds for $b \in \{2, 8, 32\}$ at $k = 1$; wider batches reach the same target with fewer rounds.

covering-radius reduction yields smaller value progress. This raises the effective geometric overhead and slows localization, isolating the ν_{eff} knob.

Panel (c) fixes $k = 1$ and varies the per-round batch width $b \in \{2, 8, 32\}$. Wider batches reach the same distance-to-optimum target with fewer rounds, confirming the depth–width exchange for this geometry.

The localized-flat negative control removes the value–distance coupling; performance then depends only on the total budget $T = Nb$ and the reciprocal-log model is rejected in favor of an Nb-only baseline (Appendix D.1, Table 7).

6.3 Q2: What determines whether the exchange works?

We next move to the Avazu CTR pipeline search space and vary the reference prior and within-round coverage while keeping the remaining search procedure fixed.

Intervention 1: prior quality. We keep the objective and search procedure fixed while replacing the reference with three distributions: an informative Qwen reference (Qwen Team, 2026), a uniform reference, and a displaced reference that sorts Qwen probabilities from largest to smallest and assigns them to architectures ordered by increasing mean validation AUC. This preserves the probability multiset and entropy while placing high mass on poor signatures. At b=1, the Qwen prior reaches the target after 6 completed rounds in every recorded run, while the uniform and displaced controls have zero hit rate and are cap-coded at $N_{\max} = 64$. At b=8, Qwen reaches the target after 2 completed rounds in every run, while both controls have zero hit rate and are cap-coded at $N_{\max} = 8$ (Figure 3b). Panel (a) diagnoses near-optimal mass p_ε across tolerances: the Qwen reference has more mass near the optimum at the tighter plotted tolerances, but not at every tolerance. Panel (b) shows its advantage in cap-coded rounds-to-target.

Intervention 2: effective coverage. We keep the prior fixed and degrade within-round coverage. The comparison is between diversity-aware selection (Diverse/PWBO in the figure), geometry-blind selection (No diversity, with the repulsion term removed), and deliberately clustered batches (nearest-neighbor expansion from the highest-prior candidate). At b=1, both diversity-aware and geometry-blind selection reach the target after $\sim$6 completed rounds in every run, while clustered selection does not reach the target within the nominal budget (Figure 3c). At b=8, diversity-aware selection reaches the target after 2 completed rounds in every run, geometry-blind selection reaches it after 3.3 ± 0.7 completed rounds (mean $\pm$ s.e.m., 10 seeds), and clustered selection does not reach the target within the nominal budget. We measure covering radius and the resulting round-wise contraction in regret. Additional width that does not create additional effective locations weakens the round-saving benefit.

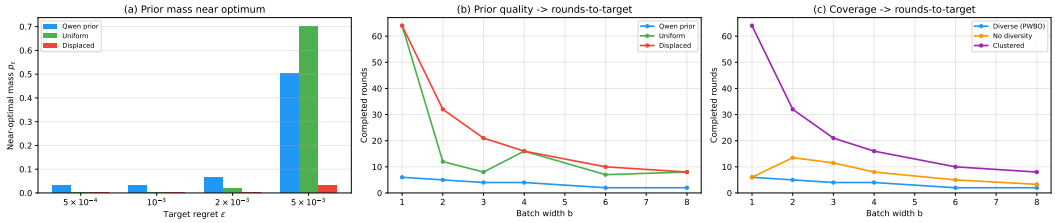

Figure 3: **Intervening on the two bottlenecks.** **(a)** Near-optimal prior mass p_ε for the Qwen, uniform, and displaced priors at multiple target levels. **(b)** Completed rounds under the three priors. **(c)** Completed rounds under diversity-aware (Diverse/PWBO), geometry-blind (No diversity), and clustered batch selection. Runs that do not reach the target within the nominal budget are shown at the corresponding complete-batch limit. For $b = 3$ and $b = 6$, the effective budgets are 63 and 60 evaluations.

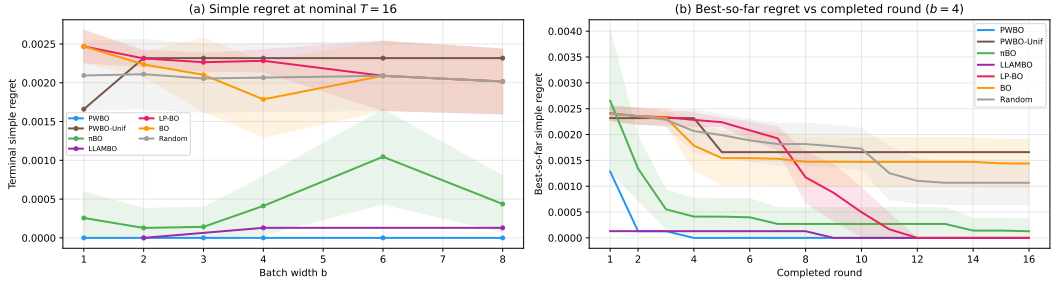

Figure 4: **Algorithmic comparison on Avazu CTR pipeline search.** **(a)** Simple regret under the nominal short cap T=16 across batch widths (10 recorded runs). Complete batches give $T_{\text{eff}} = 15$ for $b = 3$, 12 for $b = 6$, and 16 otherwise. **(b)** Best-so-far regret versus completed round at b=4, with the first batch labeled round 1. Runs stop at first hit; after an exact-optimum hit, only the best-so-far value is extended, not post-hit selections or geometry. Table 2 separately reports the controlled GP-update comparison; full-budget component ablations are in Appendix D.3 (Table 9).

6.4 Q3: Does PWBO turn these principles into a better search rule?

We compare PWBO with random search, BO, πBO, LLAMBO, and LP-BO on Avazu under equal total evaluations and equal adaptive rounds. The short-budget simple-regret view (T=16) complements the primary rounds-to-target comparison.

All methods share the signature embedding $z(\zeta)$. BO, πBO, LP-BO, and PWBO use the same Matérn-5/2 GP and Noisy-EI acquisition. LLAMBO instead uses frozen Qwen for warmstarting, candidate sampling, and in-context surrogate modeling, without fine-tuning. Both πBO and LLAMBO receive the same Qwen-generated reference prior π_θ as PWBO; LP-BO uses a distance-based batch penalty without prior weighting in its selection score. Implementation details are given in Appendix D.5.

At b=8, PWBO reaches the target in 2.0 rounds on average, compared with 3.3 for the strongest competitor, πBO (Table 1). Across the six tested widths, PWBO has the lowest reported rounds-to-target at five widths and ties with πBO at b=3; it also reaches the target in every search seed at every width. The advantage is earlier attainment rather than a uniquely better endpoint: several competitors achieve comparable final regret at the full budget.

Feedback beyond the fixed prior. Table 2 tests whether GP updating improves search beyond a fixed Qwen prior under the short nominal budget T=16. PWBO (controlled) updates the GP after each batch, whereas Frozen GP uses the same prior and batch-scoring rule without conditioning

Table 1: **Method comparison on Avazu.** Rounds are mean cap-coded completed batches to $\varepsilon = 10^{-4}$; non-hits receive $N_{\max} = \lfloor 64/b \rfloor$, so hit rate is reported separately. Regret is terminal best-so-far regret under the nominal $T = 64$ cap. Results use 10 runs per method–width setting

Method	$b = 4$			$b = 8$		
	Hit	Rounds	Regret	Hit	Rounds	Regret
PWBO	1.00	$\mathbf{4.0 \pm 0.0}$	$< 10^{-4}$	1.00	$\mathbf{2.0 \pm 0.0}$	$< 10^{-4}$
πBO	0.90	7.2 ± 1.9	0.0001 ± 0.0001	1.00	3.3 ± 0.7	$< 10^{-4}$
LLAMBO	1.00	9.0 ± 0.0	$< 10^{-4}$	1.00	4.0 ± 0.0	$< 10^{-4}$
LP-BO	1.00	9.6 ± 0.5	$< 10^{-4}$	1.00	5.7 ± 0.2	$< 10^{-4}$
BO	0.00	16.0 ± 0.0	0.0014 ± 0.0002	0.10	7.5 ± 0.5	0.0013 ± 0.0003
Random	0.10	15.5 ± 0.5	0.0011 ± 0.0002	0.20	7.6 ± 0.3	0.0010 ± 0.0003
PWBO-Unif	0.00	16.0 ± 0.0	0.0017 ± 0.0000	0.00	8.0 ± 0.0	0.0017 ± 0.0000

Table 2: **GP updating beyond a fixed prior on Avazu.** Nominal T=16 with 10 seeds. Each cell reports hit rate / median cap-coded completed rounds; non-hits are assigned $N_{\max} = \lfloor 16/b \rfloor$.

T=16**: GP-update controls**				
Variant	$b = 1$	$b = 2$	$b = 4$	$b = 8$
PWBO (controlled)	**1.00/6**	**1.00/5**	**1.00/4**	**1.00/2**
Frozen GP	0.00/16	0.00/8	0.00/4	1.00/2
Prior ranking	0.00/16	0.00/8	0.00/4	0.00/2
Prior sampling	0.20/16	0.20/8	0.50/4	0.30/2
Prior-anchored FPS	0.00/16	0.00/8	0.00/4	0.00/2

the GP on observations. Prior ranking, Prior sampling, and Prior-anchored FPS are non-adaptive controls based on prior order, prior sampling, and geometric coverage.

PWBO reaches the target in every seed at all widths. Frozen GP fails at $b \in \{1, 2, 4\}$ but succeeds at b=8, while the non-adaptive controls are less reliable. Thus GP updating provides additional benefit when the search requires localization across rounds, but is not necessary when a wide batch already reaches the target.

Generalization to structured search spaces. To validate that PWBO's advantages transfer beyond CTR pipeline search, we additionally evaluate on NAS-Bench-201 (Dong & Yang, 2020), a standard neural architecture search benchmark with 15,625 enumerated architectures and precomputed training curves. Using the same Qwen prior and experimental protocol, PWBO achieves the lowest final regret (0.23 vs. 0.44 for LLAMBO) and competitive convergence speed (2.0 rounds vs. 7.5 at b=4), supporting method-level transfer to this structured discrete search space. Full details are given in Appendix D.6.6.

7 Conclusion

We study expensive search when parallel evaluation is available but feedback remains sequential, asking how many feedback rounds are needed to reach a target regret when each round can evaluate multiple candidates. The iso-regret frontier makes this tradeoff explicit: under the stated conditions, the required depth depends on the near-optimal mass supplied by the reference prior and on whether a wider batch produces useful coverage. These conditions motivate PWBO, which combines prior-weighted proposals, diversity-aware batch selection, and confidence-based elimination. Experiments separate the tradeoff from the algorithm that exploits it: controlled interventions vary prior quality and effective coverage, including a setting in which the exchange breaks down, and the CTR study compares PWBO with competing methods and separately tests GP updating under a fixed-prior control. Together, these results connect a favorable prior starting point, non-redundant coverage that makes each batch more useful, and GP feedback updates that improve search when further localization is needed. The scope of the present characterization and natural extensions are discussed in Appendix E.

AI USE STATEMENT

A frozen large language model (Qwen3.5-9B) is itself a component of the method and baselines studied here: it produces the reference prior, proposes candidates, and serves as the LLAMBO surrogate, as described in Section 5.1 and Appendix D.5. Separately, and only for preparing this manuscript, we used AI-based tools for language polishing and stylistic editing. All scientific contributions were performed by the authors, including problem formulation, theoretical analysis, algorithm design, experimental execution, and data interpretation.

REPRODUCIBILITY STATEMENT

Upon acceptance we will release the complete PWBO implementation, all five baselines (BO, πBO, LLAMBO, LP-BO, random search), the RQ1–RQ3 experiment runners, analysis and plotting scripts, and all configuration files. The Avazu dataset is publicly available, and the reference prior uses the frozen `Qwen/Qwen3.5-9B` model without fine-tuning. Appendix C records the realized search space, data split, budget grids, outcome definitions, frontier-estimation procedure, computational requirements, and complete numerical results; Appendix A.2.1 and Appendix A.2.2 give the formal assumptions and proofs.

REFERENCES

Arpit Agarwal, Shivani Agarwal, Sepehr Assadi, and Sanjeev Khanna. Learning with limited rounds of adaptivity: Coin tossing, multi-armed bandits, and ranking from pairwise comparisons. In *Conference on Learning Theory (COLT)*, volume 65, pp. 39–75, 2017.

Noor Awad, Neeratyoy Mallik, and Frank Hutter. DEHB: Evolutionary hyberband for scalable, robust and efficient hyperparameter optimization. In *International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 2147–2153, 2021.

James Bergstra, Rémi Bardenet, Yoshua Bengio, and Balázs Kégl. Algorithms for hyper-parameter optimization. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 24, pp. 2546–2554, 2011.

Jun Shern Chan, Neil Chowdhury, Oliver Jaffe, James Aung, Dane Sherburn, Evan Mays, Giulio Starace, Kevin Liu, Leon Maksin, Tejal Patwardhan, Aleksander Madry, and Lilian Weng. Mlebench: Evaluating machine learning agents on machine learning engineering. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 50466–50494, 2025.

Xuanyi Dong and Yi Yang. NAS-Bench-201: Extending the scope of reproducible neural architecture search. In *International Conference on Learning Representations (ICLR)*, 2020.

Stefan Falkner, Aaron Klein, and Frank Hutter. BOHB: Robust and efficient hyperparameter optimization at scale. In *International Conference on Machine Learning (ICML)*, volume 80, pp. 1437–1446, 2018.

