COMAS: CO-OPTIMIZING MODEL AND ACTION-SELECTION POLICY IN CONTEXTUAL BANDITS

Anonymous authors
Paper under double-blind review

ABSTRACT

The performance of contextual bandit algorithms critically depends on the model configuration used to estimate the reward function, including its family, architecture, and regularization. However, selecting an effective configuration requires data collected by the action-selection policy, which itself depends on the reward estimator, creating a circular dependency between model configuration and policy learning. We introduce COMAS, a framework that co-optimizes model configuration and action selection by treating configuration as a sequential decision variable. At each episode, COMAS selects a model configuration, trains the corresponding reward estimator on historical interaction data, and uses a Gaussian process surrogate to model the episode-level reward as a function of the configuration, thereby guiding subsequent configuration selection. Under suitable assumptions, we show that COMAS achieves a sublinear regret upper bound that separately accounts for regret due to configuration selection and policy learning. Experiments on synthetic and real-world contextual bandit problems demonstrate the effectiveness of co-optimizing heterogeneous model configurations and action-selection policies.

1 INTRODUCTION

Contextual bandits (Slivkins, 2019; Lattimore & Szepesvári, 2020) is a framework that models sequential decision-making problems in which a learner observes the context of each request, such as user profiles, locations, and purchase histories, and selects an action (arm) to maximize the cumulative reward. This framework has a wide range of real-world applications, including personalized recommendation (Li et al., 2010), online advertising (Geng et al., 2021), personalized healthcare (Yao et al., 2021; Alban et al., 2026), and clinical trials (Aziz et al., 2021; Varatharajah & Berry, 2022).

Contextual bandit algorithms (Li et al., 2010; Zhang et al., 2021b) rely on accurate reward estimation to effectively balance exploration and exploitation. The quality of the reward estimation depends critically on the choice of model configuration, including the model family, architecture, and regularization (Foster et al., 2019; Kang et al., 2024). However, the appropriate model configuration is generally unknown a priori and must be selected from data collected through bandit interactions with the environment. This creates a *circular dependency*: the model configuration determines the reward estimator and, consequently, which arms are selected and what data are collected, while the resulting data determine which configuration is effective.



Figure 1: **Contextual bandit interaction loop.** The action-selection policy selects actions based on context, observes noisy rewards, and updates the reward estimator to guide future action selection.

Conventional approaches typically fix the model configuration before interaction (Zhou et al., 2020; Zhang et al., 2021a) or tune hyperparameters separately from the action-selection process (Falkner et al., 2018; Kang et al., 2024), failing to account for the feedback loop among model choice, action selection, and the data collected during interaction. This motivates the following question: ***How can we co-optimize model configuration and arm-selection policy in contextual bandits?*** To address this challenge, we propose COMAS, a framework that co-optimizes model configuration and arm-selection policy by treating model configuration as a sequential decision variable. This co-optimization introduces several key challenges that we address with a set of novel techniques.

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.

① **Heterogeneous configuration and policy co-optimization.** Existing model-selection methods (Agarwal et al., 2017; Foster et al., 2020; Pacchiano et al., 2020b; Cutkosky et al., 2021) provide regret guarantees for selecting among base learners online, while online hyperparameter-tuning methods adapt statistical parameters within a bandit algorithm (Ding et al., 2022; Kang et al., 2024). Our setting differs in that a configuration can jointly specify heterogeneous components, including the learner family, architecture, and regularization, and its reward estimator determines the action-selection policy. This creates a coupling between configuration and policy, as each influences the data available to the other. We address this coupling through episodic configuration selection: at the start of each episode, a meta-controller selects a configuration that remains fixed for several rounds and induces the corresponding action-selection policy. The resulting episode reward then guides subsequent selections through a Gaussian process (GP) (Srinivas et al., 2010; Chowdhury & Gopalan, 2017) surrogate over configurations, exploiting structure across heterogeneous configurations.

② **Episodic evaluation with varying episode lengths.** Since sub-optimal configurations may perform poorly, evaluating them over longer episodes can lead to sub-optimal action selection. We therefore use shorter episodes for new configurations in early rounds and progressively longer episodes for configurations that remain promising in later rounds. Since configuration performance is observed through episode-level average rewards, longer episodes provide more reliable estimates but also lead to length-dependent observation noise. We therefore use a heteroscedastic GP likelihood (Makarova et al., 2021), with observation variance determined by episode length.

③ **Regret analysis.** Since we co-optimize the model configuration and action-selection policy, analyzing regret is challenging because the two levels are coupled: the configuration determines how actions are selected, while the resulting data determines how well each configuration can be evaluated. We therefore separate the total regret into an initial calibration cost and four post-calibration effects. First, the configuration pool itself may not contain a configuration whose limiting performance matches the contextual-bandit oracle. Second, the meta-controller may not always select the best limiting configuration in the pool. Third, because every configuration is repeatedly refit on the growing shared dataset, its performance can drift before stabilizing. Finally, episode-level rewards contain stochastic variation. (more details are in Sec. 3).

Building on the challenges outlined above and the methods proposed to address them, we summarize the key contributions of this paper as follows:

- **Structured shared-dataset configuration selection.** We formulate episodic online selection over heterogeneous contextual bandit configurations using the raw episode-average reward as the meta signal and a common interaction dataset shared across learner configurations.
- **Regret analysis under shared-data retraining.** We develop a general regret decomposition that separates the effects of the initial calibration, the quality of the available candidate models, online model selection, repeated retraining on growing shared data, and stochastic feedback. We further derive a high-probability regret bound for the UCB-based variant while accommodating predictably learned GP hyperparameters, including ARD kernels, through the confidence radius.
- **Sufficient condition for stable retraining.** We give a concrete condition under which repeatedly retraining a learner on growing shared data leads to stable long-run performance. We verify these conditions for well-specified ridge learners under standard excitation and margin assumptions. For misspecified ridge learners, we provide a separate result showing that stable long-run behavior holds under the additional assumption that the training-data statistics converge over time.
- **Experimental results.** We evaluate COMAS across discrete search spaces spanning linear, polynomial, and neural learners, and separately study a mixed continuous extension that jointly selects a learner family/architecture and optimizes continuous hyperparameters within that choice.

1.1 Related Works

This section focuses on the most relevant work: online model and algorithm selection, and GP-based online hyperparameter optimization. Contextual bandits with fixed model classes are in Sec. A.

Online model selection and algorithm selection. CORRAL (Agarwal et al., 2017) and other model-selection methods (Pacchiano et al., 2020b; Arora et al., 2021; Cutkosky et al., 2021) combine a finite collection of base bandit algorithms and compete with the best member of the collection. Regret balancing schedules base learners to keep their empirical regrets of the same order (Abbasi-Yadkori et al., 2020), while elimination methods remove learners whose observed regret violates their

candidate bounds (Pacchiano et al., 2020a). Related reductions adapt to unknown misspecification or model complexity (Foster et al., 2019; 2020). These results show that online model selection can achieve sublinear overhead. Related work also arises in bandit-based hyperparameter search for AutoML, such as Hyperband (Li et al., 2018), which evaluates configurations offline using validation loss rather than bandit feedback, and in meta-learning of exploration strategies across simulated tasks (Sharaf & Daumé III, 2019), which adapts across tasks rather than within a single online instance. COMAS differs in three respects. First, its framework can operate without requiring a well-specified candidate configuration. Second, all configurations share a common interaction dataset, allowing data collected under one configuration to train another. Third, COMAS uses a GP surrogate over configurations to transfer information across them, rather than treating each configuration independently.

Online hyperparameter optimization and GP-based selection. Syndicated Bandits (Ding et al., 2022) place an outer learner over a grid of exploration parameters and avoid regret bounds that grow exponentially with the number of tuned hyperparameters. Kang et al. (2024) study continuous hyperparameter optimization for generalized linear contextual bandits, tuning exploration parameters online using arm rewards as feedback. Bandit-over-bandit methods tune algorithmic parameters in non-stationary environments (Cheung et al., 2019), while Population-Based Bandits (Parker-Holder et al., 2020) use a probabilistic surrogate to tune hyperparameter schedules online in reinforcement learning. Offline methods such as BOHB (Falkner et al., 2018) instead tune hyperparameters separately from interaction. On the modeling side, GP-UCB and kernelized Thompson sampling exploit covariance structure to generalize across a search space (Srinivas et al., 2010; Chowdhury & Gopalan, 2017), while heteroscedastic GP bandits allow observation noise to vary across queries (Makarova et al., 2021). Unlike existing online hyperparameter-tuning methods, which typically tune statistical or exploration parameters within a fixed learner family, COMAS selects from a heterogeneous set of configurations spanning model families, architectures, and regularization.

2 PRELIMINARIES

Contextual bandits. We consider a stochastic contextual bandit problem with finitely many actions, in which the learner sequentially selects an action for each context to maximize the cumulative reward. Let $\mathcal{C}$ denote the context space, $\mathcal{A}$ the action set, and $\mathcal{X} \subset \mathbb{R}^d$ the space of context–action feature vectors. For context c_t and action a, let $x_{t,a} = \varphi(c_t, a) \in \mathcal{X}$, where $\varphi : \mathcal{C} \times \mathcal{A} \to \mathcal{X}$ is a known feature map. At the beginning of round t, the environment reveals a context $c_t \in \mathcal{C}$ and an available action set $\mathcal{A}_t \subseteq \mathcal{A}$. The learner then selects an action $a_t \in \mathcal{A}_t$ and observes a stochastic reward $y_t \doteq f(x_{t,a_t}) + \varepsilon_t$, where $f : \mathcal{X} \to [0,1]$[1] is the unknown mean reward function and ε_t is R-sub-Gaussian reward noise. Let $\mathcal{J}_{t-1}$ denote the information available before round t, and let $\mathcal{G}_t$ denote the information available after x_{t,a_t} is selected but before ε_t is observed. Then, for every $\lambda \in \mathbb{R}$, $\mathbb{E}\left[\exp^{\lambda\varepsilon_t} \mid \mathcal{G}_t\right] \leq \exp\left(\lambda^2 R^2/2\right)$.

Model configurations. To select an action for a given context, the learner first estimates the unknown reward function f. The quality of this estimate depends on the choice of model family, architecture, and regularization, which together define a *configuration*. Let $\mathcal{M}$ denote the set of finite[2] configurations. Because f is unknown and potentially non-linear, the quality of any configuration $m \in \mathcal{M}$ is unknown a priori. Each configuration induces a different trade-off between approximation bias and estimation variance. Moreover, configuration components can interact, so the effect of one component may depend on the settings of others. Thus, the configuration that best balances these factors is unknown a priori, and the learner must adapt the *entire configuration* m jointly, rather than tuning an individual component (Ding et al., 2022).

Performance measure. Let $a_t^\star \doteq \operatorname{argmax}_{a \in \mathcal{A}_t} f(x_{t,a})$ denote an optimal action for context c_t. Selecting action a_t incurs instantaneous regret $r_t \doteq f(x_{t,a_t^\star}) - f(x_{t,a_t}) \geq 0$. Our goal is to design a sequential policy π that jointly adapts the model configuration and selects an action a_t at each round to minimize the *cumulative regret* over T rounds:

$$\mathfrak{R}_T(\pi) \doteq \sum_{t=1}^{T} \left(f(x_{t,a_t^\star}) - f(x_{t,a_t})\right). \tag{1}$$

[1] This assumption is not restrictive since any bounded function can be rescaled to satisfy this requirement.

[2] In our experiments, we also consider continuous components.

A policy π is a good policy if it achieves *sub-linear regret*, i.e., $\lim_{T\to\infty} \mathfrak{R}_T(\pi)/T = 0$, which implies that its average regret vanishes as the number of rounds increases.

3 COMAS: CO-OPTIMIZING MODEL AND ACTION-SELECTION POLICY

In this section, we present COMAS, a framework for online co-optimization of model configuration and action-selection policy in contextual bandits. COMAS consists of two key components: model configuration selection and action-selection policy, discussed in the subsequent sections.

3.1 MODEL CONFIGURATION SELECTION

Our model configuration consists of heterogeneous components, including the learner family, architecture, and regularization. Since the reward estimator determines the action-selection policy, configuration selection and action selection are coupled: the selected configuration determines how actions are chosen. At the same time, the resulting observations affect the information available for future configuration selection. To address this coupling, we use episodic configuration selection. At the start of each episode, a meta-controller selects a configuration that remains fixed for several rounds and induces the corresponding action-selection policy.

Episodic-level feedback for model configuration. Let episode q consist of H_q consecutive rounds, with starting round τ_q, and let $\mathcal{H}_q \doteq \mathcal{J}_{\tau_q-1}$ denote the information available at the start of episode q. At the beginning of episode q, the meta-controller selects a configuration $m_q \in \mathcal{M}$, which remains fixed throughout the episode. The corresponding base learner is refit on the shared dataset and selects actions according to the time-dependent policy $\pi_t^{m_q}$ for rounds $t \in \{\tau_q, \ldots, \tau_q + H_q - 1\}$. To measure the performance of the configuration in the episode, we use episode-level feedback (average rewards) $\bar{y}_q \doteq \frac{1}{H_q} \sum_{t=\tau_q}^{\tau_q+H_q-1} y_t$.

Varying episode length. Since sub-optimal configurations can lead to poor action selection, evaluating them over long episodes yields lower cumulative rewards and, consequently, higher regret. We therefore use shorter episodes for new configurations in early rounds and progressively increase the episode length for promising configurations in later rounds. Since episode-level average rewards measure configuration performance, longer episodes yield more reliable estimates but also introduce length-dependent observation noise. We use a scheduler that chooses H_q before episode q, with episode lengths bounded by $H_q \le C_H(\tau_q + 1)^\beta$ for constants $C_H \ge 1$ and $0 \le \beta < 1$. The episode length also determines the feedback noise seen by the meta-controller: since $\bar{y}_q$ averages H_q round-level rewards, its observation variance decreases with H_q. Accordingly, the meta-GP uses an observation variance that adapts to H_q. This design balances shorter commitments when configuration quality is uncertain against more precise feedback from longer episodes. Our current regret analysis accommodates varying episode lengths but treats this schedule conservatively.

Selecting model configuration via heteroscedastic GP. Since the variance of the episode-level feedback depends on the episode length, we model it with a Gaussian process (GP) with heteroscedastic observation noise (Kirschner & Krause, 2018; Makarova et al., 2021). Before episode q, the GP may additionally relearn a scalar residual likelihood-noise term $\lambda_{\mathrm{n},q} \in [\underline{\lambda}_{\mathrm{n}}, \overline{\lambda}_{\mathrm{n}}]$. For a retained observation from episode $q' < q$, the current GP fit uses observation-noise variance $\gamma_{q',q} \doteq \sigma_m^2/H_{q'} + \lambda_g + \lambda_{\mathrm{n},q}$, where $\sigma_m^2, \lambda_g > 0$. The term $\sigma_m^2/H_{q'}$ captures the reduction in observation variance from averaging $H_{q'}$ round-level rewards, λ_g is a fixed episode-length-independent noise floor, and $\lambda_{\mathrm{n},q}$ is the bounded scalar residual noise relearned for the current GP fit.

Estimating performance of model configuration. We use $g : \mathcal{M} \to [0, 1]$ to denote the stabilized performance of a configuration. In the stationary heteroscedastic-GP setting of Makarova et al. (2021), $\mathbb{E}[\bar{y}_q] \doteq g(m_q)$; in our shared-dataset setting, Assumption 1 instead allows $\mathbb{E}[\bar{y}_q \mid \mathcal{H}_q] = g(m_q) + b_q$. Before episode q, let θ_q denote the GP kernel hyperparameters learned from the retained meta history, and write $k_q(m, m') \doteq k_{\theta_q}(m, m')$. For the fixed-kernel stationary special case used below to connect with existing GP-UCB theory, we retain the standard assumption that g belongs to the RKHS induced by a fixed kernel k, with $\|g\|\, k \le \mathcal{B}_g$ for some $\mathcal{B}_g > 0$.

For that fixed-kernel stationary special case, following standard practice in the GP literature (Chowdhury & Gopalan, 2017), we assume without loss of generality that $k(m, m') \le 1$ for all $m, m' \in \mathcal{M}$.

We also assume bounded episode-level feedback, i.e., $|\bar{y}_q| \le \mathcal{B}_q$ for all $q \ge 1$. Conditioned on the selected configurations and their corresponding episode-level feedback, the current heteroscedastic GP posterior mean and covariance are given by

$$\mu_{q-1}(m) = \mathbf{k}_{q-1,q}(m)^\top (\mathbf{K}_{q-1,q} + \mathbf{\Gamma}_{q-1,q})^{-1} \bar{\mathbf{y}}_{q-1},$$
$$\sigma^2_{q-1}(m, m') = k_q(m, m') - \mathbf{k}_{q-1,q}(m)^\top (\mathbf{K}_{q-1,q} + \mathbf{\Gamma}_{q-1,q})^{-1} \mathbf{k}_{q-1,q}(m')$$

where $\mathbf{\Gamma}_{q-1,q} \doteq \operatorname{diag}(\gamma_{1,q}, \ldots, \gamma_{q-1,q})$ is the likelihood-noise matrix used in the current fit, $\mathbf{k}_{q-1,q}(m) \doteq (k_q(m, m_{q'}))^\top_{q' \in [q-1]}$, $\mathbf{K}_{q-1,q} \doteq (k_q(m_{q'}, m_{q''}))_{q',q'' \in [q-1]}$, and $\bar{\mathbf{y}}_{q-1} \doteq (\bar{y}_1, \ldots, \bar{y}_{q-1})^\top$. For brevity, we denote $\sigma^2_{q-1}(m) = \sigma^2_{q-1}(m, m)$.

Selection next model configuration. Upper confidence bound (UCB) is a widely used approach for balancing exploration and exploitation in sequential decision-making (Srinivas et al., 2010; Chowdhury & Gopalan, 2017). We use a UCB-based strategy to select configurations in our meta-level optimization problem. At the start of episode q, we update the GP posterior mean and standard deviation using the retained meta-level observations and their corresponding heteroscedastic observation-noise variances. We then select the configuration m_q for the next episode by maximizing the UCB:

$$m_q = \arg\max_{m \in \mathcal{M}} \left(\mu_{q-1}(m) + \alpha_q \sigma_{q-1}(m) \right), \tag{2}$$

For the fixed-kernel stationary case, Kirschner & Krause (2018) and Makarova et al. (2021) give the heteroscedastic GP-UCB coefficient $\alpha^{\text{stat}}_q = \mathcal{B}_g + \sqrt{2 \ln \left(\frac{\det(\mathbf{K}^{\text{stat}}_{q-1} + \mathbf{\Gamma}^{\text{stat}}_{q-1})^{1/2}}{\delta \det(\mathbf{\Gamma}^{\text{stat}}_{q-1})^{1/2}} \right)}$, where $\mathbf{K}^{\text{stat}}_{q-1}$ and $\mathbf{\Gamma}^{\text{stat}}_{q-1}$ are the corresponding fixed-kernel Gram and likelihood matrices. Our UCB uses α_q in Eq. (2), with the theory requiring Assumption 4 in the shared-dataset setting. Alternatively, we can use Thompson sampling (TS) (Agrawal & Goyal, 2013; Chowdhury & Gopalan, 2017) to select the next configuration by drawing a sample function from the GP posterior, i.e., $\hat{g}_q \sim \text{GP}(\mu_{q-1}(\cdot), \alpha^2_q \sigma^2_{q-1}(\cdot))$ and then choosing its maximizer:

$$m_q = \arg\max_{m \in \mathcal{M}} \hat{g}_q(m). \tag{3}$$

After selecting the configuration m_q, the learner runs the corresponding configuration for episode q and obtains an episode-level reward $\bar{y}_q$ based on H_q round-level rewards. Before starting the next episode, the learner incorporates the available episode-level feedback into the GP posterior. The same procedure is repeated for subsequent episodes, with each newly selected configuration evaluated using the shared-dataset learner at its current state.