Matthias Feurer, Aaron Klein, Katharina Eggensperger, Jost Tobias Springenberg, Manuel Blum, and Frank Hutter. Efficient and robust automated machine learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 28, pp. 2962–2970, 2015.

Zijun Gao, Yanjun Han, Zhimei Ren, and Zhengqing Zhou. Batched multi-armed bandits problem. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, pp. 503–513, 2019.

Javier González, Zhenwen Dai, Philipp Hennig, and Neil Lawrence. Batch Bayesian optimization via local penalization. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 51, pp. 648–657, 2016.

Siyuan Guo, Cheng Deng, Ying Wen, Hechang Chen, Yi Chang, and Jun Wang. DS-Agent: Automated data science by empowering large language models with case-based reasoning. In *International Conference on Machine Learning (ICML)*, volume 235, pp. 16813–16848, 2024. arXiv:2402.17453.

Frank Hutter, Holger H Hoos, and Kevin Leyton-Brown. Sequential model-based optimization for general algorithm configuration. In *International conference on learning and intelligent optimization*, pp. 507–523. Springer, 2011.

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 (ICLR)*, 2022.

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.

Tarun Kathuria, Amit Deshpande, and Pushmeet Kohli. Batched Gaussian process bandit optimization via determinantal point processes. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 29, pp. 4206–4214, 2016. arXiv:1611.04088.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, pp. 6402–6413, 2017.

Benjamin Letham, Brian Karrer, Guilherme Ottoni, and Eytan Bakshy. Constrained bayesian optimization with noisy experiments. *Bayesian Analysis*, 14(2):495–519, 2019.

Lisha Li, Kevin Jamieson, Giulia DeSalvo, Afshin Rostamizadeh, and Ameet Talwalkar. Hyperband: A novel bandit-based approach to hyperparameter optimization. *Journal of Machine Learning Research*, 18(185):1–52, 2018.

Yibo Li, Zijie Lin, Ailin Deng, Xuan Zhang, Yufei He, Shuo Ji, Tri Cao, and Bryan Hooi. Just-in-time reinforcement learning: Continual learning in LLM agents without gradient updates. In *International Conference on Machine Learning (ICML)*, 2026. arXiv:2601.18510.

Tennison Liu, Nicolás Astorga, Nabeel Seedat, and Mihaela van der Schaar. Large language models to enhance bayesian optimization. In *International Conference on Learning Representations (ICLR)*, 2024. arXiv:2402.03921.

Chris Lu, Samuel Holt, Claudio Fanconi, Alex J. Chan, Jakob Foerster, Mihaela van der Schaar, and Robert Tjarko Lange. Discovering preference optimization algorithms with and for large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, pp. 86528–86573, 2024. arXiv:2406.08414.

Neeratyoy Mallik, Edward Bergman, Carl Hvarfner, Danny Stoll, Maciej Janowski, Marius Lindauer, Luigi Nardi, and Frank Hutter. PriorBand: Practical hyperparameter optimization in the age of deep learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, pp. 7377–7391, 2023.

Tung Nguyen and Aditya Grover. LICO: Large language models for in-context molecular optimization. In *International Conference on Learning Representations (ICLR)*, 2025. arXiv:2406.18851.

Randal S. Olson, Nathan Bartley, Ryan J. Urbanowicz, and Jason H. Moore. Evaluation of a tree-based pipeline optimization tool for automating data science. In *Genetic and Evolutionary Computation Conference (GECCO)*, pp. 485–492, 2016.

Vianney Perchet, Philippe Rigollet, Sylvain Chassang, and Erik Snowberg. Batched bandit problems. *The Annals of Statistics*, 44(2):660–681, 2016.

Qwen Team. Qwen3.5-9B. Hugging Face model repository, 2026. `https://huggingface.co/Qwen/Qwen3.5-9B`.

Bernardino Romera-Paredes, Mohammadamin Barekatain, Alexander Novikov, Matej Balog, M. Pawan Kumar, Emilien Dupont, Francisco J. R. Ruiz, Jordan S. Ellenberg, Pengming Wang, Omar Fawzi, Pushmeet Kohli, and Alhussein Fawzi. Mathematical discoveries from program search with large language models. *Nature*, 625(7995):468–475, 2024.

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.

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

Artur Souza, Luigi Nardi, Leonardo B. Oliveira, Kunle Olukotun, Marius Lindauer, and Frank Hutter. Bayesian optimization with a prior for the optimum. In *European Conference on Machine Learning and Knowledge Discovery in Databases (ECML PKDD)*, volume 12977, pp. 265–296, 2021.

Chris Thornton, Frank Hutter, Holger H. Hoos, and Kevin Leyton-Brown. Auto-WEKA: Combined selection and hyperparameter optimization of classification algorithms. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pp. 847–855, 2013.

Edan Toledo, Karen Hambardzumyan, Martin Josifoski, Rishi Hazra, Nicolas Baldwin, Alexis Audran-Reiss, Michael Kuchnik, Despoina Magka, Minqi Jiang, Alisia Maria Lupidi, Andrei Lupu, Roberta Raileanu, Tatiana Shavrina, Kelvin Niu, Jean-Christophe Gagnon-Audet, Michael Shvartsman, Shagun Sodhani, Alexander H. Miller, Abhishek Charnalia, Derek Dunfield, Carole-Jean Wu, Pontus Stenetorp, Nicola Cancedda, Jakob Nicolaus Foerster, and Yoram Bachrach. AI research agents for machine learning: Search, exploration, and generalization in MLE-bench. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. Spotlight; arXiv:2507.02554.

Patara Trirat, Wonyong Jeong, and Sung Ju Hwang. AutoML-Agent: A multi-agent LLM framework for full-pipeline AutoML. In *International Conference on Machine Learning (ICML)*, volume 267, pp. 60099–60146, 2025. arXiv:2410.02958.

Christopher KI Williams and Carl Edward Rasmussen. *Gaussian processes for machine learning*. MIT press Cambridge, MA, 2006.

Andrew Gordon Wilson, Zhiting Hu, Ruslan Salakhutdinov, and Eric P. Xing. Deep kernel learning. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 51, pp. 370–378, 2016.

Chengrun Yang, Xuezhi Wang, Yifeng Lu, Hanxiao Liu, Quoc V. Le, Denny Zhou, and Xinyun Chen. Large language models as optimizers. In *International Conference on Learning Representations (ICLR)*, 2024. arXiv:2309.03409.

A Formal Theory

A.1 Notation and definitions

The first table contains the reader-facing quantities used in the main text. The second keeps proof and implementation symbols local to the appendix.

Core quantities.

Symbol	Meaning	First use
N, b, T	Adaptive depth, slot width, and total slots $T = Nb$.	§3
m_n, ℓ_n	Distinct signatures and minimum replicates per signature in round n, with $m_n \ell_n \le b$.	§3
ε, δ	Target simple regret and permitted failure probability.	§3
$N^\star$	Policy-class minimum rounds needed at slot width b for the target guarantee.	§3
$\widehat{N}_{\mathrm{PWBO}}$, δ_{emp}	PWBO's observed target-attainment round count and its preregistered confidence criterion.	§6
$\mathcal{R}_0$, $\mathcal{R}_n$	Fixed reachable signature domain and plausible set after round n.	§3
W, w_n	Initial reachable value range and largest regret remaining after round n.	§3
p_ε, d_{eff}	Near-optimal reference mass and its finite-scale exponent.	§4
ν_{eff}	Geometric penalty converting m_n distinct signatures into m_n/ν_{eff} effective cover points.	§4

Technical quantities.

Symbol	Meaning	First use
$a = (\zeta, u)$	Executable candidate, with functional signature ζ and implementation u.	§5.1
π_θ, q_n, H_n	Fixed reference prior, history-conditioned proposal, and proposal history.	§5.1
ρ	Multiplicative coefficient in the per-round slack recursion.	§4
z, Δ, s_{conn}	Signature embedding, its resolution, and the scale used to count components.	App. A.2.1
ν, $D(w)$, $\mathcal{N}(S, r)$	Component bound, largest component diameter, and covering number.	App. A.2.1
L, μ, χ	Upper value slope, sharp-growth slope, and ratio $\chi = L/\mu$.	App. A.2.1
κ_0, C_0, C_1, Λ	Dispersion and covering constants collected into ν_{eff}.	App. A.2.1
$T_{\max}$	Evaluation horizon fixed before search for simultaneous confidence control.	App. A.2.1
s_n, τ_n, τ	Pointwise sampling radius, per-round additive error, and its fixed-allocation value.	App. A.2.1
$\mathcal{C}_n$, Σ_n, $a_n(\zeta)$	Generated candidate pool, eligible signature set, and fixed executable representative.	§5.1
$\mathcal{E}_{\mathrm{cal}}$	Global event on which all confidence intervals are simultaneously valid.	§5.1
β_t, c_t, h	Acquisition weight, repulsion weight, and kernel bandwidth.	§5.1

A.2 Proofs and theoretical extensions

A.2.1 Full setup for the exchange theorem

Fixed-support comparator and counting identity. Let ζ index signatures, let $\bar{f}(\zeta)$ be their expected objective, and fix the nonempty reachable domain $\mathcal{R}_0 := \mathrm{supp}(\pi_\theta)$. Assume $\bar{f}$ attains finite extrema on $\mathcal{R}_0$, choose $\zeta^\star \in \arg\max_{\zeta \in \mathcal{R}_0} \bar{f}(\zeta)$, and define $W := \max_{\mathcal{R}_0} \bar{f} - \min_{\mathcal{R}_0} \bar{f}$. After round n, calibrated elimination leaves a plausible set $\mathcal{R}_n \subseteq \mathcal{R}_0$ containing $\zeta^\star$ on the global event $\mathcal{E}_{\mathrm{cal}}$ in Equation (6). Its slack is

$$w_n := \sup_{\zeta \in \mathcal{R}_n} \big(\bar{f}(\zeta^\star) - \bar{f}(\zeta)\big), \qquad w_0 = W.$$

If, above the floors described below, one round satisfies

$$w_n \le \rho w_{n-1}, \qquad \rho < 1, \tag{7}$$

then $w_N \le \rho^N W$, so any round count satisfying

$$N \ge \left\lceil \frac{\log(W/\varepsilon)}{\log(1/\rho)} \right\rceil \tag{8}$$

is sufficient. This inequality is only accounting; the geometry below is what supplies ρ.

Formal prior-relative dimension. For $w > 0$, define $\mathcal{Z}_w := \{\zeta \in \mathcal{R}_0 : \bar{f}(\zeta) \ge \bar{f}(\zeta^\star) - w\}$ and $p_w := \pi_\theta(\mathcal{Z}_w)$. Let $\varepsilon_{\min} > 0$ be the finest resolved value scale and fix a largest analyzed tolerance $\varepsilon_{\max} < W$.

Definition 3 (Formal prior-relative near-optimality dimension)**.** Over $w \in [\varepsilon_{\min}, \varepsilon_{\max}]$,

$$d_{\text{eff}} := \sup_w \frac{\log(1/p_w)}{\log(W/w)}. \tag{9}$$

Equivalently, d_{eff} is the smallest exponent for which $p_w \ge (w/W)^{d_{\text{eff}}}$ throughout this finite range. The exchange statement assumes $0 < d_{\text{eff}} < \infty$; if $d_{\text{eff}} = 0$, the reference already places unit mass on every analyzed near-optimal set and the dimension-dependent calculation is unnecessary.

Definition 2 is the reader-facing summary of this expression. A separate assumption is required to make the same exponent govern metric covering numbers.

Geometry and assumptions. Represent a signature by $z(\zeta) \in \mathbb{R}^{d_z}$ and let $\mathcal{Z}_\pi := z(\mathcal{R}_0)$. For a discrete representation, let $\Delta := \min_{\zeta \ne \zeta'} \|z(\zeta) - z(\zeta')\|$. Fix a connectivity scale $s_{\text{conn}} \ge \Delta$: two points are linked when their distance is at most s_{conn}, and $\text{Comp}_{s_{\text{conn}}}(S)$ denotes the equivalence classes induced by transitive closure. Fix before search a maximum evaluation horizon $T_{\max}$ and require the realized $T = Nb \le T_{\max}$. The upper bound assumes:

(A1) **Upper Lipschitz.** $|\bar{f}(\zeta) - \bar{f}(\zeta')| \le L\|z(\zeta) - z(\zeta')\|$.

(A2) **Noise and spatial calibration.** Evaluation noise is σ_{alea}-sub-Gaussian. Let S_n be the signatures selected in round n, and let $\widehat{f}_n(\zeta_i)$ average their ℓ_n replicates. With $s_n := \sigma_{\text{alea}}\sqrt{2\log(2T_{\max}/\delta)/\ell_n}$, the proof uses the spatial envelopes

$$L_n(\zeta) := \max_{\zeta_i \in S_n} \{\widehat{f}_n(\zeta_i) - s_n - L\|z(\zeta) - z(\zeta_i)\|\},$$

$$U_n(\zeta) := \min_{\zeta_i \in S_n} \{\widehat{f}_n(\zeta_i) + s_n + L\|z(\zeta) - z(\zeta_i)\|\}.$$

These envelopes satisfy the single global event $\mathcal{E}_{\text{cal}}$ in Equation (6) with probability at least $1 - \delta$. A learned surrogate may replace them only if it certifies the same global validity and no-wider spatial envelopes.

(A3) **Prior mass.** The fixed reference has $0 < d_{\text{eff}} < \infty$ in Definition 3.

(A4) **Attainable dispersion.** If m_n distinct locations are evaluated in round n, their covering radius over the plausible set is at most κ_0 times the smallest radius attainable with m_n centers.

(A5) **Mass–geometry alignment and bounded fragmentation.** For every $w \in [\varepsilon_{\min}, W]$, $z(\mathcal{Z}_w)$ has at most ν s_{conn}-components. With

$$D(w) := \max_{S \in \text{Comp}_{s_{\text{conn}}}(z(\mathcal{Z}_w))} \text{diam}(S),$$

its covering number satisfies $\mathcal{N}(z(\mathcal{Z}_w), r) \le C_0 \nu \max\{1, (D(w)/r)^{d_{\text{eff}}}\}$ for every $r > 0$.

(A6) **Sharp growth.** For every s_{conn}-component S of $z(\mathcal{Z}_W)$, if $\zeta_S^\star$ maximizes $\bar{f}$ among signatures whose embeddings lie in S, then every ζ with $z(\zeta) \in S$ satisfies $\bar{f}(\zeta_S^\star) - \bar{f}(\zeta) \ge \mu\|z(\zeta) - z(\zeta_S^\star)\|$.

Let $C_1 \ge C_0$ absorb both the covering constant and the constant needed when centers must belong to the plausible set. The representation floor is $\varepsilon_{\min} := \mu\Delta < \varepsilon_{\max}$ and $\chi := L/\mu$. Farthest-point traversal gives $\kappa_0 = 2$ (Appendix D.6.5). The connectivity scale trades component count against growth conditioning; it may be chosen to minimize the resulting overhead.

Slots, distinct signature locations, and replicates. The slot width b always counts expensive evaluations, and all b slots are consumed. For the bound, let m_n be the number of distinct signature locations receiving at least ℓ_n replicates each, so $m_n \ell_n \le b$; any remaining slots may be assigned arbitrarily and can only add information. In the no-replication geometric-coverage regime, $m_n = b$ and $\ell_n = 1$. Near the statistical floor, replication reduces mean-estimation uncertainty but also reduces the distinct-design count available for geometric coverage.

Lemma 1 (Per-round contraction with noise)**.** *Under (A1)–(A6), a round using m_n distinct signature locations and ℓ_n replicates per location satisfies, on $\mathcal{E}_{\mathrm{cal}}$,*

$$w_n \le \rho_n w_{n-1} + \tau_n, \qquad \rho_n = \left(\frac{\nu_{\mathrm{eff}}}{m_n}\right)^{1/d_{\mathrm{eff}}}, \qquad \tau_n = \frac{4\sigma_{\mathrm{alea}}\sqrt{2\log(2T_{\max}/\delta)}}{\sqrt{\ell_n}}, \tag{10}$$

where

$$\nu_{\mathrm{eff}} := \nu \Lambda^{d_{\mathrm{eff}}}, \qquad \Lambda := 8\kappa_0 \chi C_1^{1/d_{\mathrm{eff}}},$$

and C_1 absorbs the covering and center-selection constants. Ignoring the additive floor, a genuine contraction requires $m_n > \nu_{\mathrm{eff}}$.

The raw count ν is the number of pieces one batch must cover. The remaining factor Λ collects geometry, imperfect dispersion, and confidence-interval constants. This decomposition motivates the single reader-facing overhead ν_{eff}.

Theorem 2 (Technical upper bound)**.** *Suppose $m_n = m$, $\ell_n = \ell$, and $m\ell \le b$ in every round. Let $\rho = (\nu_{\mathrm{eff}}/m)^{1/d_{\mathrm{eff}}} < 1$ and $\tau = 4\sigma_{\mathrm{alea}}\sqrt{2\log(2T_{\max}/\delta)/\ell}$. If $\max\{\varepsilon_{\min}, \tau/(1-\rho)\} < \varepsilon \le \varepsilon_{\max} < W$, then the conditional recursion in Lemma 1 reaches simple regret at most ε for any*

$$N \ge \left\lceil \frac{d_{\mathrm{eff}}}{\log(m/\nu_{\mathrm{eff}})} \log \frac{W}{\varepsilon - \tau/(1-\rho)} \right\rceil. \tag{11}$$

Equivalently, the minimum sufficient round count for this fixed allocation is at most the right-hand side. For the no-replication geometric-coverage regime $m = b$ and $\ell = 1$, this yields the slot-width scaling in Theorem 1.

A.2.2 Proof of Lemma 1 and Theorem 2

Proof. Work on $\mathcal{E}_{\mathrm{cal}}$ from (A2), so the best reachable signature is never eliminated. Since $\mathcal{R}_{n-1} \subseteq \mathcal{Z}_{w_{n-1}}$, (A5) and (A4) give a covering radius

$$r_n \le \kappa_0 D(w_{n-1}) \left(\frac{C_1 \nu}{m_n}\right)^{1/d_{\mathrm{eff}}} \tag{12}$$

for the m_n distinct signature locations evaluated in round n.

By (A2), every selected-location replicate mean has sampling-error radius $s_n = \sigma_{\mathrm{alea}}\sqrt{2\log(2T_{\max}/\delta)/\ell_n}$ on $\mathcal{E}_{\mathrm{cal}}$, and its spatial envelope adds at most Lr_n at distance r_n. Accounting once for the nearest observation to the optimum and once for the nearest observation to a surviving point, UCB-versus-LCB elimination yields

$$w_n \le 4Lr_n + 4s_n. \tag{13}$$

This is the point at which calibration enters; sub-Gaussian observations alone do not imply that an arbitrary learned surrogate has this spatial envelope.

To relate the current diameter to the previous value slack, consider any s_{conn}-component S of $z(\mathcal{Z}_W)$. Assumption (A6) places every point of $S \cap z(\mathcal{Z}_{w_{n-1}})$ within distance w_{n-1}/μ of its component maximizer. Thus

$$D(w_{n-1}) \le 2w_{n-1}/\mu.$$

Substitution into Equations 12–13 gives

$$w_n \le 8\kappa_0 \chi \left(\frac{C_1 \nu}{m_n}\right)^{1/d_{\mathrm{eff}}} w_{n-1} + \frac{4\sigma_{\mathrm{alea}}\sqrt{2\log(2T_{\max}/\delta)}}{\sqrt{\ell_n}}.$$

With $\Lambda = 8\kappa_0 \chi C_1^{1/d_{\mathrm{eff}}}$ and $\nu_{\mathrm{eff}} = \nu \Lambda^{d_{\mathrm{eff}}}$, this is exactly Lemma 1.

For fixed m and ℓ, unrolling the recursion gives

$$w_N \le \rho^N W + \tau \sum_{j=0}^{N-1} \rho^j \le \rho^N W + \frac{\tau}{1-\rho}. \tag{14}$$

Solving $\rho^N W \le \varepsilon - \tau/(1-\rho)$ proves Equation (11). This also exposes the allocation trade-off: increasing ℓ lowers the additive floor but reduces $m \le \lfloor b/\ell \rfloor$ and hence weakens the geometric contraction. A complete adaptive policy must choose (m_n, ℓ_n) from past data and certify the simultaneous spatial bounds used above; the fixed-allocation result does not itself supply that policy. □

The theorem is conditional on a calibrated spatial confidence bound of the form used in the proof above. Establishing that bound for the chosen surrogate, and choosing adaptive (m_n, ℓ_n) while preserving simultaneous calibration and geometric coverage, remain explicit proof and implementation obligations rather than consequences of sub-Gaussian observation noise alone.

A.2.3 A RUNNING TWO-DIMENSIONAL CASE

This case separates the two quantities in the depth–width law. The reference mass around the optimum determines d_{eff}, whereas covering the current plausible region introduces ν_{eff}. Neither quantity is a property of the objective alone.

Objective and reference mass. Let $\mathcal{R}_0 = \{x : \|x - x^\star\|_2 \le R\}$ and consider

$$f_{\text{cone}}(x) = -W \frac{\|x - x^\star\|_2}{R}. \tag{15}$$

Its w-optimal set is the disk of radius Rw/W. Keep this objective and reachable disk fixed, but use the radial reference family

$$\Pr_{\pi_\alpha}(\|X - x^\star\|_2 \le r) = \Big(\frac{r}{R}\Big)^\alpha, \qquad 0 \le r \le R, \quad \alpha > 0. \tag{16}$$

Then $p_w = (w/W)^\alpha$ and $d_{\text{eff}} = \alpha$. For $\alpha \ge 2$, the same exponent also gives a valid, possibly loose, upper bound on the covering number of a two-dimensional disk. This is the mass–geometry alignment used by the theorem; it does not follow from prior mass alone.

Plausible-region coverage and geometric overhead. Take $z(x) = x$ for the baseline case. For m selected locations S and a current plausible set $\mathcal{R}_{n-1} \subseteq \mathcal{Z}_{w_{n-1}}$, define

$$r_m(S; \mathcal{R}_{n-1}) := \sup_{q \in z(\mathcal{R}_{n-1})} \min_{s \in z(S)} \|q - s\|_2, \qquad r_m^*(\mathcal{R}_{n-1}) := \inf_{|S|=m} r_m(S; \mathcal{R}_{n-1}). \tag{17}$$

Assumption A4 requires an attainable placement with $r_m(S; \mathcal{R}_{n-1}) \le \kappa_0 r_m^*(\mathcal{R}_{n-1})$; it does not require an exact continuous m-center solution. The geometric sweep below retains $d_{\text{eff}} = 2$ but varies admissible plausible-region embeddings and their conditioning; only its first panel is the radial identity-embedding baseline. Expanding Lemma 1 gives

$$\nu_{\text{eff}} = \nu C_1 (8\kappa_0 \chi)^{d_{\text{eff}}}, \qquad \chi = L/\mu.$$

For fixed d_{eff} and common proof constants, define the normalized overhead

$$\widetilde{\nu} := \frac{\nu_{\text{eff}}}{\nu_{\text{eff},0}} = \nu(\kappa_0 \chi)^{d_{\text{eff}}}, \qquad \nu_{\text{eff},0} := C_1 8^{d_{\text{eff}}}. \tag{18}$$

Figure 5 fixes $m = 12$ and compares representative realizations with $d_{\text{eff}} = 2$, $\kappa_0 = 1$, and $\widetilde{\nu} \in \{1, 4, 16\}$. The selected locations are optimized on a dense discretization and validated on a finer grid. They illustrate how fragmentation and conditioning can make the same number of locations cover the current plausible-region embedding less effectively; they do not prove continuous-domain optimality or identify a unique geometry from $\widetilde{\nu}$.

From one-round coverage to resource envelopes. In the noiseless, no-replication regime, $m = b > \nu_{\text{eff}}$. Repeating the one-round contraction gives the continuous sufficient envelopes

$$N_{\text{env}}(\varepsilon, b) := \max\left\{1, \frac{d_{\text{eff}} \log(W/\varepsilon)}{\log(b/\nu_{\text{eff}})}\right\}, \qquad \frac{\varepsilon_{\text{env}}(T, b)}{W} := \Big(\frac{\nu_{\text{eff}}}{b}\Big)^{T/(b d_{\text{eff}})}. \tag{19}$$

The first expression relaxes an integer sufficient round count, and the second fixes b and substitutes $N = T/b$. They visualize the conditional upper-rate mechanism, not observed curves or a matching characterization of minimax complexity.

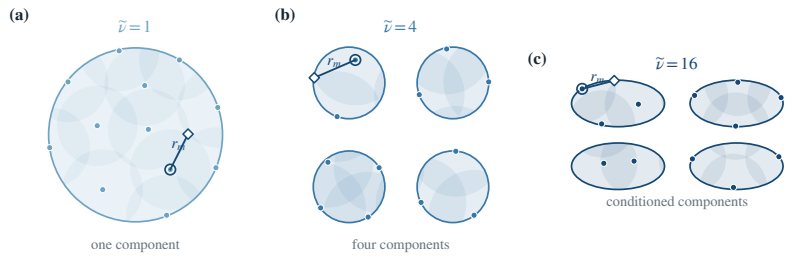

Figure 5: Fixed-width covering in three representative two-dimensional plausible regions. Every panel uses $m = 12$ selected locations. From left to right, the constructions are compatible with $(\nu, \chi) = (1, 1)$, $(4, 1)$, and $(4, 2)$ when $d_{\text{eff}} = 2$, using the ideal-placement normalization $\kappa_0 = 1$ to label $\widetilde{\nu} = 1, 4, 16$. The outer contour is the boundary of the current plausible-region embedding $z(\mathcal{R}_{n-1})$, whose maximum remaining regret is w_{n-1}; it is not a coverage disk. Filled dots are dense-grid optimized approximate m-center locations, not objective maximizers, and the objective optimizer $x^\star$ is not separately marked. The translucent disks have the grid-estimated covering radius r_m; the diamond is a worst-covered validation point $q^\star$, and the circled filled dot is its nearest selected location $s^\star$, so their separation is r_m. The placements are not certified continuous-domain optima.