3.2 Action-Selection Policy

Let $\mathfrak{A}$ be a contextual bandit algorithm such that $|f^{\mathfrak{A}}_t(x) - f(x)| \le h(x, \mathcal{O}_t)$ holds with probability at least $1 - \delta$ for any $x \in \mathcal{X}$, and $\sqrt{\sum_{t=1}^{T} [h(x_{t,a_t}, \mathcal{O}_t)]^2} = \tilde{O}(\tilde{d} \log T)$, where $\tilde{d}$ denotes the effective dimension. This condition is satisfied by many contextual bandit algorithms, including Lin-UCB (Chu et al., 2011), UCB-GLM (Li et al., 2017), IGP-UCB (Chowdhury & Gopalan, 2017), and Neural-UCB (Zhou et al., 2020). We assume these guarantees also hold for the algorithm $\mathfrak{A}$ used by COMAS for action selection. At each round $t \in [\tau_q, \tau_q + H_q - 1]$, COMAS first estimates the reward function using the selected configuration m_q and then selects the action a_t by maximizing

$$\text{UCB}_t(x) = f^{\mathfrak{A}}_t(x) + h(x, \mathcal{O}_t). \tag{4}$$

The exact form of $h(x, O_t)$ depends on the reward estimator and exploration rule induced by m_q. In the regret analysis of Sec. 4, the resulting evolution of the base action-selection policy is captured through the limiting configuration value $g(m)$ and the shared-dataset drift term.

3.3 COMAS: Co-optimizing Model Configuration and Action Selection Policy

COMAS combines the configuration selection of Sec. 3.1 with the action-selection policy of Sec. 3.2 and works as follows. It first runs an initial calibration phase of T_{cal} rounds whose observations are added to the shared interaction dataset $\mathcal{D}_t \doteq \{(x_{s,a_s}, y_s)\}_{s=1}^{t-1}$ (Line 1). Episode-level feedback from calibration may be discarded, in which case the corresponding model configurations and episode-level

feedback are excluded from the retained meta history $\mathcal{E}_0$ and are not used for GP fitting (Line 2). At the start of episode q, the meta-controller chooses the GP kernel hyperparameters θ_q and scalar residual likelihood noise $\lambda_{\mathrm{n},q}$ using only information available before episode q, constructs the current likelihood matrix $\mathbf{\Gamma}_{q-1,q}$ from the retained episode lengths, and updates the heteroscedastic GP posterior using $\mathcal{E}_{q-1}$ (Line 4). It then selects the configuration m_q by maximizing the UCB in Eq. (2) or by drawing from the posterior in Eq. (3) (Line 5), and the scheduler sets the episode length H_q (Line 6).

COMAS **C**o-**O**ptimizing **M**odel Configuration and **A**ction **S**election in Contextual Bandits

Require: Horizon T, Configuration set $\mathcal{M}$, Episode scheduler, $\mathsf{ACQ} \in \{\mathrm{UCB}, \mathrm{TS}\}$, $\mathfrak{A}$, C_H, and β

1: Run the calibration phase for T_{cal} rounds and add its observations to $\mathcal{D}_{T_{\mathrm{cal}}+1}$
2: Initialize the retained meta history $\mathcal{E}_0$ (calibration feedback may be excluded)
3: **for** $q = 1, 2, \ldots$ until the horizon T is exhausted **do**
4: Choose GP kernel hyperparameters θ_q and residual likelihood noise $\lambda_{\mathrm{n},q}$, construct $\mathbf{\Gamma}_{q-1,q}$, and update GP posterior $(\mu_{q-1}, \sigma_{q-1})$ using $\mathcal{E}_{q-1}$
5: Select configuration m_q using Eq. (2) if $\mathsf{ACQ} = \mathrm{UCB}$, or Eq. (3) if $\mathsf{ACQ} = \mathrm{TS}$
6: Choose episode length $H_q \le C_H(\tau_q + 1)^{\beta}$
7: **for** $t = \tau_q, \ldots, \tau_q + H_q - 1$ **do**
8: Instantiate the base learner's reward estimator specified by m_q and refit it on $\mathcal{D}_t$
9: Observe context c_t and action set $\mathcal{A}_t$, and select $a_t = \mathrm{argmax}_{a \in \mathcal{A}_t} \mathrm{UCB}_t(x_{t,a})$ (Eq. (4))
10: Play action a_t, observe reward y_t, and set $\mathcal{D}_{t+1} = \mathcal{D}_t \cup \{(x_{t,a_t}, y_t)\}$
11: **end for**
12: Compute $\bar{y}_q = \frac{1}{H_q} \sum_{t=\tau_q}^{\tau_q + H_q - 1} y_t$ and set $\mathcal{E}_q = \mathcal{E}_{q-1} \cup \{(m_q, \bar{y}_q, H_q)\}$
13: **end for**

In each round t of the episode, COMAS instantiates the reward estimator used by the contextual bandit algorithm or the base learner $\mathfrak{A}$ according to m_q and refits it on the current shared dataset $\mathcal{D}_t$ (Line 8), allowing the selected configuration to leverage data collected under earlier configurations as well as observations from the current episode. The base learner then observes the context c_t and available action set $\mathcal{A}_t$, selects the action with the highest UCB (Line 9), and adds the resulting observation to the shared dataset (Line 10). At the end of the episode, COMAS computes the episode-level feedback $\bar{y}_q$ and adds $(m_q, \bar{y}_q, H_q)$ to the retained meta history (Line 12). The episode length H_q is retained because the effective likelihood variance of this observation is recomputed at each later GP fit as $\gamma_{q,q'} = \sigma_m^2 / H_q + \lambda_g + \lambda_{\mathrm{n},q'}$. After calibration, retained meta observations are carried forward across episodes and remain available for subsequent GP updates. The same procedure is repeated for subsequent episodes until the horizon T is exhausted.

4 Regret Analysis

In this section, we first introduce the key definitions and assumptions required to establish the regret upper bound for COMAS, followed by the resulting guarantee. The full proof is provided in Sec. B.

Assumption 1 (Vanishing drift)**.** *Each configuration $m \in \mathcal{M}$ has a deterministic limiting value $g(m) \in [0, 1]$. Recall that $f(x)$ is the mean reward at feature vector x. For episode q, let $r_{q,h} := \mathbb{E}[y_{\tau_q+h} \mid \mathcal{J}_{\tau_q+h-1}] = \mathbb{E}[f(x_{\tau_q+h,a_{\tau_q+h}}) \mid \mathcal{J}_{\tau_q+h-1}]$, and define $b_q := H_q^{-1} \sum_{h=0}^{H_q-1} r_{q,h} - g(m_q)$. There exist constants $C_d > 0$ and $\alpha > 0$ such that, with probability at least $1 - \delta_{\mathrm{stab}}$, every analyzed episode satisfies $|b_q| \le d_q := C_d(\tau_q + 1)^{-\alpha}$.*

Since all configurations are refit on the shared interaction dataset $\mathcal{D}_t$, the reward of m_q depends on data collected by earlier configurations, making episode rewards non-stationary in m. Assumption 1 requires only that this non-stationarity vanishes over time: b_q measures the gap between the current pre-round conditional reward and the limiting value of the deployed configuration and decays polynomially.

Definition 1 (Regret components)**.** *Let N denote the number of post-calibration episodes, $m^\star \in \arg\max_{m \in \mathcal{M}} g(m)$, $g^\star := g(m^\star)$, and let $\mathfrak{R}_{\mathrm{cal}}$ denote the calibration regret. Define the pool gap and configuration-selection regret as $M_T^{\mathrm{pool}} := \sum_{t=T_{\mathrm{cal}}+1}^{T} [f(x_{t,a_t^\star}) - g(m^\star)]$ and $\mathfrak{R}_T^g := \sum_{q=1}^{N} H_q [g(m^\star) - g(m_q)]$.*

Here, M_T^{pool} measures the gap between the best stable value available in the configuration pool and the contextual-bandit oracle, while $\mathfrak{R}_T^g$ measures configuration-selection regret relative to the configuration with the largest limiting stable value $g(m)$ in the pool. Appendix B.2 shows that the episode-level feedback satisfies $\bar{y}_q = g(m_q) + b_q + \eta_q$, where η_q combines reward noise with the centered fluctuation of the already-mean reward $f(x_{q,h})$ around $r_{q,h}$ due to not-yet-resolved context/action randomness. It is conditionally sub-Gaussian with variance proxy $(R^2 + 1/4)/H_q$.