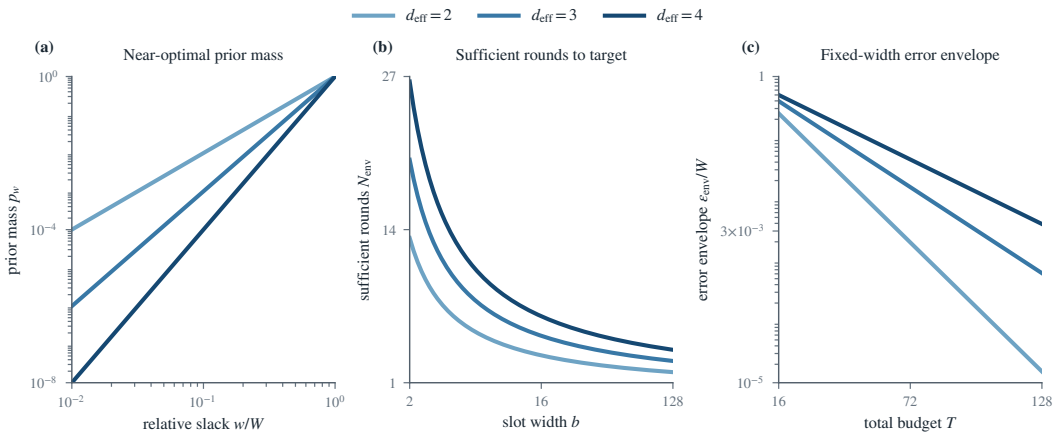

Figure 6: Prior-mass sweep with the objective and normalized geometric overhead fixed. The same color denotes the same d_{eff} in every panel. (a) Near-optimal prior mass. (b) The continuous sufficient-round envelope at $W/\varepsilon = 100$. (c) The fixed-width error envelope at $b = 16$. Common geometric constants are absorbed into the width units.

Changing reference mass. Figure 6 varies $\alpha = d_{\text{eff}} \in \{2, 3, 4\}$ while keeping the objective, reachable disk, and normalized geometric overhead fixed. Panel (a) shows how less near-optimal prior mass raises d_{eff}; panels (b) and (c) then show the corresponding increase in sufficient rounds and slower decay of the fixed-width error envelope.

Changing geometric overhead. The factors inside $\widetilde{\nu}$ have distinct roles. The component bound ν records how many separated pieces one batch must cover. The conditioning $\chi = L/\mu$ converts between objective slack and embedding distance; poor conditioning can make a fixed value resolution occupy a larger or more anisotropic embedding region, but does not prescribe a unique shape. The placement factor κ_0 leaves the plausible region unchanged and measures how closely the selected batch approaches its best m-center radius.

Figure 7(a) visualizes these three mechanisms separately. Panels (b) and (c) then hold $d_{\text{eff}} = 2$ fixed and compare $\widetilde{\nu} \in \{1, 4, 16\}$. Panel (b) plots N_{env} against normalized slot width at $W/\varepsilon = 100$;

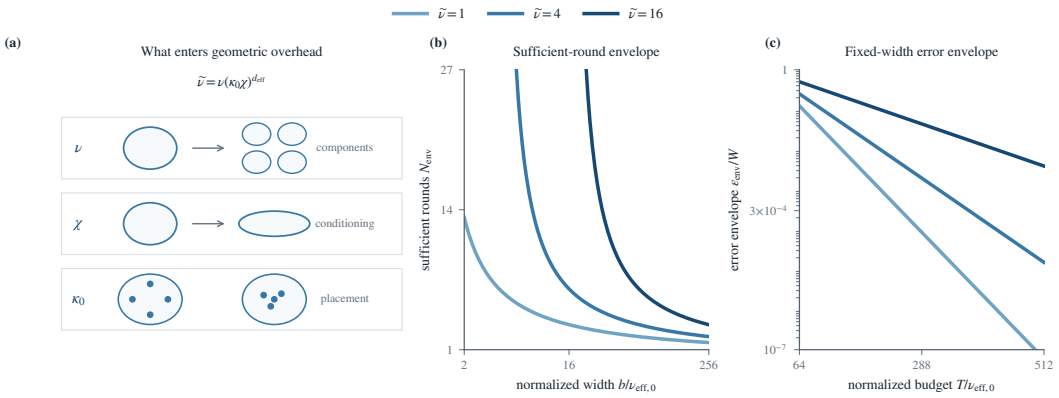

Figure 7: How the ingredients of geometric overhead affect the resource envelopes when $d_{\text{eff}} = 2$. (a) The component bound ν concerns fragmentation, χ is the slack–embedding conditioning, and κ_0 measures placement quality on a fixed region. The icons are schematic: none of these scalars uniquely determines a region shape. (b) The continuous sufficient-round envelope versus normalized slot width at $W/\varepsilon = 100$. (c) The fixed-width error envelope versus normalized total budget at $b/\nu_{\text{eff},0} = 64$. The same color tracks one $\widetilde{\nu}$ across panels (b) and (c); all curves are analytic envelopes.

Table 3: How each ingredient enters the running case. Not every assumption changes d_{eff} or ν_{eff}.

Changed ingredient	Assumption	Effect on d_{eff}	Effect on $\widetilde{\nu}$	Other effect or boundary
Radial prior exponent α	A3, A5	$d_{\text{eff}} = \alpha$	Fixed in the controlled sweep	Requires mass–covering alignment
Component count ν	A5	None	Linear in ν	Bounds separated pieces
Conditioning $\chi = L/\mu$	A1, A6	None	Proportional to $\chi^{d_{\text{eff}}}$	Converts slack and embedding diameter
Placement factor κ_0	A4	None	Proportional to $\kappa_0^{d_{\text{eff}}}$	Changes achieved cover, not region shape
Noise and replication	A2	None	None	Change m and the additive floor
Localized-flat family	A6 fails	Target-scale area only	Not applicable	Main contraction does not apply

each curve begins only when $b/\nu_{\text{eff},0} > \widetilde{\nu}$. Panel (c) fixes $b/\nu_{\text{eff},0} = 64$ and plots the conditional error envelope against normalized total budget.

Noise and replication add a separate floor. The preceding envelopes isolate geometry. On the global calibration event, for fixed m and ℓ with $m\ell \le b$, let $\rho = (\nu_{\text{eff}}/m)^{1/d_{\text{eff}}}$ and let τ be the noise term from Lemma 1. Then

$$w_N \le \rho^N W + \frac{\tau}{1-\rho}. \tag{20}$$

Increasing ℓ lowers τ but may reduce m. It therefore changes both the statistical floor and the distinct-design width available for coverage without changing d_{eff} or ν_{eff}.

Localized-flat boundary. At each target ε, define

$$f^{\text{flat}}_{x^\star}(x) = \max\{0,\, 2\varepsilon - L\|x - x^\star\|_2\}. \tag{21}$$

Its informative support has radius $2\varepsilon/L$. Outside it, every noiseless observation is zero, so the hard family fails A6 and the ν_{eff} contraction is not applicable. Because the instance itself is indexed by the requested target, its quadratic target-scale area is not another fixed-objective evaluation of

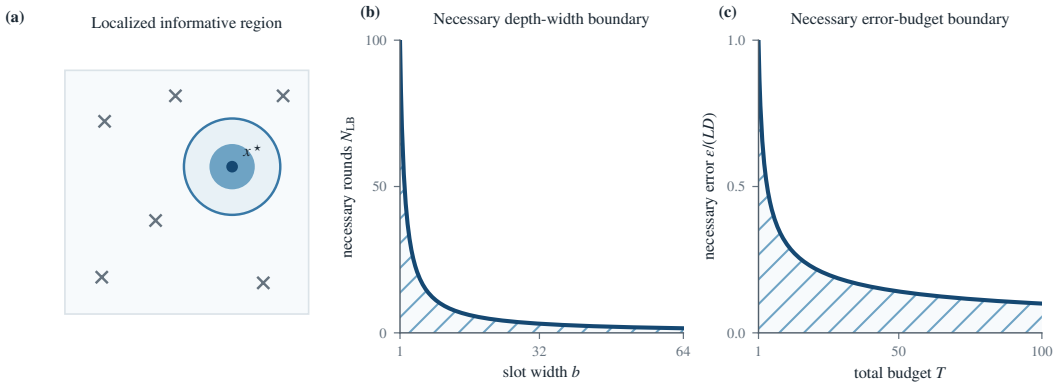

Figure 8: Localized-flat boundary. (a) Queries outside the bump support are indistinguishable. (b) The necessary depth decreases as $1/b$ at a fixed target. (c) The necessary error scale decreases as $T^{-1/2}$. Hatched regions lie below the necessary boundaries and therefore cannot be guaranteed uniformly over this family.

Definition 2. For $0 < \varepsilon \le \bar{c}_2 LD$, Appendix A.2.5 gives

$$N \ge \frac{C_2}{b}\left(\frac{LD}{\varepsilon}\right)^2, \qquad \frac{\varepsilon}{LD} \ge c_2 T^{-1/2}, \quad c_2 = \sqrt{C_2}. \tag{22}$$

Figure 8 shows these necessary boundaries with constants normalized for visualization.

A.2.4 Proposal coverage

Lemma 2 (Prior coverage). *Let $\mathcal{C}_n$ consist of independent draws from q_n, and let $p_\varepsilon = \pi_\theta(\mathcal{Z}_\varepsilon)$. Since $q_n(\mathcal{Z}_\varepsilon) \ge \eta p_\varepsilon$, $|\mathcal{C}_n| \ge (\eta p_\varepsilon)^{-1}\log(1/\delta)$ suffices for $\mathcal{C}_n$ to contain an ε-optimal signature with probability at least $1-\delta$.*

Proof. The probability that none of the $|\mathcal{C}_n|$ independent draws lands in $\mathcal{Z}_\varepsilon$ is at most $(1-\eta p_\varepsilon)^{|\mathcal{C}_n|} \le \exp(-\eta p_\varepsilon |\mathcal{C}_n|)$. Setting this at most δ and solving for $|\mathcal{C}_n|$ gives the claim. □

The bound replaces expensive evaluations by cheap proposal draws. It guarantees a hit in the near-optimal set, not density throughout the plausible region; the latter remains the separate end-to-end coverage requirement (A4).

A.2.5 Flat hard family and lower-bound status

Remark 1 (Tightness status). Equation (15) demonstrates the upper-rate geometry, but exact real-valued distance observations may reveal the unknown center through triangulation. Consequently, the claim that one batch can shrink a conical version space by at most a factor b does not follow from a cell-counting argument. A matching lower bound for noisy real-valued sharp-growth feedback would require an explicit indistinguishable family and an information bound that respects within-batch non-adaptivity. Establishing such a bound remains open.

Proposition 1 (Localized-flat instances can be Nb-limited). *For every fixed dimension d, there are constants $c_d, C_d > 0$ and an L-Lipschitz localized-bump family on a domain of diameter D such that, for $0 < \varepsilon \le c_d LD$, any algorithm that returns an ε-optimal design with probability at least $2/3$ must satisfy $Nb \ge C_d(LD/\varepsilon)^d$. The proof counts at most Nb distinct queried designs; replicates consume slots without testing additional bump locations. Hence no uniform reciprocal-logarithmic slot-width frontier holds over this family. This is a counterexample to a universal depth–width exchange; it does not show that sharp growth is necessary for every objective admitting multiplicative progress.*

Proof of Proposition 1. It is nevertheless possible to prove the contrasting Nb-only lower bound. Fix d and $0 < \varepsilon \le c_d LD$. On a d-dimensional cube of diameter D, choose a packing of $M \ge c'_d(LD/\varepsilon)^d$ centers separated by more than $4\varepsilon/L$. For each center x_j, define

$$f_j(x) = \max\{0, 2\varepsilon - L\|x - x_j\|\}.$$

Each instance is L-Lipschitz, has optimum value 2ε, and has a flat background. An ε-optimal report must lie within distance ε/L of x_j, while a query outside the radius-$2\varepsilon/L$ support returns zero and supplies no information about which unhit center is active.

Place a uniform distribution over the M instances. Until the support of the active bump is hit, every adaptive transcript is identical, so each distinct queried location can intersect the support of at most one packed instance. After at most Nb distinct locations, the algorithm has directly tested at most Nb centers; its final report can cover at most one additional center. Its average success probability is therefore at most $(Nb + 1)/M$. By Yao's principle, achieving success probability at least $2/3$ on every instance requires

$$Nb \ge C_d \left(\frac{LD}{\varepsilon}\right)^d.$$

Here $C_d > 0$ depends only on the fixed dimension. Replicates consume slots without testing new centers and cannot improve this bound. The construction proves that Nb-only hard instances are possible without sharp growth; it does not make sharp growth necessary for every objective admitting a logarithmic exchange. □

A.2.6 Boundary cases of the exchange frontier

Corollary 1 (Cold-start candidate coverage)**.** *At N=1, draw a candidate pool $\mathcal{C}_1$ from $q_1 = \pi_\theta$, deduplicate it to Σ_1, and suppose $|\Sigma_1| \ge b$. Select $S \subseteq \Sigma_1$ with $|S| = b$ and define $r_{\Sigma_1}(S) := \max_{\zeta\in\Sigma_1} \min_{\zeta'\in S}\|z(\zeta) - z(\zeta')\|$. If $|\mathcal{C}_1| \ge p_\varepsilon^{-1}\log(1/\delta)$, then with probability at least $1 - \delta$ the selected set contains a signature with regret at most $\varepsilon + Lr_{\Sigma_1}(S)$. When one-round dispersion gives $r_{\Sigma_1}(S) = O(\mathrm{diam}(\mathcal{Z}_\pi)b^{-1/d_{\mathrm{eff}}})$, the geometric coverage term mirrors the $b^{-1/d_{\mathrm{eff}}}$ one-round scaling. This inclusion result does not guarantee that a noisy one-round policy identifies and reports that signature.*

Corollary 2 (Reference sampling in the slack regime)**.** *Ignoring observations, T independent draws from π_θ contain an ε-optimal reachable signature with probability at least $1 - \delta$ whenever $T \ge p_\varepsilon^{-1}\log(1/\delta)$.*

Both corollaries are candidate-inclusion statements, not guarantees that a noisy policy identifies the best sampled signature. The first isolates the one-round finite-pool covering term. The second marks a slack regime in which non-adaptive inclusion is already possible; below that budget, adaptive elimination may matter. Appendix A.2.8 gives the short argument, and Appendix D.6.3 discusses how the fixed reference changes the finite-scale mass exponent.

A.2.7 Proof of Corollary 1

At N=1 the batch is committed before any observation exists, so there is no elimination step. Because $q_1 = \pi_\theta$, Lemma 2 specializes to: the pool contains a candidate whose signature ζ_ε is ε-optimal with probability at least $1 - \delta$ if $|\mathcal{C}_1| \ge p_\varepsilon^{-1}\log(1/\delta)$. Deduplication retains ζ_ε in Σ_1. By the definition of $r_{\Sigma_1}(S)$, some selected $\zeta_s \in S$ satisfies $\|z(\zeta_s) - z(\zeta_\varepsilon)\| \le r_{\Sigma_1}(S)$. Assumption (A1) then gives $\bar{f}(\zeta_s) \ge \bar{f}(\zeta^\star) - \varepsilon - Lr_{\Sigma_1}(S)$. Substituting the one-round dispersion scaling stated in Corollary 1 recovers its $b^{-1/d_{\mathrm{eff}}}$ form. □

A.2.8 Reference coverage in the slack regime

Draw the whole budget $T = Nb$ independently from π_θ and ignore observations while forming the pool. The probability that no draw is ε-optimal is $(1 - p_\varepsilon)^T \le \exp(-Tp_\varepsilon)$. Hence $T \ge p_\varepsilon^{-1}\log(1/\delta)$ proves Corollary 2.

This argument concerns inclusion only. Under noisy evaluation, identifying which member of the pool to report requires a separate allocation or confidence guarantee. Thus the threshold locates a regime in which adaptive proposal generation is unnecessary for coverage.

A.2.9 KL REGULARIZATION, DISTRIBUTION-LEVEL OBJECTIVE, AND FIXED POINT

For a generic reference distribution π_0 and bounded utility u, the KL-regularized objective

$$\mathbb{E}_{a\sim\pi}[u(a)] - \frac{1}{\beta} D_{\mathrm{KL}}(\pi\|\pi_0)$$

is maximized by the Gibbs distribution

$$\pi(a) \propto \pi_0(a)\exp(\beta u(a)).$$

Larger β places more weight on utility and permits a greater departure from the reference distribution (Li et al., 2026). PWBO instantiates this template on each finite eligible set with reference $\widetilde{q}_n$, acquisition utility, and an additional diversity term.

For a distribution π_n on the finite eligible signature set Σ_n, define

$$\mathrm{Div}(\pi_n) := -\mathbb{E}_{\zeta,\zeta'\sim\pi_n}[k_h(z(\zeta), z(\zeta'))], \qquad k_h(z, z') = \exp(-\|z - z'\|^2/2h^2), \tag{23}$$

and the KL-regularized objective

$$J(\pi_n) = \mathbb{E}_{\zeta\sim\pi_n}[\alpha(a_n(\zeta); m_t)] + \gamma\mathrm{Div}(\pi_n) - \frac{1}{\beta} D_{\mathrm{KL}}(\pi_n\|\widetilde{q}_n). \tag{24}$$

Without diversity, the standard Gibbs solution is

$$\pi_n(\zeta) \propto \widetilde{q}_n(\zeta)\exp(\beta\alpha(a_n(\zeta); m_t)). \tag{25}$$

Its corresponding per-signature ranking cannot enforce batch coverage; Appendix A.2.10 states the obstruction.

Expanding Equation (24) over Σ_n, the quadratic term contributes $-2\gamma\sum_{\zeta'}\pi_n(\zeta')k_h(z(\zeta), z(\zeta'))$ to the derivative. Stationarity with a normalization multiplier gives

$$\pi_n(\zeta) \propto \widetilde{q}_n(\zeta)\exp\Big(\beta\big[\alpha(a_n(\zeta); m_t) - 2\gamma\mathbb{E}_{\zeta'\sim\pi_n}k_h(z(\zeta), z(\zeta'))\big]\Big). \tag{26}$$

For fixed $\widetilde{q}_n$, the normalized right-hand side is a contraction in total variation for sufficiently small $c = \beta\gamma$; boundedness of k_h supplies a finite Lipschitz constant for this map. Equation (5) replaces the expectation under the unknown fixed point by the empirical distribution of signatures already selected in the batch and permits round-dependent weights β_t and c_t.

A.2.10 WHY GEOMETRY-BLIND RANKING DOES NOT ENSURE COVERAGE

Proposition 2 (Unary utilities alone do not imply coverage)**.** *Consider a top-b rule whose input is only one scalar utility $g(\zeta)$ per signature and does not use signature locations or selected-set-dependent terms. Uniformly over embeddings of the same scored signatures, this rule cannot guarantee a covering radius that decreases with b.*

The rule selects the same signatures whenever the scalar utilities and tie-breaking order are the same. Embed those selected signatures in an arbitrarily small neighborhood and place an unselected signature near the opposite edge of the domain. The score input is unchanged, but the selected set has covering radius arbitrarily close to the domain diameter. A unary score may of course be engineered from geometry in advance; the proposition only says that proposal or acquisition utilities by themselves do not imply coverage. The selected-set term in Equation (5) exposes the geometry directly. □

B PWBO DETAILS

B.1 IMPLEMENTATION DETAILS

Representation requirements. For the candidate representation introduced in Section 5.1, three conditions support the theory-to-implementation interface:

- (R1) z is invariant under functional equivalence, with no cosmetic directions.

- (R2) $\bar{f}(\zeta)$ is L-Lipschitz in z, so the surrogate can interpolate.
- (R3) $\log q_n(\zeta)$ is commensurable across signatures in one pool and measures proposal plausibility rather than coding idiom; the representative rule $a_n(\zeta)$ is fixed before expensive evaluation and satisfies $\mathbb{E}[y \mid \zeta, a_n(\zeta)] = \bar{f}(\zeta)$.

A fixed-schema DSL satisfies these conditions by construction.

Schema granularity. Recording more decisions in ζ can reduce variation among implementations of one signature, but it can also change d_{eff}, ν_{eff}, and the representation floor by enlarging the represented space. For a fixed allocation, the relevant bound has the form $w_N \lesssim W(\nu_{\text{eff}}/m)^{N/d_{\text{eff}}} + \tau/(1-\rho)$ above that floor, with $\rho = (\nu_{\text{eff}}/m)^{1/d_{\text{eff}}}$. Finer schemas are therefore not uniformly better.

Cold-start space filling. When $n = 1$, $q_1 = \pi_\theta$ and acquisition is flat. For an optional operational threshold τ_{prior}, define the filtered signature set $\Sigma_{1,\tau_{\text{prior}}} := \{\zeta \in \Sigma_1 : \log \pi_\theta(\zeta) \geq \tau_{\text{prior}}\}$. The prior-guided k-center design is

$$S^\star = \arg \min_{S \subseteq \Sigma_{1,\tau_{\text{prior}}},\, |S|=b} \max_{\zeta \in \Sigma_{1,\tau_{\text{prior}}}} \min_{\zeta' \in S} \|z(\zeta) - z(\zeta')\|. \tag{27}$$

The threshold filters only this finite first-round signature set and does not redefine $\mathcal{R}_0$. Corollary 1 uses $\tau_{\text{prior}} = -\infty$. For finite τ_{prior}, PWBO requires at least b eligible signatures after filtering, executes their fixed representatives $a_1(\zeta)$, and renormalizes $\widetilde{q}_1$ on that set; the coverage mass becomes $p_{\varepsilon,\tau_{\text{prior}}} := \pi_\theta(\mathcal{Z}_\varepsilon \cap \{\zeta : \log \pi_\theta(\zeta) \geq \tau_{\text{prior}}\})$.

Annealing. PWBO uses $\beta_t = \beta_0(1 - e^{-(t-b)/\tau_\beta})$ for $t > b$, which continuously moves from prior-dominated to acquisition-guided selection. The repulsion coefficient c_t is held fixed in the experiments reported here.

Width allocation. Near the noise floor, PWBO may evaluate m_n distinct signatures and use the remaining slots for replicates, always respecting $m_n \ell_n \leq b$. The fixed-allocation bound in Appendix A.2.1 exposes the trade-off; a data-dependent allocation must also preserve the global calibration event and is not claimed by the current theorem.

Bayesian-optimization components. Given observations through evaluation t, a surrogate m_t assigns each candidate a predictive mean and uncertainty, and an acquisition function $\alpha(\cdot; m_t)$ ranks candidates for evaluation (Jones et al., 1998; Snoek et al., 2012; Shahriari et al., 2016). Common surrogate families include Gaussian processes (Williams & Rasmussen, 2006), tree-structured estimators (Bergstra et al., 2011; Hutter et al., 2011), and deep kernels (Wilson et al., 2016). PWBO models absolute AUC and separates epistemic from aleatoric uncertainty. It uses a sparse GP with a Matérn kernel while t is small, a deep ensemble (Lakshminarayanan et al., 2017) when the kernel assumption is doubtful, and optionally a learned kernel with more data. Noise-aware acquisition such as Noisy-EI or qNEI (Letham et al., 2019) avoids chasing seed variance, and the surrogate is updated after each round.

Proposal and evaluation implementation. The LLM parameters remain frozen during search. Each round draws its core proposals independently from q_n: each draw uses the untempered reference π_θ with probability η and $\pi_\theta(\cdot \mid H_n)$ otherwise. Optional conditioned draws at multiple temperatures may augment this guaranteed pool but do not enter the coverage bound. Selection uses $\widetilde{q}_n$ normalized on Σ_n and evaluates each selected signature through $a_n(\zeta)$.

Calibrating β_0 and c_t. β_0 is set so $\mathrm{Var}[\beta_0 \alpha(a_n(\zeta))] \approx \mathrm{Var}[\log \widetilde{q}_n(\zeta)]$ over $\zeta \in \Sigma_n$. The value of c_t is set so the empirical repulsion term has comparable spread; this tuning heuristic is distinct from the fixed-point condition on $c = \beta\gamma$ in Appendix A.2.9.

Evidence store. Append-only (a_i, x_i, y_i) with vector index. The fixed reference proposer π_θ is frozen during each search; q_n changes only through the updated context H_n. Any model consolidation occurs in a separate low-frequency outer loop.

Multi-temperature sampling. The core proposals are independent draws from q_n, with the reference branch sampled at temperature 1 to preserve $q_n(A) \geq \eta\pi_\theta(A)$. Optional extra draws split the conditioned branch across temperatures 0.7, 1.0, and 1.5 to reduce concentration; they augment the pool but are not counted in Lemma 2, and eligible candidates are scored under $\widetilde{q}_n$ formed from the untempered q_n.

B.2 Complete algorithm

Algorithm 1 Prior-Weighted Bayesian Optimization (PWBO)

Require: Reference prior π_θ, batch size b, rounds N, GP surrogate
Ensure: Best observed design across all completed rounds

Initialize plausible set $\mathcal{R}_0 \leftarrow$ all eligible signatures
Initialize GP surrogate m_0 with empty history
for round $n = 1, 2, \ldots, N$ **do**
  Generate candidate pool $\mathcal{C}_n$ from π_θ (or history-conditioned prior for $n > 1$)
  Form eligible set $\Sigma_n \leftarrow \{\zeta(a) : a \in \mathcal{C}_n\} \cap \mathcal{R}_{n-1}$
  Renormalize prior $\widetilde{q}_n \leftarrow q_n$ restricted to Σ_n
  Initialize selected set $S \leftarrow \emptyset$
  for $j = 1, 2, \ldots, m_n$ **do** ▷ Greedy distinct-design selection
    $\zeta_{\text{next}} \leftarrow \arg\max_{\zeta \in \Sigma_n \setminus S} \Big\{ \log \widetilde{q}_n(\zeta) + \beta_t\, \alpha(a_n(\zeta); m_t) - \frac{2c_t}{\max\{1, |S|\}} \sum_{\zeta' \in S} \exp\big(- \frac{\|z(\zeta) - z(\zeta')\|^2}{2h^2} \big) \Big\}$
    $S \leftarrow S \cup \{\zeta_{\text{next}}\}$
  end for
  Assign remaining $b - m_n$ slots via replication policy
  Execute all b slots, observe outcomes y
  Update GP surrogate m_n with new observations
  Compute confidence bounds $L_n(\zeta), U_n(\zeta)$ for all $\zeta \in \mathcal{R}_{n-1}$
  Update plausible set: $\mathcal{R}_n \leftarrow \{\zeta \in \mathcal{R}_{n-1} : U_n(\zeta) \geq \max_{\zeta' \in \mathcal{R}_{n-1}} L_n(\zeta')\}$
end for
return the evaluated design with the largest observed outcome

C Experimental Protocol

This appendix fixes the reporting contract for Section 6. It records the protocol facts, the realized candidate space, and the complete RQ1–RQ3 results on the Avazu benchmark.