Assumption 2 (Learned-kernel regularity)**.** *Let the configuration set $\mathcal{M} = \{m_1, \ldots, m_K\}$ be finite. At the start of episode q, the meta-GP relearns hyperparameters θ_q from a fixed compact kernel family Θ. Let $\boldsymbol{\mu}_\theta$ and K_θ denote the prior mean and Gram matrix on the pool. For some $K_0 \succ 0$ and $\chi \geq 1$, $0 \prec K_\theta \preceq \chi K_0$ for all $\theta \in \Theta$, and $B_m := \sup_{\theta\in\Theta} \|\mathbf{g} - \boldsymbol{\mu}_\theta\|_{K_\theta^{-1}} < \infty$, where $\mathbf{g} := (g(m_1), \ldots, g(m_K))^\top$.*

Here, B_m is the finite-pool analog of the RKHS-norm term in standard GP-UCB analyses (Chowdhury & Gopalan, 2017), while χ controls the variation of the learned kernels relative to the fixed reference kernel K_0. As hyperparameters are relearned between episodes, the analysis uses K_0 to control all posterior variances via the reference information-gain envelope $\overline{\Gamma}_T^{\mathrm{ref}}$, defined in Sec. B.5.

The remaining likelihood, episode-schedule, and optimism conditions used by the finite-grid guarantee are stated in Appendix B. Write $S_T := \sum_{q=1}^N H_q^2$ for the squared episode-length sum.

Theorem 1 (Finite-grid GP-UCB regret)**.** *Under Assumptions 1-4 and the conditional R-sub-Gaussian reward-noise assumption, suppose $\alpha \geq 1/2$. Then, for any $\delta_\eta, \delta_z \in (0,1)$, with probability at least $1 - \delta_{\mathrm{stab}} - \delta_\eta - \delta_z$, $\mathfrak{R}_T(\pi) \leq \mathfrak{R}_{\mathrm{cal}} + M_T^{\mathrm{pool}} + \widetilde{O}\left(\left[B_m\sqrt{\chi\,\overline{\Gamma}_T^{\mathrm{ref}}} + \chi\,\overline{\Gamma}_T^{\mathrm{ref}}\right]\sqrt{S_T} + \sqrt{T}\right)$. Thus, beyond calibration and pool mismatch, the cost of online configuration selection is governed by the reference information gain $\overline{\Gamma}_T^{\mathrm{ref}}$, kernel/prior mismatch (B_m, χ), and the episode-length term S_T. The general $\alpha > 0$ bound is given in Appendix B.5.*

Under the oracle alignment and the end-to-end conditions of Corollary 1, the square-root episode-growth schedule $\beta = 1/2$ gives the explicit rate $\mathfrak{R}_T(\pi) = \widetilde{O}(T^{3/4})$.

Proof outline. We decompose the contextual bandit regret into calibration cost, pool mismatch, configuration selection regret, shared-dataset drift, and martingale noise. GP confidence bounds reduce the configuration-selection term to accumulated posterior widths. Since the kernel and bounded scalar likelihood noise may be relearned between episodes, the proof compares learned kernels to a fixed reference geometry, makes the stochastic-noise event uniform over the likelihood-noise interval, and bounds posterior variance through a count-based lower reference precision. Finally, Assumption 1 and the episode-length schedule in Assumption 3 control the accumulated drift. Combining these bounds yields Theorem 1; the complete proof is given in Sec. B.

Comparison with independent configurations. Under Assumption 1, the diagonal meta-GP reference analyzed in Appendix B.6 yields $\widetilde{O}(\sqrt{KS_T})$ configuration-selection regret. For growing finite pools with pool-uniform regularity constants, our reference-kernel bound has smaller asymptotic dependence on K whenever $\overline{\Gamma}_T^{\mathrm{ref}} = o(\sqrt{K})$. Standard information-gain bounds for Matérn and squared-exponential kernels satisfy this regime under the conditions given in Sec. B.5.

5 Experiments

We evaluate COMAS on contextual bandit problem instances derived from synthetic and real datasets. We first compare it with online model-selection methods and fixed configurations on a finite pool of heterogeneous configurations (Sec. 5.1), and then evaluate its ability to jointly select configurations and tune in a mixed space with continuous hyperparameters (Sec. 5.2).

Configurations. A configuration $m \in \mathcal{M}$ specifies the learner family, its architecture, the regularization λ of its reward estimator, and an exploration scale ζ multiplying the confidence width $h(x, \mathcal{O}_t)$ (Sec. 3.2). The families are linear and polynomial learners with LinUCB (Li et al., 2010) and neural networks of varying width and depth with NeuralUCB (Zhou et al., 2020). In the *discrete* setting, $\mathcal{M}$ is a grid of $K \in [27, 33]$ configurations over all four components; in the *continuous* setting, family and architecture stay discrete while λ and ζ vary over log-scale ranges (Sec. C.1).

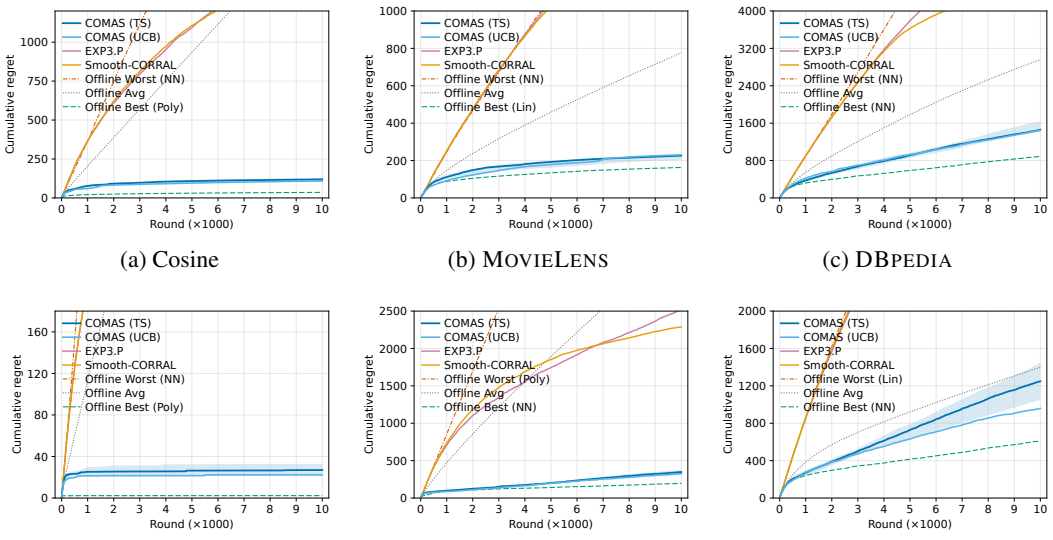
Environments. *Synthetic:* f is a known non-linear function of $x_{t,a}$, either oscillatory (Cosine) or quadratic (Square). *Classification:* an n-class dataset becomes an n-armed bandit with reward 1 for the true label and 0 otherwise, using the disjoint arm encoding $x_{t,a} = e_a \otimes c_t$ (Li et al., 2010). We use SHUTTLE from UCI (Dua & Graff, 2017), MNIST (LeCun et al., 1998) with ImageNet-pretrained ResNet-18 features (He et al., 2016), and DBPEDIA (Zhang et al., 2015) with MiniLM sentence embeddings (Wang et al., 2020; Reimers & Gurevych, 2019). *Recommendation:* on MOVIELENS (Harper & Konstan, 2015), contexts are factors of the factorized user–movie rating matrix and rewards are Bernoulli with mean given by the product of user and movie factors (Sec. C.1).

Baselines and references. We compare with EXP3.P (Auer et al., 2002) and Smooth-CORRAL (Pacchiano et al., 2020b), which treat the K configurations as independent arms. Three offline references run each configuration alone for T rounds: *offline-best* and *offline-worst* (lowest and highest final regret) and *offline-average* (mean over the pool, i.e. committing to a configuration chosen at random). We use these as finite-horizon empirical references for standalone configuration performance; the regret analysis, instead, evaluates online configuration selection via the stable-value comparator $g^\star$ with refitting on shared interaction data.

Protocol. All methods share environments, configuration pools, base learners, and seeds. COMAS refits every selected configuration on the shared interaction data $\mathcal{D}_t$, whereas the baselines refit each configuration only on its own rounds. We report both acquisition rules of COMAS: COMAS (UCB), which is analyzed in Theorem 1, and COMAS (TS) (Eq. (40)). Regret is the cumulative regret $\mathfrak{R}_T(\pi)$ of Eq. (1) over $T = 10^4$ rounds, computed exactly because f is known in every environment (for MOVIELENS, f is the Bernoulli mean). Curves show the median over 10 seeds with the interquartile range (IQR) shaded; final regret (median $\pm$ standard deviation) for every method is in Tab. 3.

5.1 COMPARISON WITH MODEL SELECTION AND FIXED CONFIGURATIONS

Configuration choice matters. Committing to a configuration without knowing the pool is costly: offline-average ends at $2.3\times$ the regret of offline-best on MNIST, $3.4\times$ on DBPEDIA, $4.7\times$ on MOVIELENS, $17\times$ on SHUTTLE, $51\times$ on Cosine, and $455\times$ on Square. No family is best everywhere: the polynomial is best in both synthetic environments but worst on SHUTTLE, the linear is best on MOVIELENS but worst on MNIST, and on DBPEDIA, the best and worst configurations are both neural, differing only in hyperparameters.



(a) Cosine (b) MOVIELENS (c) DBPEDIA

(d) Square (e) SHUTTLE (f) MNIST

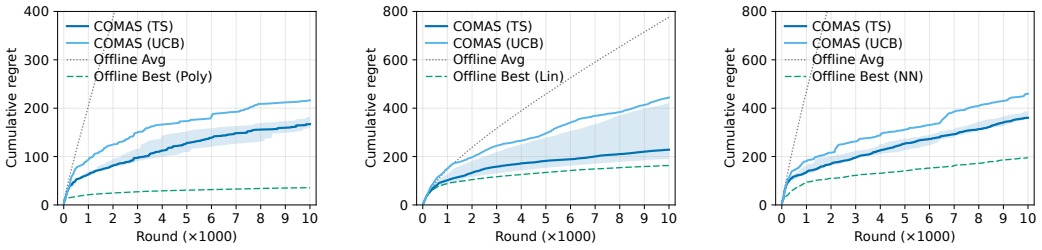
Figure 2: **Comparison with model selection and fixed configurations.** Cumulative regret of COMAS (TS) and COMAS (UCB) against EXP3.P, Smooth-CORRAL and the offline references of the same pool; parentheses give the learner family of offline-best and offline-worst. Vertical axes are truncated to keep low-regret curves legible.