C.1 Evidence status and provenance

Table 4 records which artifacts currently exist. The Avazu experiment provides complete Q1–Q3 results and provides evidence consistent with the exchange law and with the roles of prior quality and effective within-round coverage, under the conditions stated in Section 4.

C.2 Avazu data and realized candidate space

The experiments use the first 1,000,000 Avazu examples split chronologically into 800,000 training, 100,000 validation, and 100,000 test examples. Search decisions use validation AUC only; the test split is not used for search or method selection. The candidate space comprises 576 architectures formed by crossing 4 interaction families, 2 embedding dimensions, 2 layer counts, 4 learning rates, 3 hidden dimensions, and 3 dropout levels, of which 562 remain valid after removing degenerate cases. Table 5 reports the realized architecture space.

C.3 Search protocol, budgets, and outcomes

Table 6 summarizes the Avazu experiment protocol. For nominal cap T and batch width b, each run permits $N_{\max} = \lfloor T/b \rfloor$ complete adaptive rounds and executes at most $T_{\text{eff}} = bN_{\max}$ evalua-

Table 4: Status of the empirical evidence used by the revised experimental narrative.

Artifact	Available evidence	Role in final paper
Avazu 1M experiment	562 valid pipeline signatures; up to 8 training seeds; 4,246 checkpoints	Complete Q1–Q3 results
`Qwen/Qwen3.5-9B` prior	Frozen BF16; 304 pilot and 6,000 confirm slots; 8 indices per slot	Q2 prior specification
Q3 method comparison	Six methods; $b \in \{1, 2, 3, 4, 6, 8\}$; T=64, 10 search seeds	Q3 performance comparison
Mechanism studies	T=64 component ablations; T=16 controlled GP-update and static-prior comparisons	Component and feedback evidence

Table 5: Realized architecture space for the Avazu experiment. Counts describe the actual candidate pool.

Field	Realized values	Candidate counts / note
Interaction family	{DNN, Wide&Deep, DCN, AutoInt}	4 values
Embedding dimension	{8, 16}	2 values
Num layers	{1, 2}	2 values
Learning rate	$\{10^{-4}, 5 \times 10^{-4}, 10^{-3}, 5 \times 10^{-3}\}$	4 values
Hidden dimension	{64, 128, 256}	3 values
Dropout	{0.0, 0.1, 0.3}	3 values
Total architectures	:	576

tions; no incomplete final batch is launched. All methods within a method–width cell use the same effective cap.

Table 6: Avazu experiment protocol.

Item	Protocol
Dataset	Avazu first 1M; chronological 800K/100K/100K split
Architectures	562 valid signatures, drawn from a 576-point grid (4 families $\times$ 2 emb $\times$ 2 layers $\times$ 4 lr $\times$ 3 hidden $\times$ 3 dropout)
Training seeds	Up to 8 per architecture (512 with 8 seeds, 50 with 3); 4,246 checkpoints total
Search seeds	10 independent seeds per method–width cell
Nominal budget	T=64 for the method comparison; $T \in \{16, 32, 64\}$ for the separation study
Complete-batch budget	$N_{\max} = \lfloor T/b \rfloor$ and $T_{\text{eff}} = bN_{\max}$; no incomplete final batch
Batch width b	$\{1, 2, 3, 4, 6, 8\}$ (method comparison); $\{1, 2, 4, 8\}$ (separation study)
Primary outcome	Validation AUC over the 562 valid signatures; test AUC not used for search
Endpoint	First completed batch reaching ε=10^{-4}; non-hits are assigned $N_{\max}$ and accompanied by hit rate
Compute	NVIDIA L20 GPUs; $\approx$265 GPU-hours training the 4,246 checkpoints plus CPU time for search loops and analysis

Outcome definition. For each architecture ζ, the recorded outcome is the validation AUC of its trained checkpoint; $\widehat{\bar{f}}(\zeta)$ denotes the mean over the available training seeds. The pool optimum $\widehat{f}^\star$ is the largest mean validation AUC across all 562 valid signatures. We report the nonnegative simple regret $\widehat{f}^\star - \widehat{\bar{f}}(\widehat{\zeta})$ where $\widehat{\zeta}$ is the recommendation of a given method. Test AUC is not used for search or method selection.

Endpoint and cap coding. The primary endpoint is rounds-to-target: the first completed adaptive batch at which the cumulative-best validation AUC comes within $\varepsilon{=}10^{-4}$ of $\widehat{f}^{\star}$. Completed batches are numbered from one, so the first evaluated batch is round 1. A run that misses the target is assigned $N_{\max} = \lfloor T/b \rfloor$; these are cap-coded summaries rather than survival-analysis estimates, so hit rate is reported separately whenever non-hits occur. All methods are summarized over 10 independent search seeds. At the nominal $T{=}64$, widths $b = 3$ and $b = 6$ have $T_{\text{eff}} = 63$ and 60, respectively; at nominal $T{=}16$, they have $T_{\text{eff}} = 15$ and 12. At $b{=}8$, BO and random search have hit rates of 10–20%; their reported mean rounds therefore include cap-coded non-hits.

Short-budget trajectory view. The nominal-$T{=}16$ view in Figure 4(a) truncates each full-budget $T{=}64$ trajectory after $N_{\max} = \lfloor 16/b \rfloor$ completed batches. Avazu search stops administratively at the first hit. If a trajectory ends before the short-budget cap at the exact pool optimum, only its terminal best-so-far value is carried forward; no post-hit selections, feedback, or geometry are reconstructed. In this view, PWBO reaches the target at every tested width, while πBO has 30–50% hit rate for $b \geq 4$ and LLAMBO misses at $b \in \{4, 8\}$. This method-comparison view is distinct from the controlled GP-update study in Table 2.

Per-training timing. A single pipeline training in the CTR benchmark takes a measured median $t_{\text{eval}} \approx 7$ minutes. This per-training measurement is reported separately from the search-round results and is not a measured end-to-end search latency. We do not convert capped rounds for non-hits into time to target.

Resource interpretation. Evaluation count measures aggregate work, whereas sequential batch completions create feedback waits. With homogeneous evaluation costs and ideal parallel execution, fewer completed batches to the same target imply less waiting; this accounting includes initialization. Our results compare search in evaluations and rounds, not measured end-to-end latency. Scheduling and proposal-generation overheads are outside this idealized interpretation.

D Additional Results

D.1 RQ1: synthetic frontier details

Figure 2 in the main text reports the synthetic frontier, the power-cusp k-family convergence curves, and the width-acceleration panel. Table 7 reports the reciprocal-log-vs-Nb model comparison across all synthetic environments, giving the corresponding per-prior frontier fits and held-out ΔNLL for the $k = 1$ objective and the localized-plateau negative control. The $k = 1$ objective and the plateau both live on a two-dimensional disk of radius $R = 1$ ($k = 1$: $W = 0.5$; plateau: $h = 0.1$, $r_0 = 0.2$, $L = 1$). The k-family sweep uses $k \in \{1, 2, 3\}$ with the same uniform prior and the same geometric batch grid $b \in \{2, 3, 4, 6, 8, 11, 16, 23, 32, 45, 64\}$ (11 values), with 30 seeds per cell. The main-text frontier reports the interpolated $\widehat{N}(\varepsilon, b)$ at a shared accuracy target with seed-bootstrap bands; the dense b grid is what makes the reciprocal-log shape and its prior ordering visually resolvable.

Table 7: RQ1 synthetic frontier fits. Negative ΔNLL favors reciprocal-log scaling; positive values favor the Nb-only model. Fits use the (N, b) grid and leave-one-b-out cross-validation.

Objective	Prior	d_{fit}	ν_{fit}	ΔNLL
$k = 1$ (power-cusp)	informed	2.68	0.48	-0.07
$k = 1$ (power-cusp)	uniform	3.31	0.88	-0.66
$k = 1$ (power-cusp)	adversarial	3.79	0.97	-1.57
Localized plateau	informed	0.04	1.79	$+9.83$
Localized plateau	uniform	0.64	1.74	$+5.92$
Localized plateau	adversarial	1.34	1.73	$+6.95$

D.2 RQ2: PRIOR SPECIFICATION AND SENSITIVITY

The reference Qwen prior is defined before feedback is observed. The frozen `Qwen/Qwen3.5-9B` revision runs in BF16 with thinking disabled and generates architecture configurations using a fixed prompt and structured parser. The parser accepts exactly 8 integer indices from the registered set of 562 valid candidate signatures. A pilot phase precedes an independent confirm run. The confirm frequencies $\widehat{q}$ define $\pi_Q = 0.99\widehat{q} + 0.01U$ exactly.

The uniform control is $U(a) = 1/562$. The displaced control sorts Qwen probabilities from largest to smallest and assigns them to architectures ordered by increasing mean validation AUC. Thus it preserves support, the probability multiset, and entropy without using held-out outcomes.

Table 8: Prior controls and sensitivity analyses. The effective dimension d_{eff} measures concentration near the optimum; lower is better. The d_{eff} values here are phenomenological shape parameters from the reciprocal-log model fit; see Figure 3 for the main-text prior intervention.

Prior	Support size	d_{eff}	Regret at 64 eval	Status
`Qwen/Qwen3.5-9B`	562	0.5784	0.0000	Passed
Uniform	562	0.9966	0.0004	Passed
Displaced	562	1.0749	0.0017	Passed

D.3 RQ3: MECHANISM AND ABLATION DETAILS

The three studies below serve different purposes. Table 9 tests components of full PWBO on the 562-signature benchmark; Table 2 in the main text isolates GP updating in a controlled configuration on the same benchmark; Table 10 reports exploratory diagnostics under a separate 60-candidate protocol. The short-budget study changes the control configuration as well as the budget and is not simply a truncation of the full-budget component ablation.

Short-budget GP-update controls. PWBO (controlled) and Frozen GP use the fixed Qwen prior (η=1), with no history-conditioned LLM proposals. They share initialization and the batch-scoring rule; elimination only records diagnostics and never prunes the selection pool, and the annealing schedule depends only on the round index, not on recent improvements. Frozen GP leaves the surrogate at its unconditioned prior by disabling every observation-conditioned fit or refit, including during warm-up. Observed AUCs still update the EI incumbent, elimination diagnostics, best-observed recommendation, and first-hit record. With the unconditioned GP, predictive means and variances are identical across candidates, so the incumbent-dependent EI term adds a common offset without changing their ranking. Execution stops administratively after a hit; this stopping rule is shared by both controls and is not part of the Frozen-GP intervention. The comparison isolates GP posterior updating rather than prohibiting all uses of performance feedback; it does not test history-conditioned LLM proposals or active elimination.

Full-budget component ablations. Table 9 reports component ablations on the full 562-signature Avazu benchmark. At the full budget (T=64, b=8), every variant reaches the target, but only full PWBO attains it in two rounds: removing either repulsion or elimination delays attainment to about three rounds (3.0–3.3), and removing both yields a comparable delay. These ablations complement the Q2 coverage intervention by also testing elimination and the removal of both components.

Exploratory mechanism diagnostics. Coverage is evaluated in the same normalized embedding used by the selection rule. For each round we record the maximum distance from a plausible architecture to the selected batch, the median pairwise distance, and the fraction of duplicate architectures. Calibration is evaluated before elimination by nominal-versus-empirical interval coverage. Safe elimination is measured by optimum survival and by the false-elimination fraction. Contraction uses the same plausible set and held-out objective to compute w_n/w_{n-1}. Figure 3 in the main text organizes the prior and coverage interventions. An exploratory mechanism scan on a 60-candidate prior-scan protocol (geometric coverage, calibration and optimum survival, and plausible-set con-

Table 9: **Component ablations on Avazu: full budget** (10 seeds). T=64, b=8; mean rounds $\pm$ s.e.m. and final regret. Every variant reaches the target.

T=64, b=8**: component ablations**		
Variant	Rounds	Final regret
Full PWBO	2.0 ± 0.0	$< 10^{-4}$
No repulsion	3.3 ± 0.7	$< 10^{-4}$
No elimination	3.0 ± 0.0	$< 10^{-4}$
Neither (both off)	3.3 ± 0.7	$< 10^{-4}$

traction) is reported in Table 10; the main-text Q3 evidence, including the feedback-separation study, uses the full 562-signature benchmark above.

Table 10: Exploratory mechanism ablations on the 60-candidate prior-scan protocol, which differs from the main Q3 study above. Each row changes one theorem-facing component while holding the proposal pool, budget, and evaluation seeds fixed. Optimum survival and false elimination are computed from the calibrated bounds *before* any elimination, so for the acquisition-only variant they report the counterfactual quality of elimination decisions rather than an active pruning outcome.

Variant	Changed interface	Simult. coverage	Marginal coverage	Optimum survival	False elimination
Full PWBO	None	0.875	0.877	1.000	0.000
No repulsion	Dispersion (A4)	0.844	0.859	0.984	0.016
Acquisition only	No elimination	0.875	0.877	0.969	0.031

D.4 Embedding smoothness validation

Assumption (A1) requires the objective to be L-Lipschitz in the signature embedding: $|\bar{f}(\zeta) - \bar{f}(\zeta')| \leq L\|z(\zeta) - z(\zeta')\|$. We validate this empirically on the 562 valid Avazu candidates by computing all $\binom{562}{2} = 157{,}641$ pairwise embedding distances and objective differences.