COMAS outperforms the model-selection baselines. On all six environments, both COMAS variants end below EXP3.P, Smooth-CORRAL, and offline-average (Fig. 2). Against Smooth-CORRAL, the stronger baseline, final regret is 3.3–29× lower for COMAS (TS) and 3.3–35× lower for COMAS (UCB). Both baselines end above offline-average on MOVIELENS, MNIST, and DBPEDIA, i.e. they do worse than committing to a random configuration. The margin over the offline average is smallest for COMAS (TS) on MNIST (1249 vs. 1384), where its IQR matches the offline average; COMAS (UCB) ends at 959.

Gap to offline-best. No online method knows offline-best in advance; it is a stringent finite-horizon reference. On the four real-data environments, COMAS ends within 1.4–2.0× of it, whereas committing to a random configuration costs 2.3–17×. In the synthetic environments, the ratio is larger (3.0–11.5×), but the absolute gap is at most 84. Most of the excess regret is incurred early, while configurations are still being distinguished (Sec. C.3). The gap then stays nearly constant on the synthetic environments and grows slowly on the real-data environments. COMAS (UCB) has the lower median on five of six environments, most clearly on MNIST (959 vs. 1249), but a heavier upper tail across seeds (e.g. standard deviation 2506 vs. 165 on DBPEDIA).

5.2 CONTINUOUS CONFIGURATION SPACE

Model-selection methods require a finite pool. Because the GP surrogate of COMAS is defined over the configuration space itself, the same procedure applies when λ and ζ are continuous; this setting is empirical and not covered by Theorem 1. COMAS (TS) stays below offline-average on all six environments (Fig. 3 and Sec. C.4), with final regret within 4% of the discrete run on four of them, lower on MNIST (1085 vs. 1249) and higher on Cosine (167 vs. 119). COMAS (UCB) is higher than in the discrete setting on the three environments shown (e.g. 444 vs. 231 on MOVIELENS). The continuous space thus removes the need to fix a grid in advance, but, being larger to explore, does not lower regret. Syndicated Bandits (Ding et al., 2022), which tunes hyperparameters within one learner family, is compared in Sec. C.5.



(a) Cosine (b) MOVIELENS (c) SHUTTLE

Figure 3: **Continuous configuration space.** Cumulative regret of COMAS (TS) and COMAS (UCB) with continuous regularization and exploration scale. Offline references are those of the discrete pool (Fig. 2). All six environments are in Fig. 4.

6 CONCLUSION

COMAS studies online configuration selection for contextual bandits when candidate configurations differ in learner family, architecture, and hyperparameters while sharing interaction data. The framework supports both discrete and mixed configuration spaces, while our theoretical analysis focuses on the finite-grid setting. For the finite-grid method, we learn a GP surrogate from episode-average rewards and jointly adapt configuration and action selection. The analysis decomposes regret into calibration, pool mismatch, configuration-selection regret, shared-dataset drift, and stochastic fluctuations. Under stability and GP confidence conditions, we obtain a regret guarantee that allows predictable kernel and bounded likelihood-noise relearning, and provide sufficient stability conditions for shared-dataset ridge learners. Empirically, COMAS outperforms online model-selection baselines across heterogeneous configuration pools, while ablations highlight the importance of shared-data refitting and support the played-path stability assumption. We further demonstrate the method in continuous and mixed configuration spaces. Extending the analysis to nonlinear learners, continuous configuration spaces, and sharper guarantees for covariance sharing remains future work.

AI USE STATEMENT

Generative AI tools were used to assist with drafting, editing, code review, and mathematical consistency checks. All AI-assisted text, code changes, and derivations were reviewed by the authors, who take responsibility for the final content and claims.

ETHICS STATEMENT

This work is theoretical, focusing on the design and analysis of algorithms for contextual bandits. The proposed methods do not directly involve human subjects, personal data, or real-world deployments. While the framework could potentially be applied in systems that interact with users, we emphasize that ethical considerations, such as fairness, privacy, and informed consent, must be addressed in practical deployments. Our primary goal is to advance the theoretical understanding of contextual bandit algorithms, and we do not anticipate any negative societal impacts.

REPRODUCIBILITY STATEMENT

This paper presents theoretical results, including formal proofs of regret upper bounds. All assumptions, definitions, and derivations are stated explicitly in the main text (see Sec. 3 and the Appendix). The details of our experimental setup are provided in Sec. 5 and the Appendix. Additionally, the code used in our experiments has been included in the supplementary material, enabling full reproduction of the results reported in this paper.

REFERENCES

Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. In *Proc. NeurIPS*, pp. 2312–2320, 2011.

Yasin Abbasi-Yadkori, Aldo Pacchiano, and My Phan. Regret balancing for bandit and RL model selection. *arXiv:2006.05491*, 2020.

Marc Abeille and Alessandro Lazaric. Linear Thompson sampling revisited. In *Proc. AISTATS*, pp. 176–184, 2017.

Alekh Agarwal, Haipeng Luo, Behnam Neyshabur, and Robert E. Schapire. Corralling a band of bandit algorithms. In *Proc. COLT*, pp. 12–38, 2017.

Shipra Agrawal and Navin Goyal. Thompson sampling for contextual bandits with linear payoffs. In *Proc. ICML*, pp. 127–135, 2013.

Andres Alban, Stephen E. Chick, and Spyros I. Zoumpoulis. Learning personalized treatment strategies with predictive and prognostic covariates in adaptive clinical trials. *Management Science*, 72(7):5943–5962, 2026.

Raman Arora, Teodor V. Marinov, and Mehryar Mohri. Corralling stochastic bandit algorithms. In *Proc. AISTATS*, pp. 2116–2124, 2021.

Peter Auer, Nicolò Cesa-Bianchi, Yoav Freund, and Robert E. Schapire. The nonstochastic multiarmed bandit problem. *SIAM Journal on Computing*, 32(1):48–77, 2002.

Maryam Aziz, Emilie Kaufmann, and Marie-Karelle Riviere. On multi-armed bandit designs for dose-finding trials. *Journal of Machine Learning Research*, 22(14):1–38, 2021.

Wang Chi Cheung, David Simchi-Levi, and Ruihao Zhu. Learning to optimize under non-stationarity. In *Proc. AISTATS*, pp. 1079–1087, 2019.

Sayak Ray Chowdhury and Aditya Gopalan. On kernelized multi-armed bandits. In *Proc. ICML*, pp. 844–853, 2017.

Wei Chu, Lihong Li, Lev Reyzin, and Robert E. Schapire. Contextual bandits with linear payoff functions. In *Proc. AISTATS*, pp. 208–214, 2011.

Ashok Cutkosky, Christoph Dann, Abhimanyu Das, Claudio Gentile, Aldo Pacchiano, and Manish Purohit. Dynamic balancing for model selection in bandits and RL. In *Proc. ICML*, pp. 2276–2285, 2021.

Qin Ding, Yue Kang, Yi-Wei Liu, Thomas C. M. Lee, Cho-Jui Hsieh, and James Sharpnack. Syndicated bandits: A framework for auto tuning hyper-parameters in contextual bandit algorithms. In *Proc. NeurIPS*, pp. 1170–1181, 2022.

Dheeru Dua and Casey Graff. UCI machine learning repository. `http://archive.ics.uci.edu/ml`, 2017.

David Eriksson, Michael Pearce, Jacob Gardner, Ryan D. Turner, and Matthias Poloczek. Scalable global optimization via local Bayesian optimization. In *Proc. NeurIPS*, pp. 5496–5507, 2019.

Stefan Falkner, Aaron Klein, and Frank Hutter. BOHB: Robust and efficient hyperparameter optimization at scale. In *Proc. ICML*, pp. 1437–1446, 2018.

Sarah Filippi, Olivier Cappé, Aurélien Garivier, and Csaba Szepesvári. Parametric bandits: The generalized linear case. In *Proc. NeurIPS*, pp. 586–594, 2010.

Dylan J. Foster, Akshay Krishnamurthy, and Haipeng Luo. Model selection for contextual bandits. In *Proc. NeurIPS*, pp. 14714–14725, 2019.

Dylan J. Foster, Claudio Gentile, Mehryar Mohri, and Julian Zimmert. Adapting to misspecification in contextual bandits. In *Proc. NeurIPS*, pp. 11478–11489, 2020.

Tong Geng, Xiliang Lin, Harikesh S. Nair, Jun Hao, Bin Xiang, and Shurui Fan. Comparison Lift: Bandit-based experimentation system for online advertising. In *Proc. AAAI*, pp. 15117–15126, 2021.

F. Maxwell Harper and Joseph A. Konstan. The MovieLens datasets: History and context. *ACM Transactions on Interactive Intelligent Systems*, 5(4):1–19, 2015.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proc. CVPR*, pp. 770–778, 2016.

Kwang-Sung Jun, Aniruddha Bhargava, Robert Nowak, and Rebecca Willett. Scalable generalized linear bandits: Online computation and hashing. In *Proc. NeurIPS*, pp. 99–109, 2017.

Yue Kang, Cho-Jui Hsieh, and Thomas C. M. Lee. Online continuous hyperparameter optimization for generalized linear contextual bandits. *Transactions on Machine Learning Research*, 2024.

Parnian Kassraie and Andreas Krause. Neural contextual bandits without regret. In *Proc. AISTATS*, pp. 240–278, 2022.

Johannes Kirschner and Andreas Krause. Information directed sampling and bandits with heteroscedastic noise. In *Proc. COLT*, pp. 358–384, 2018.

Tor Lattimore and Csaba Szepesvári. *Bandit Algorithms*. Cambridge University Press, 2020.

Tor Lattimore, Csaba Szepesvári, and Gellért Weisz. Learning with good feature representations in bandits and in RL with a generative model. In *Proc. ICML*, pp. 5662–5670, 2020.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Lihong Li, Wei Chu, John Langford, and Robert E. Schapire. A contextual-bandit approach to personalized news article recommendation. In *Proc. WWW*, pp. 661–670, 2010.

Lihong Li, Yu Lu, and Dengyong Zhou. Provably optimal algorithms for generalized linear contextual bandits. In *Proc. ICML*, pp. 2071–2080, 2017.

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.

Anastasiia Makarova, Ilnura Usmanova, Ilija Bogunovic, and Andreas Krause. Risk-averse heteroscedastic Bayesian optimization. In *Proc. NeurIPS*, pp. 17235–17245, 2021.

Aldo Pacchiano, Christoph Dann, Claudio Gentile, and Peter Bartlett. Regret bound balancing and elimination for model selection in bandits and RL. *arXiv:2012.13045*, 2020a.

Aldo Pacchiano, My Phan, Yasin Abbasi-Yadkori, Anup Rao, Julian Zimmert, Tor Lattimore, and Csaba Szepesvári. Model selection in contextual stochastic bandit problems. In *Proc. NeurIPS*, pp. 10328–10337, 2020b.

Jack Parker-Holder, Vu Nguyen, and Stephen J. Roberts. Provably efficient online hyperparameter optimization with population-based bandits. In *Proc. NeurIPS*, pp. 17200–17211, 2020.

Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proc. EMNLP*, pp. 3982–3992, 2019.

Amr Sharaf and Hal Daumé III. Meta-learning for contextual bandit exploration. *arXiv:1901.08159*, 2019.

Aleksandrs Slivkins. Introduction to multi-armed bandits. *Foundations and Trends® in Machine Learning*, 12(1-2):1–286, 2019.

Niranjan Srinivas, Andreas Krause, Sham Kakade, and Matthias Seeger. Gaussian process optimization in the bandit setting: No regret and experimental design. In *Proc. ICML*, pp. 1015–1022, 2010.

Sattar Vakili, Kia Khezeli, and Victor Picheny. On information gain and regret bounds in Gaussian process bandits. In *Proc. AISTATS*, pp. 82–90, 2021.

Michal Valko, Nathan Korda, Rémi Munos, Ilias Flaounas, and Nello Cristianini. Finite-time analysis of kernelised contextual bandits. In *Proc. UAI*, pp. 654–663, 2013.

Yogatheesan Varatharajah and Brent Berry. A contextual-bandit-based approach for informed decision-making in clinical trials. *Life*, 12(8):1277, 2022.

Arun Verma, Zhongxiang Dai, Xiaoqiang Lin, Patrick Jaillet, and Bryan Kian Hsiang Low. Neural dueling bandits: Preference-based optimization with human feedback. In *Proc. ICLR*, 2025.

Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. MiniLM: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. In *Proc. NeurIPS*, pp. 5776–5788, 2020.

Jiayu Yao, Emma Brunskill, Weiwei Pan, Susan Murphy, and Finale Doshi-Velez. Power constrained bandits. In *Proc. MLHC*, pp. 209–259, 2021.

Hsiang-Fu Yu, Cho-Jui Hsieh, Si Si, and Inderjit Dhillon. Scalable coordinate descent approaches to parallel matrix factorization for recommender systems. In *Proc. ICDM*, pp. 765–774, 2012.

Weitong Zhang, Dongruo Zhou, Lihong Li, and Quanquan Gu. Neural Thompson sampling. In *Proc. ICLR*, 2021a.

Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. In *Proc. NeurIPS*, pp. 649–657, 2015.

Zihan Zhang, Jiaqi Yang, Xiangyang Ji, and Simon S. Du. Improved variance-aware confidence sets for linear bandits and linear mixture MDP. In *Proc. NeurIPS*, pp. 4342–4355, 2021b.

Dongruo Zhou, Lihong Li, and Quanquan Gu. Neural contextual bandits with UCB-based exploration. In *Proc. ICML*, pp. 11492–11502, 2020.

A Additional related work

Contextual bandits and model misspecification. Contextual bandits (Slivkins, 2019; Lattimore & Szepesvári, 2020) study sequential decision-making under partial feedback and have been widely applied to recommendation (Li et al., 2010), advertising, and healthcare. Under well-specified linear rewards, LinUCB and Linear Thompson Sampling use ridge-regression confidence sets to achieve sub-linear regret (Li et al., 2010; Chu et al., 2011; Abbasi-Yadkori et al., 2011; Agrawal & Goyal, 2013; Abeille & Lazaric, 2017; Zhang et al., 2021b). Generalized linear bandits extend these guarantees to exponential-family link functions (Filippi et al., 2010; Jun et al., 2017; Li et al., 2017), while kernel bandits replace the linear class with a reproducing kernel Hilbert space and obtain rates in terms of the maximum information gain (Srinivas et al., 2010; Valko et al., 2013; Chowdhury & Gopalan, 2017). Neural bandits use deep networks as reward models, with confidence bounds often derived from neural tangent kernel arguments (Zhou et al., 2020; Zhang et al., 2021a; Kassraie & Krause, 2022), the same machinery also underlies neural dueling bandits with preference feedback (Verma et al., 2025). These methods fix the reward-model class and its hyperparameters before interaction. Richer model classes impose stronger structural assumptions, and misspecification can substantially degrade regret (Foster et al., 2020; Lattimore et al., 2020). In contrast, COMAS treats the model configuration, including the learner family, architecture, and regularization, as a sequential decision variable, co-optimizing it with the resulting action-selection policy.

B Leftover Proofs

B.1 Terminology and assumptions

This appendix proves Theorem 1 and develops supporting analyses of information gain, covariance sharing, shared-dataset stability, and end-to-end regret. The main proof proceeds in three stages. First, Lemmas 1 and 2 derive the episodic observation model and exact regret decomposition. Second, Lemmas 3 and 4 reduce configuration-selection regret to posterior uncertainty and separate GP uncertainty from the episode schedule. Third, Lemmas 6–10 control this uncertainty through a fixed reference kernel despite kernel and bounded scalar likelihood-noise relearning, while Lemma 5 controls the schedule-dependent terms. The remaining results form supporting branches: Lemma 11 and the diagonal and finite-rank analyses study information gain and covariance sharing; Lemma 14 through Proposition 5 give sufficient routes to Assumptions 1; and Proposition 6 controls pool mismatch to obtain the end-to-end corollary.

Terminology. We use three terms repeatedly throughout the analysis.

Definition 2 (Pool mismatch)**.** *The* pool mismatch *is the regret from restricting configuration selection to the available finite pool rather than comparing directly with the contextual-bandit oracle.*

Definition 3 (Configuration-selection regret)**.** *The* configuration-selection regret *is the regret from selecting suboptimal configurations within the available pool, relative to the best stable configuration value in that pool.*

Definition 4 (Stability drift)**.** *The* stability drift *is the deviation between an episode-level reward and the limiting value of its configuration, caused by refitting the base learner on the growing shared dataset.*

Thus, pool mismatch reflects the quality of the candidate pool, configuration-selection regret the cost of the meta-controller, and stability drift the evolution of the policy induced by a fixed configuration.

Technical assumptions for the finite-grid guarantee. The following conditions specify the likelihood, episode schedule, and optimism requirements used in Theorem 1.

Assumption 3 (Noise model and episode length)**.** *Before episode q, the meta-GP may relearn an $\mathcal{H}_q$-measurable scalar residual likelihood-noise term $\lambda_{\mathrm{n},q} \in [\underline{\lambda}_{\mathrm{n}}, \overline{\lambda}_{\mathrm{n}}]$, where $0 \le \underline{\lambda}_{\mathrm{n}} \le \overline{\lambda}_{\mathrm{n}} < \infty$. For every retained episode observation $q' < q$, the current GP fit uses $\gamma_{q',q} := \sigma_m^2/H_{q'} + \lambda_g + \lambda_{\mathrm{n},q}$ and $w_{q',q} := \gamma_{q',q}^{-1}$, where $\sigma_m^2, \lambda_g > 0$. Assume $C_\eta^2 \sigma_m^2 \ge R^2 + 1/4$ for a fixed $C_\eta > 0$. Episode lengths satisfy $H_q \le C_H(\tau_q + 1)^\beta$ for $C_H \ge 1$ and $\beta \in [0, 1)$.*

The $H_{q'}^{-1}$ term reflects episode averaging, while λ_g is a fixed positive variance floor. The bounded scalar $\lambda_{\mathrm{n},q}$ may be relearned from the retained meta rewards, as in the implementation. Appendix B.5 makes the concentration event uniform over its admissible interval and uses a count-based lower reference precision for the variance budget; the resulting covering factor is only logarithmic and is absorbed by $\widetilde{O}(\cdot)$. The episode-length schedule implies $S_T := \sum_{q=1}^{N} H_q^2 = O(T^{1+\beta})$.

Assumption 4 (Optimism)**.** *We select model configurations using* COMAS (UCB) *as defined in Eq. (2). Let $\bar{\rho}_T$ denote the common upper bound on the GP confidence radius over all post-calibration episodes, as constructed in Sec. B.5. For some fixed $C_u \geq 1$, $\bar{\rho}_T \leq \alpha_q \leq C_u \bar{\rho}_T$ for every $q \geq 1$.*

The lower bound ensures optimistic coverage of the limiting configuration values, while the upper bound controls over-exploration through the selection-regret term. This is the theory-matched GP-UCB condition; a fixed empirical exploration coefficient is not by itself guaranteed to satisfy it.