Figure 9 reports four diagnostics. Panel (a) shows the pairwise scatter with binned means: the monotone trend confirms that larger embedding distance corresponds to larger objective difference. Panel (b) plots the mean $|\Delta\mathrm{AUC}|$ among k-nearest neighbors as a function of k; the smooth, monotone increase from 0.004 at k=1 to 0.013 at k=50 confirms local smoothness. Panel (c) shows a t-SNE projection of the embedding colored by AUC, revealing broad spatial structure in objective values. Panel (d) reports the binned Lipschitz ratio $|\Delta f|/\|\Delta z\|$ at each distance scale; the 95th-percentile ratio stabilizes near 0.13.

The global correlation between embedding distance and $|\Delta\mathrm{AUC}|$ is $\rho = 0.245$. The moderate value reflects the low-dimensional (8-d) hand-crafted embedding, the ordinal family encoding, and training noise. Nevertheless, the nearest-neighbor $|\Delta\mathrm{AUC}|$ of 0.004 is small relative to the full AUC range of 0.297, and the bounded Lipschitz ratio supports the use of the GP surrogate and diversity repulsion in the representation space.

D.5 Prompt design, signature embedding, and baseline implementations

PWBO prompts

Init prompt (prior construction). The reference prior π_θ is built by querying the frozen `Qwen/Qwen3.5-9B` model (BF16, thinking disabled) with a fixed prompt that asks the model to select 8 architecture indices from the registered candidate set. The prompt instructs the model to prefer configurations likely to achieve high validation AUC while balancing quality and diversity, and to return a JSON array of 8 integer `arch_indices` values. The prompt is rendered with the full canonical listing of all registered candidate signatures (562 valid entries in the CTR pipeline

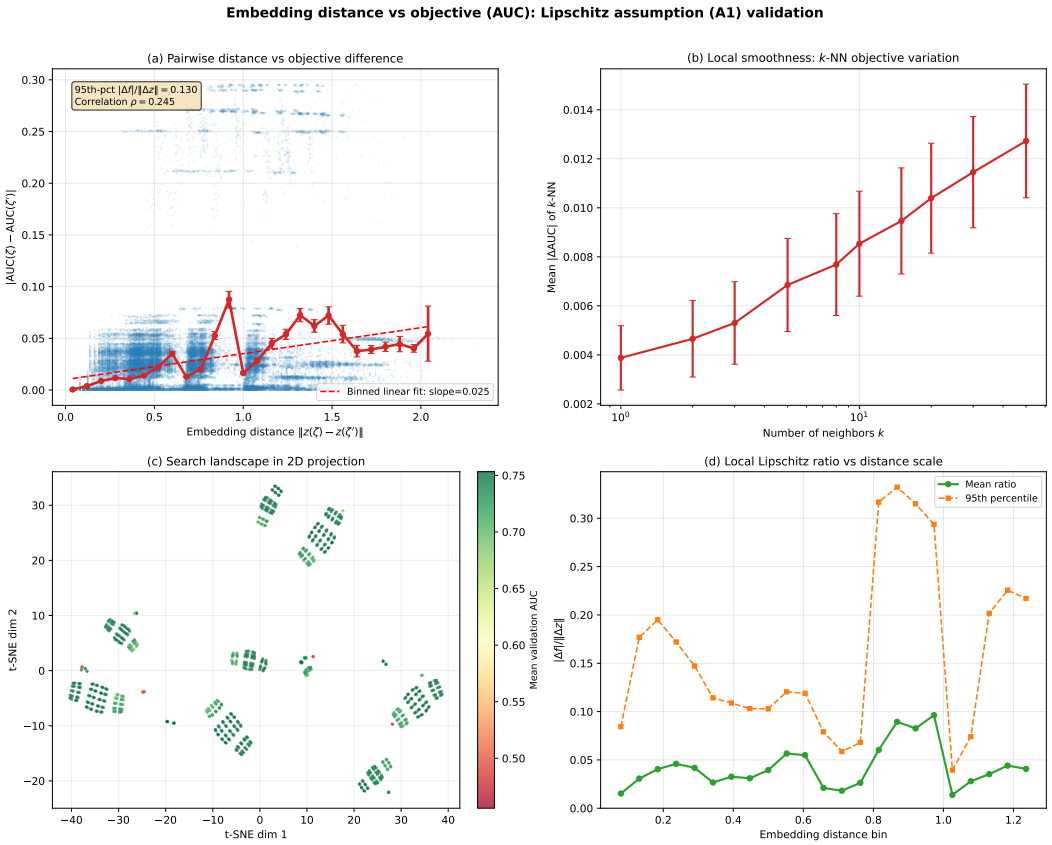

Figure 9: **Embedding smoothness validation for assumption (A1). (a)** Pairwise embedding distance vs. $|\Delta\mathrm{AUC}|$ with binned means (red) and linear trend. **(b)** Mean $|\Delta\mathrm{AUC}|$ among k-nearest neighbors vs. k. **(c)** t-SNE two-dimensional projection colored by mean validation AUC. **(d)** Local Lipschitz ratio $|\Delta f|/\|\Delta z\|$ at each distance scale (mean and 95th percentile).

domain). A pilot phase (304 valid slots) precedes an independent confirm phase (6,000 valid slots); the confirm frequencies $\widehat{q}$ define $\pi_Q = 0.99\,\widehat{q} + 0.01\,U$.

Within-round prompt (two-channel design). Within each search round, PWBO conditions the LLM proposer on observed history through a two-channel prompt. **Channel A (exploitation):** presents the history table of evaluated architectures and their validation AUC values, asks the model to analyze which field values correlate with higher AUC, and requests one new architecture proposal that is structurally close to the current best but not identical to any previously tested architecture. **Channel B (exploration):** presents the set of already-tested and untested $(family_style, embedding_dim, hidden_dim)$ combinations and asks the model to propose one architecture from the untested combinations, ignoring historical performance and focusing purely on structural coverage. The two channels are alternated across core proposal draws within each round, ensuring that the proposal pool contains both exploitation and exploration candidates. Both channels include the full registry of registered configurations and require structured JSON output ({"arch_index": <int>}).

LLAMBO PROMPTS

Init prompt (zero-shot warmstarting). For the first b_{init} rounds, LLAMBO uses a zero-shot prompt that provides the search space schema and the full architecture registry, then asks Qwen to propose the 5 most promising architectures based on domain knowledge alone, without any observed performance data. The prompt requests a JSON array of `arch_indices`.

Within-round prompts (candidate sampling and surrogate modeling). After warmstarting, LLAMBO uses two types of prompts per round. **Candidate sampling:** the prompt presents the history table of evaluated architectures sorted by AUC, asks Qwen to analyze which structural features correlate with high performance, and requests 3 new candidate proposals from untested regions. Output is a JSON array of `arch_indices`. **Surrogate modeling:** the prompt shows the observed (architecture, AUC) pairs and asks Qwen to predict AUC for each unobserved candidate in batches of 20, considering the effects of model family, embedding dimension, network depth, learning rate, hidden dimension, and dropout. Output is a JSON object mapping `arch_index` to predicted AUC values.

Signature embedding. Each architecture signature ζ is mapped to a continuous vector $z(\zeta)$ for the GP surrogate, diversity scoring, and selection. In the CTR domain, each valid signature is embedded as $z(\zeta) \in \mathbb{R}^8$,

$$z(\zeta) = \Big[\tfrac{\mathrm{fam}}{3},\ \tfrac{\log_2(\mathrm{emb})}{8},\ \tfrac{\log_2(h)}{12},\ \tfrac{L}{8},\ p_{\mathrm{drop}},\ r_{\mathrm{bag}},\ \mathbb{1}_{\mathrm{res}},\ \mathrm{clip}\big((\log_{10}\mathrm{lr}+7)/5,\ 0,\ 1\big)\Big],$$

where $\mathrm{fam} \in \{0,1,2,3\}$ is the interaction family (DNN, Wide&Deep, DCN, AutoInt), emb the embedding dimension, h the hidden width, L the layer count, p_{drop} the dropout rate, r_{bag} the feature-bagging ratio, and $\mathbb{1}_{\mathrm{res}}$ the residual-connection indicator. The family is encoded as a normalized ordinal rather than a one-hot block, so Euclidean distance in z couples the categorical family difference with the model-capacity ordering of the four families. In the NAS-Bench-201 domain, each architecture is instead embedded as a 30-dimensional one-hot vector over the six cell edges and five operations (Appendix D.6.6). All coordinates lie in $[0,1]$.

πBO implementation. We implement πBO (Hvarfner et al., 2022) as multiplicative prior-weighted expected improvement with a decaying prior exponent. At completed round t the acquisition score for candidate a is $\alpha_t(a) = \mathrm{EI}(a) \cdot \pi(a)^{\beta_t}$, where $\pi(a)$ is the prior probability (exponentiated log-probabilities, shifted so the maximum equals 1). The exponent β_t is initialized at β_0 and decayed toward $\beta_{\min}$ as optimization progresses, so the prior influence is strongest early and the acquisition gradually reverts to standard EI. For the first t_0 rounds, πBO draws its warmup candidates from the same reference prior π_θ as PWBO (identical cold-start protocol). After warmup, it uses the same Matérn-5/2 GP surrogate and Noisy-EI acquisition as PWBO, but combines them with the prior multiplicatively rather than additively. πBO receives the same Qwen-generated reference prior π_θ as PWBO, so the comparison isolates the effect of prior integration strategy (multiplicative weighting without diversity vs. additive log-prior with repulsion). Crucially, πBO has no diversity or repulsion term; repeated high-EI candidates in a dense prior region are not penalized.

LLAMBO implementation. We implement LLAMBO (Liu et al., 2024) with the frozen `Qwen/Qwen3.5-9B` as the LLM backbone, realizing all three components via in-context learning without fine-tuning. **(1) Zero-shot warmstarting:** for the first b_{init} rounds, Qwen is prompted with the search space description and task objective (without observed performance data) and asked to propose the most promising architectures from domain knowledge alone. **(2) LLM-based candidate sampling:** in each subsequent round, Qwen receives the history table of evaluated architectures and their AUC values, and proposes new candidates from high-potential regions by analyzing which structural features correlate with high performance. **(3) LLM-based surrogate modeling:** Qwen is shown the observed (architecture, AUC) pairs and asked to predict AUC for each unobserved candidate in batches of 20; the z-score normalized predictions form a discriminative surrogate mean $\widehat{f}_{\mathrm{LLM}}(\zeta)$. We then select the next batch by Expected Improvement computed from this surrogate mean and an observation noise estimate, using the best observed AUC as the incumbent.

Our implementation adapts LLAMBO to the fixed-candidate setting: instead of continuous candidate generation, the LLM surrogate predicts AUC for each of the registered architectures, and selection uses EI over these predictions. This differs from the original LLAMBO, which operates in a continuous space with LLM-generated candidates. Both πBO and LLAMBO use the same signature embedding $z(\zeta)$ as PWBO to ensure a fair comparison.

LP-BO implementation. We implement local-penalization batch BO (González et al., 2016) as a strong batch-selection baseline. LP-BO uses the same GP surrogate and Noisy-EI acquisition as PWBO but replaces the prior-weighted repulsion with a distance-based batch penalty. Within

Table 11: Selection criteria for PWBO and comparison methods.

Method	Selection criterion
BO	$\zeta_{n+1} = \arg\max_\zeta \alpha(\zeta; m_t)$
πBO	$\zeta_{n+1} = \arg\max_\zeta \pi(\zeta)^{\beta_t} \alpha(\zeta; m_t)$
LLAMBO	$\zeta_{n+1} = \arg\max_\zeta \mathrm{EI}\big(\zeta; \widehat{f}_{\mathrm{LLM}}, \sigma_{\mathrm{LLM}}\big)$
LP-BO	$\zeta_{\text{next}} = \arg\max_\zeta \alpha(\zeta; m_t) \cdot \prod_j \phi_j(\zeta)$
PWBO	$\zeta_{\text{next}} = \arg\max_{\zeta \in \Sigma_n \setminus S} \{\log \widetilde{q}_n(\zeta) + \beta_t \alpha(a_n(\zeta); m_t) - \text{repulsion}_S(\zeta)\}$

each round, the first candidate is $\arg\max$ EI. Subsequent candidates are selected by maximizing $\mathrm{EI}(\zeta) \cdot \prod_j \phi_j(\zeta)$, where $\phi_j(\zeta) = \min\{1, \max\{0, (\|z(\zeta) - z(\zeta_j)\| - r)/r\}\}$ is a linear penalty centered at each already-selected batch member ζ_j, with penalty radius $r = 0.5$ matching PWBO's diversity bandwidth. LP-BO receives the same Qwen prior as PWBO but does not use it in the selection score; this isolates the effect of batch diversity without prior guidance.

Table 11 summarizes the per-step selection criteria of PWBO and the comparison methods.

D.6 Further analyses

D.6.1 Hierarchical search and local Lipschitz

When the global Lipschitz constant is large because high-level fields interact, a practical heuristic is to search hierarchically: first choose architecture-level fields, then optimize training and feature choices inside the selected region. This can reduce local geometric complexity, but the present theorem does not imply a summed per-level bound. A formal extension would need, at every level, its own mass–geometry alignment, effective overhead, noise floor, and an accounting of errors propagated from earlier commitments.