B.2 Episode observation model and exact decomposition

Let T be the total horizon, T_{cal} the calibration-prefix length, and $T' := T - T_{\mathrm{cal}}$ the number of post-calibration rounds analyzed below. At round t, let c_t be the observed context, $\mathcal{A}_t$ the feasible action set, a_t the selected action, and $x_{t,a}$ the feature vector associated with action a. The observed reward is $y_t = f(x_{t,a_t}) + \varepsilon_t$, where $f(x) \in [0,1]$ is the mean reward conditional on the selected feature vector and ε_t is the remaining reward noise.

Let $\mathcal{J}_{t-1}$ denote the information available before round t, and let $\mathcal{G}_t$ denote the information after x_{t,a_t} is fixed but before ε_t is observed. The conditional R-sub-Gaussian reward-noise assumption is, for every $\lambda \in \mathbb{R}$,

$$\mathbb{E}\big[e^{\lambda \varepsilon_t} \mid \mathcal{G}_t\big] \leq e^{\lambda^2 R^2/2}. \tag{5}$$

After calibration, index the analyzed episodes by $q = 1, \ldots, N$. Episode q starts at round τ_q, deploys configuration $m_q \in \mathcal{M}$ for H_q rounds, and is chosen from the episode-start history $\mathcal{H}_q := \mathcal{J}_{\tau_q - 1}$; both m_q and H_q are $\mathcal{H}_q$-measurable. For $h = 0, \ldots, H_q - 1$, write $t_{q,h} := \tau_q + h$, $x_{q,h} := x_{t_{q,h},a_{t_{q,h}}}$, and $y_{q,h} := y_{t_{q,h}}$.

Define the pre-round conditional reward $r_{q,h} := \mathbb{E}[y_{q,h} \mid \mathcal{J}_{t_{q,h}-1}] = \mathbb{E}[f(x_{q,h}) \mid \mathcal{J}_{t_{q,h}-1}]$. Since f is already the mean reward conditional on the selected feature vector, this second expectation averages only the context/action randomness unresolved before round $t_{q,h}$. Let $z_{q,h} := f(x_{q,h}) - r_{q,h}$ be the corresponding centered path fluctuation, and let $\eta_q := H_q^{-1} \sum_{h=0}^{H_q-1} (z_{q,h} + \varepsilon_{t_{q,h}})$ be the centered episode-level noise. The meta-controller receives the episode average $\bar{y}_q := H_q^{-1} \sum_{h=0}^{H_q-1} y_{q,h}$. Under Assumptions 1, $b_q := H_q^{-1} \sum_h r_{q,h} - g(m_q)$ is the signed stability drift of the played episode and $|b_q| \leq d_q$ on the stability event $\mathcal{E}_{\mathrm{stab}}$.

Lemma 1 (Episode-average meta observation)**.** *The episode-average reward is a noisy observation of the stable value of the selected configuration. The centered path fluctuation satisfies $\mathbb{E}[z_{q,h} \mid \mathcal{J}_{t_{q,h}-1}] = 0$, and averaging H_q rounds reduces the sub-Gaussian variance proxy of the combined path fluctuation and reward noise by a factor H_q:*

$$\mathbb{E}\big[e^{\lambda \eta_q} \mid \mathcal{H}_q\big] \leq \exp\left[\frac{\lambda^2}{2H_q}\left(R^2 + \frac{1}{4}\right)\right]. \tag{6}$$

Consequently, the meta feedback is $\bar{y}_q = g(m_q) + b_q + \eta_q$, separating stable configuration value, stability drift, and centered episode noise.

Proof. Although f is already the mean-reward function, the random variable $f(x_{q,h})$ still varies with the unresolved context/action path. Since $f(x_{q,h}) \in [0,1]$, its centered fluctuation around $r_{q,h}$ contributes a $1/4$ sub-Gaussian variance proxy, while $\varepsilon_{t_{q,h}}$ contributes R^2. The two centered terms form a martingale-difference sequence, and averaging H_q rounds divides the variance proxy by H_q.

Because $f(x_{q,h}) \in [0,1]$, Hoeffding's lemma gives

$$\mathbb{E}\big[e^{\lambda z_{q,h}} \mid \mathcal{J}_{t_{q,h}-1}\big] \leq e^{\lambda^2/8}. \tag{7}$$

Moreover $z_{q,h}$ is $\mathcal{G}_{t_{q,h}}$-measurable, so iterated expectation with Eq. (5) yields

$$\mathbb{E}\Big[e^{\lambda(z_{q,h}+\varepsilon_{t_{q,h}})} \mid \mathcal{J}_{t_{q,h}-1}\Big] \le \exp\left[\frac{\lambda^2}{2}\left(R^2+\frac{1}{4}\right)\right].$$

Since (m_q, H_q) is $\mathcal{H}_q$-measurable, iterating this bound over the H_q rounds proves Eq. (6). Finally,

$$\bar{y}_q = \frac{1}{H_q}\sum_h r_{q,h} + \eta_q = g(m_q) + b_q + \eta_q$$

by the definition of b_q in Assumptions 1. □

Lemma 2 (Exact regret decomposition)**.** *Using the main-text regret components, let* $B_T^{\mathrm{play}} := \sum_{q=1}^N H_q b_q$ *be the signed cumulative stability drift along the played episodes, let* $D_T^{\mathrm{play}} := \sum_{q=1}^N H_q d_q$ *be its played-path drift budget obtained by summing the per-episode Assumptions 1 bounds, and let* $Z_T := \sum_{q=1}^N \sum_{h=0}^{H_q-1} z_{q,h}$ *be the cumulative centered path fluctuation. Thus* $|B_T^{\mathrm{play}}| \le D_T^{\mathrm{play}}$ *on* $\mathcal{E}_{\mathrm{stab}}$*. The realized contextual-bandit regret has the pathwise decomposition* $\mathfrak{R}_T(\pi) = \mathfrak{R}_{\mathrm{cal}} + M_T^{\mathrm{pool}} + \mathfrak{R}_T^g - B_T^{\mathrm{play}} - Z_T$*. Moreover, under Assumption 1, with probability at least* $1-\delta_z$ *over the martingale term* Z_T*,*

$$\mathfrak{R}_T(\pi) \le \mathfrak{R}_{\mathrm{cal}} + M_T^{\mathrm{pool}} + \mathfrak{R}_T^g + D_T^{\mathrm{play}} + \sqrt{\frac{T'}{2}\log\frac{1}{\delta_z}}.$$

Proof. Add and subtract $H_q g(m_q)$ within each episode to decompose the original regret into four terms: pool mismatch, configuration-selection regret, stability drift, and a martingale-difference term. The martingale-difference term is then bounded using Hoeffding's inequality.

For episode q,

$$\sum_{h=0}^{H_q-1} f(x_{q,h}) = H_q g(m_q) + H_q b_q + \sum_{h=0}^{H_q-1} z_{q,h}.$$

Substituting this expression into the definition of $\mathfrak{R}_T(\pi)$ and using the definitions of M_T^{pool} and $\mathfrak{R}_T^g$ in Assumptions 1 give the pathwise decomposition stated above. On Assumptions 1, $|B_T^{\mathrm{play}}| \le D_T^{\mathrm{play}}$. Finally, iterating Eq. (7) over the T' analyzed rounds gives

$$\Pr\left(-Z_T \le \sqrt{\frac{T'}{2}\log\frac{1}{\delta_z}}\right) \ge 1-\delta_z,$$

which proves the high-probability upper bound stated in the lemma. □

Lemma 2 is the bridge between the original contextual bandit objective and the meta problem. Even a perfect meta-controller cannot remove M_T^{pool} if the candidate pool itself misses the contextual oracle; conversely, the GP proof below only needs to control $\mathfrak{R}_T^g$ and the stability drift.

B.3 UCB SELECTION REDUCTION AND EPISODE UNCERTAINTY

We now reduce configuration-selection regret to the posterior uncertainty of the meta-GP along the configurations actually played. Choose $m^\star \in \arg\max_{m\in\mathcal{M}} g(m)$, so $g(m^\star) = g^\star$, and let $\mu_{q-1}(m)$ and $\sigma_{q-1}(m)$ be the meta-GP posterior mean and latent standard deviation before episode q. Let ρ_q denote the confidence radius required in episode q and α_q the UCB exploration coefficient. We write $\mathcal{E}_{\mathrm{conf}}$ for the simultaneous confidence event $|\mu_{q-1}(m) - g(m)| \le \rho_q \sigma_{q-1}(m)$ for every analyzed q and $m \in \mathcal{M}$.

Define $\mathcal{P}_T := \sum_{q=1}^N H_q \sigma_{q-1}(m_q)$ as the episode-length-weighted posterior width accumulated along the played configurations, and let $U_N := \max_{q\le N} \alpha_q$ be the largest UCB coefficient. Finally, $\mathcal{B} := \{q : \alpha_q < \rho_q\}$ denotes the episodes on which the chosen UCB coefficient does not cover the confidence radius.

Lemma 3 (UCB selection reduces regret to posterior width)**.** *Under simultaneous GP confidence, UCB optimism converts the stable-value gap of the selected configuration into posterior width. On $\mathcal{E}_{\mathrm{conf}}$, if $\alpha_q \geq \rho_q$, then*

$$g(m^\star) - g(m_q) \leq 2\alpha_q \sigma_{q-1}(m_q). \tag{8}$$

Summing with episode lengths gives

$$\mathfrak{R}_T^g \leq 2U_N \mathcal{P}_T. \tag{9}$$

If coverage fails on the episodes in $\mathcal{B}$, those episodes contribute at most their lengths, so $\mathfrak{R}_T^g \leq \sum_{q\in\mathcal{B}} H_q + 2U_N\mathcal{P}_T$.