D.6.2 Treating the schema space as continuous

The decoded schema (Table 5) is a finite set of field choices, yet the analysis works in a continuous embedding. In the experiments we map each signature to a hand-crafted continuous vector $z(\zeta)$ (8-dimensional normalized features for CTR pipelines, 30-dimensional one-hot for NAS-Bench-201; details are given in Appendix D.5). It is this embedding, not the discrete field enumeration, that carries the metric and measure-theoretic structure the theory uses: distances $\|z(\zeta) - z(\zeta')\|$, the prior measure π_θ, and the near-optimal sets $\mathcal{Z}_\varepsilon$. The continuous picture is faithful only above the representation floor defined in Appendix A.2.1; below it neighbouring schemas are no longer separated by the embedding and the problem reverts to discrete best-arm identification. This is why the formal near-optimality dimension is defined over $[\mu\Delta, \varepsilon_{\max}]$ rather than only as $\varepsilon \to 0$.

Fragmentation is the one place where the continuous picture cannot be borrowed silently. A finite image is totally disconnected in the ambient topology, so counting topological components would return one component per point. Assumptions (A5)–(A6) therefore use the explicit scale $s_{\mathrm{conn}} \geq \Delta$ from Appendix A.2.1. A continuous relaxation agrees with this reading when distinct pieces are separated by more than s_{conn} and sampled densely enough that missing points neither split nor merge a piece.

D.6.3 What the prior controls

Two things set d_{eff}. One is the shape of $\bar{f}$ near its optimum, since a flatter optimum makes $\mathcal{Z}_\varepsilon$ larger at every tolerance. The other is where π_θ places its mass, since Definition 2 measures $\mathcal{Z}_\varepsilon$ under the prior rather than under Lebesgue measure. Only the second is under our control, and it is the reason the same objective can have very different exchange rates under different proposers.

The uninformative case fixes a reference scale. If π_θ is uniform on a d-dimensional box and $\bar{f}$ grows sharply away from its optimum, then $\mathcal{Z}_\varepsilon$ is a ball of radius proportional to ε and $\pi_\theta(\mathcal{Z}_\varepsilon) \propto \varepsilon^d$, so $d_{\mathrm{eff}} = d$. An informative prior can reduce this finite-scale exponent when its concentration increases toward the optimum as tolerance tightens; it may instead change only constants.

This distinguishes an exponent effect from a warm-start effect. A better initial point changes the range factor $\log(W/\varepsilon)$ once. A smaller fitted d_{eff}, when the geometric alignment assumption is also unchanged, reduces the predicted cost accumulated across rounds.

Prior bias enters the same exponent. A fixed reference π_θ that systematically underweights the region containing the optimum lowers p_ε at the affected scales, raising $\log(1/p_\varepsilon)/\log(W/\varepsilon)$ and potentially d_{eff}; the search slows accordingly. The failure is recoverable only insofar as a later q_n-generated pool contains a candidate from that region. The bound $q_n(A) \geq \eta\pi_\theta(A)$ preserves this possibility independently of the history; once the candidate is present, acquisition can promote it but cannot generate it. A larger core pool raises the chance of a hit, while optional multi-temperature draws may add coverage without entering the guarantee. Under (A1) and the fixed-support comparator, every positive resolved tolerance has $p_\varepsilon > 0$; an operational generator or parser with zero near-optimal mass therefore defines a smaller reachable support and requires the comparator to be recomputed (Appendix D.6.4).

D.6.4 Biased prior analysis

Because $\zeta^\star \in \text{supp}(\pi_\theta)$ and (A1) places a sufficiently small embedding neighborhood inside $\mathcal{Z}_\varepsilon$, the support definition gives $p_\varepsilon > 0$ for every positive resolved tolerance. If measured proposal mass is zero, the implemented generator–parser pipeline does not realize the theoretical reference support; the guarantee must then use its actual support and best reachable comparator. More subtly, a reference that assigns positive but small mass to the optimal region increases $\log(1/p_\varepsilon)/\log(W/\varepsilon)$ at the affected scales and can therefore increase d_{eff}. The exploration floor guarantees $q_n(\mathcal{Z}_\varepsilon) \geq \eta p_\varepsilon$, so increasing the core pool size $|\mathcal{C}_n|$ raises the chance that a later pool contains such a candidate. Acquisition can promote the candidate only after this proposal event; it cannot recover a signature absent from the pool. Pre-registered criterion: degradation is graceful if PWBO's median AUC stays $\geq$ BO-random-init for $T \geq 20$.

D.6.5 Greedy construction of the space-filling set

Apply farthest-point traversal to $\Sigma_{1,\tau_{\text{prior}}}$: choose any $\zeta_1 \in \Sigma_{1,\tau_{\text{prior}}}$, then add $\zeta_{j+1} = \arg\max_{\zeta \in \Sigma_{1,\tau_{\text{prior}}}} \min_{i \leq j} \|z(\zeta) - z(\zeta_i)\|$ until b signatures have been selected. This is a 2-approximation to the discrete k-center objective in Equation (27), costing $O(|\Sigma_{1,\tau_{\text{prior}}}| \cdot b)$. Each selected signature is executed through its fixed representative $a_1(\zeta)$.

D.6.6 NAS-Bench-201 validation

To validate that PWBO's advantages transfer beyond CTR pipeline search, we conduct experiments on NAS-Bench-201 (Dong & Yang, 2020), a standard benchmark for neural architecture search with fully enumerated architectures and pre-computed training curves.

Search space and setup. NAS-Bench-201 defines a cell-based search space with 4 nodes and 6 edges, where each edge selects from 5 operations: `none`, `skip_connect`, `nor_conv_1x1`, `nor_conv_3x3`, and `avg_pool_3x3`. The total space contains $5^6 = 15{,}625$ unique architectures. We use the CIFAR-10 validation accuracy at epoch 200 as the objective, with the best architecture achieving 91.61%. The target threshold is set to $\varepsilon = 1.0$ (within 1% of the optimum).

Prior construction. We construct the LLM prior π_θ using the same protocol as in the Avazu experiments. Specifically, we prompt Qwen3.5-9B to select the top-8 architectures it believes will achieve high validation accuracy on CIFAR-10. We sample 500 architectures in a pilot phase and 2,000 in a confirmation phase, yielding a prior distribution over the 15,625 candidates. The resulting prior places higher mass on architectures with 3x3 convolutions and skip connections, reflecting the model's domain knowledge. The top-10 architectures by prior probability achieve an average accuracy of 88.36%, compared to 84.17% for random selection, confirming that the prior contains useful signal.

Embedding. Each architecture is encoded as a 30-dimensional vector: for each of the 6 edges, we use a 5-dimensional one-hot encoding of the operation choice, concatenated across all edges. This

yields a compact representation for the Matérn-5/2 GP surrogate and Noisy-EI acquisition used by the GP-based methods.

Baselines and protocol. We compare PWBO against random search, standard BO, πBO, LLAMBO, and LP-BO. All methods use the same Qwen prior π_θ and embedding. We use the nominal cap $T = 256$, vary the batch width $b \in \{1, 2, 4, 8\}$, and run each configuration with 10 random seeds. Thus $N_{\max} = \lfloor 256/b \rfloor$ complete batches are executed, with no early stopping or incomplete final batch. Completed batches are numbered from one; a non-hit is assigned $N_{\max}$ and identified separately through hit rate. The cold-start protocol is identical across all prior-based methods: sample from π_θ for completed round 1, then switch to the method-specific acquisition strategy.

Results. Table 12 reports mean $\pm$ standard deviation of cap-coded completed rounds and the corresponding hit rate over 10 seeds. PWBO achieves the lowest final regret (0.23) and competitive convergence speed (2.0 rounds at $b = 4$), outperforming all baselines. LLAMBO, which lacks the diversity mechanism, achieves higher regret (0.44) and requires more rounds (7.5 at $b = 4$), confirming that PWBO's diversity repulsion helps avoid local optima in discrete structured spaces.

Table 12: NAS-Bench-201 rounds to target under nominal $T = 256$ (lower is better). Each cell reports mean $\pm$ standard deviation of cap-coded completed rounds, with hit rate in parentheses. The first evaluated batch is round 1.

Method	$b = 1$	$b = 2$	$b = 4$	$b = 8$
PWBO (Qwen)	3.0 ± 0.4 (1.00)	2.0 ± 0.0 (1.00)	2.0 ± 0.4 (1.00)	1.8 ± 0.4 (1.00)
πBO	16.5 ± 24.4 (1.00)	5.1 ± 4.7 (1.00)	5.1 ± 2.8 (1.00)	4.3 ± 3.6 (1.00)
LLAMBO (Qwen)	3.1 ± 0.5 (1.00)	15.2 ± 14.9 (1.00)	7.5 ± 7.1 (1.00)	6.4 ± 8.7 (0.90)
LP-BO	36.5 ± 36.0 (1.00)	54.7 ± 48.7 (0.80)	17.5 ± 20.6 (0.90)	3.8 ± 2.1 (1.00)
BO	37.8 ± 73.3 (1.00)	9.3 ± 9.0 (1.00)	14.6 ± 20.2 (0.90)	2.8 ± 1.5 (1.00)
Random	35.9 ± 26.2 (1.00)	16.1 ± 15.0 (1.00)	7.5 ± 6.0 (1.00)	4.6 ± 4.2 (1.00)

Figure 10 visualizes the convergence behavior at $b = 4$. PWBO (blue) reaches low regret within 2 completed rounds and maintains the best final solution quality. The diversity mechanism prevents premature convergence to suboptimal architectures, a benefit that is particularly pronounced in discrete search spaces where local optima are common.

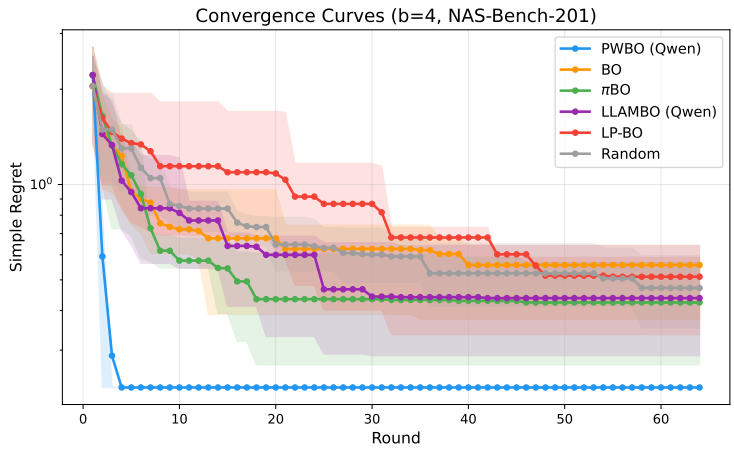

Figure 10: NAS-Bench-201 convergence curves at $b = 4$, indexed by completed batches from round 1. PWBO achieves the lowest final regret and competitive convergence speed, validating its effectiveness on structured discrete search spaces beyond CTR pipeline optimization.

Discussion. These results confirm that PWBO's advantages are not specific to the Avazu CTR task. The method generalizes to a fundamentally different domain (neural architecture search) with a larger search space (15,625 vs. 576 candidates) and a different objective (image classification accuracy vs. CTR prediction AUC). The consistency of PWBO's performance across both domains supports the claim that the depth–width exchange and diversity-aware acquisition are general principles for batch Bayesian optimization with LLM priors.

Mechanism ablation on NB201. Table 13 reports the 2×2 repulsion$\times$elimination ablation on NB201 at b=8. On this larger search space, elimination prunes 84.6% of candidates (13,220 of 15,625), substantially reducing the search space for subsequent rounds. Removing within-batch repulsion increases rounds-to-target from 2.0 to 3.0 (regret unchanged at 0.23), confirming that diversity-aware selection also helps on this benchmark, while elimination does not change rounds-to-target. The quality analysis further confirms that elimination correctly identifies low-quality candidates: eliminated architectures have lower mean accuracy (83.4 vs. 85.2), and 91.3% of eliminated candidates fall below the 90th percentile of true quality.

Table 13: NAS-Bench-201 mechanism ablation (nominal T=64, b=8, 10 seeds). Rounds count completed batches from one; here $N_{\max} = 8$ and all variants hit the target. Elimination prunes 85% of the search space while preserving top candidates.

Variant	Rounds	Regret	#Elim	Elim%
Full PWBO	2.0	0.23	13,220	84.6%
No repulsion	3.0	0.23	12,898	82.5%
No elimination	2.0	0.23	0	0%
Bare (prior + EI)	3.0	0.23	0	0%

E Discussion and Future Directions

The depth–width exchange isolates two levers that a practical search procedure can act on: near-optimal prior mass and effective within-round coverage. PWBO operationalizes both, with prior-weighted proposals concentrating expensive evaluations where the reference distribution places mass and repulsive batch selection converting additional width into additional coverage. The experiments in Section 6 show that both levers behave as the characterization predicts and that the two mechanisms are complementary rather than redundant.

Several natural extensions would broaden the reach of the present results. First, the theorem assumes coverage of the plausible set itself, whereas PWBO realizes coverage within a finite proposal pool; formalizing this interface is a natural next step, and our empirical coverage measurements provide a template for such an analysis. Second, the signature embedding $z(\cdot)$ is validated on Avazu in Appendix D.4, where the diagnostics support its use at the resolved scales of the benchmark; extending the approach to learned h brepresentations that carry objective-relevant variation across domains is a promising direction. Third, the present theory fixes the allocation between distinct designs and replicated evaluations and applies above the representation and statistical floors; adaptive replication, multi-fidelity evaluation, and benchmarks without a scalar objective would each enrich the resource model and are natural subjects for future work.