Proof. Optimism upper-bounds the value of $m^\star$ by its UCB. The selected configuration has at least as large a UCB, and confidence then converts that UCB back to $g(m_q)$ plus one more width term.

On $\mathcal{E}_{\mathrm{conf}}$ and for $\alpha_q \geq \rho_q$,

$$g(m^\star) \leq \mu_{q-1}(m^\star) + \alpha_q\sigma_{q-1}(m^\star) \leq \mu_{q-1}(m_q) + \alpha_q\sigma_{q-1}(m_q) \leq g(m_q) + 2\alpha_q\sigma_{q-1}(m_q),$$

which proves Eq. (8). Multiply by H_q and sum to obtain Eq. (9). If $q \in \mathcal{B}$, use the trivial bound $g(m^\star) - g(m_q) \leq 1$ instead. □

Lemma 4 (Episode uncertainty reduction)**.** *Let $S_T := \sum_{q=1}^N H_q^2$ be the squared episode-length sum, which measures the cost of committing one meta decision for an entire episode, and let $V_N := \sum_{q=1}^N \sigma_{q-1}^2(m_q)$ be the posterior-variance sum along the played configurations. The accumulated posterior width then separates as $\mathcal{P}_T \leq \sqrt{S_T V_N}$, isolating the schedule contribution in S_T from the GP uncertainty contribution in V_N.*

Proof. This is Cauchy–Schwarz applied to the two sequences $(H_q)_q$ and $(\sigma_{q-1}(m_q))_q$. It isolates all schedule dependence in S_T and all GP uncertainty in V_N. Hence, we have

$$\mathcal{P}_T = \sum_q H_q\sigma_{q-1}(m_q) \leq \left(\sum_q H_q^2\right)^{1/2} \left(\sum_q \sigma_{q-1}^2(m_q)\right)^{1/2}. \qquad \square$$

B.4 Episode-schedule sums

Lemma 5 (Schedule and drift sums)**.** *Under the episode-length condition in Assumptions 3, the squared episode lengths obey $S_T = O(T^{1+\beta})$. For $a > 0$, let $\mathcal{D}_a(T)$ denote the usual power/harmonic envelope for summing $(t+1)^{-a}$:*

$$\mathcal{D}_a(T) := \begin{cases} T^{1-a}, & 0 < a < 1, \\ \log(T+1), & a = 1, \\ 1, & a > 1. \end{cases}$$

Then $\sum_q H_q(\tau_q+1)^{-a} = O(\mathcal{D}_a(T))$, and the Assumptions 1 stability-decay condition gives $D_T^{\mathrm{play}} = O(\mathcal{D}_\alpha(T))$. These bounds isolate the schedule growth and cumulative stability drift used in the main regret bound.

Proof. The bound on S_T follows from $\sum_q H_q = T'$ and the largest episode length. For the drift sum, expand each episode into its constituent rounds and compare τ_q with the round indices inside that episode.

Let $H_{\max} := \max_q H_q$. Then $H_{\max} \leq C_H(T+1)^\beta$ and

$$S_T \leq H_{\max} \sum_q H_q = O(T^{1+\beta}).$$

Because $\beta < 1$, all sufficiently late episodes satisfy $H_q \leq \tau_q + 1$. Hence, for $t = \tau_q + h$, $t+1 \leq 2(\tau_q+1)$ and

$$H_q(\tau_q+1)^{-a} \leq 2^a \sum_{h=0}^{H_q-1} (\tau_q + h + 1)^{-a}.$$

Summing over episodes reduces the problem to the standard harmonic/power sum $\sum_{t\le T}(t+1)^{-a} = O(\mathcal{D}_a(T))$. The finitely many early episodes contribute only a constant. Multiplying by C_d proves the last claim. □

B.5 Predictably learned ARD GP under a dominating reference kernel

This subsection proves Theorem 1. The main technical issue is that the GP kernel used by the algorithm may change after every episode. We therefore never telescope determinants of the learned kernels. Instead, all adaptive posteriors are compared with one fixed reference kernel, which acts as a common geometric "ruler" throughout the proof.

Regularity conditions for learned-kernel family. The learned GP may change between episodes, so the proof needs one uniform set of regularity constants that covers every predictable kernel and mean selected by the fitting rule. The following quantities make that comparison precise. Let $\mathcal{Z}$ be the descriptor domain and $z(m) \in \mathcal{Z}$ the descriptor of configuration m. A compact set $\Theta \subset \mathbb{R}^{d_\Theta}$ indexes prior mean functions μ_θ and positive-definite kernels k_θ on $\mathcal{Z}$. Write $\boldsymbol{\mu}_\theta$ and K_θ for their restrictions to $\mathcal{M} = \{m_1, \dots, m_K\}$. Before episode q, the fitting rule chooses $\theta_q = \Psi_q(\mathcal{H}_q) \in \Theta$, with $\boldsymbol{\mu}_q^0 := \boldsymbol{\mu}_{\theta_q}$ and $K_q := K_{\theta_q}$. The update may be marginal-likelihood optimization or any other predictable rule. Assume $\theta \mapsto (\boldsymbol{\mu}_\theta, K_\theta)$ is continuous, $K_\theta \succ 0$ on the analyzed pool for every θ, and the mean family is bounded there. Compactness then gives $\inf_{\theta\in\Theta} \lambda_{\min}(K_\theta) > 0$.

Fix a deterministic reference matrix $K_0 \succ 0$ and define

$$\chi := \max\left\{1, \sup_{\theta\in\Theta} \lambda_{\max}\left(K_0^{-1/2} K_\theta K_0^{-1/2}\right)\right\} < \infty.$$

Then $0 \prec K_\theta \preceq \chi K_0$ for every $\theta \in \Theta$. Define

$$B_m := \sup_{\theta\in\Theta} \|\mathbf{g} - \boldsymbol{\mu}_\theta\|_{K_\theta^{-1}}, \qquad \mathbf{g} := (g(m_1), \dots, g(m_K))^\top, \qquad \kappa_0^2 := \max_j (K_0)_{jj}.$$

Here B_m is one global meta-function mismatch radius; its subscript does not index an individual configuration.

The finite-pool theorem only requires these finite-dimensional quantities to be finite. When comparing a *sequence* of increasingly dense finite pools, one additionally needs them to remain bounded uniformly in the pool size. A sufficient descriptor-space condition is that the pool values are restrictions of one function $\mathrm{g} : \mathcal{Z} \to \mathbb{R}$, that K_0 is the restriction of a bounded reference kernel k_0, and that, for constants $B_\mathcal{H}$ and χ independent of the pool,

$$\sup_{\theta\in\Theta} \|\mathrm{g} - \mu_\theta\|_{\mathcal{H}_{k_\theta}} \le B_\mathcal{H}, \qquad \chi k_0 - k_\theta \succeq_{\ker} 0, \qquad \sup_{z\in\mathcal{Z}} k_0(z,z) < \infty.$$

The minimum-norm interpolation property gives $B_m \le B_\mathcal{H}$ on every finite restriction, and the kernel order gives the same χ on every pool. Thus the growing-pool comparison does not hide an additional $\sqrt{K}$ factor inside B_m.

For the additive family/continuous kernel used in the finite backend, a simple sufficient condition for finite-pool invertibility is $K_\theta = K_{\mathrm{fam},\theta} + K_{\mathrm{cont},\theta}$ with positive semidefinite summands, pairwise-distinct continuous descriptor vectors over the *entire* pool, and a strictly positive-definite continuous subkernel. Distinct full descriptors alone are not sufficient if the continuous part is repeated across families. For stationary Matérn subkernels with fixed smoothness and length-scales/output scales in compact positive intervals, pool-uniform domination can be verified by a uniform spectral-density ratio bound. These assumptions guarantee finite regularity constants; they do not claim those constants are numerically small.

Bounded learned scalar likelihood noise. The finite-grid backend may relearn one scalar residual likelihood-noise term before each episode. Let $\lambda_{\mathrm{n},q} \in [\underline{\lambda}_\mathrm{n}, \overline{\lambda}_\mathrm{n}]$, with $0 \le \underline{\lambda}_\mathrm{n} \le \overline{\lambda}_\mathrm{n} < \infty$, be $\mathcal{H}_q$-measurable. For a fixed test value $\lambda \in [\underline{\lambda}_\mathrm{n}, \overline{\lambda}_\mathrm{n}]$ and a retained observation q', define its effective variance $\gamma_{q'}(\lambda) := \sigma_m^2/H_{q'} + \lambda_g + \lambda$ and precision $w_{q'}(\lambda) := \gamma_{q'}(\lambda)^{-1}$. Thus the actual fit before episode q uses $\gamma_{q',q} := \gamma_{q'}(\lambda_{\mathrm{n},q})$ and $w_{q',q} := w_{q'}(\lambda_{\mathrm{n},q})$ for every retained $q' < q$. Assume $C_\eta > 0$ satisfies

$$C_\eta^2 \sigma_m^2 \ge R^2 + \frac{1}{4}. \tag{10}$$

The additional $1/4$ is not extra reward noise: it controls the centered fluctuation of the mean reward $f(x_{q,h})$ around the pre-round conditional reward $r_{q,h}$ caused by unresolved context/action randomness. Uniformly over the admissible likelihood-noise interval, the precision is bounded between $w_- := 1/(\sigma_m^2 + \lambda_g + \overline{\lambda}_{\mathrm{n}})$ and $w_+ := 1/(\lambda_g + \underline{\lambda}_{\mathrm{n}})$, so $w_- \le w_{q'}(\lambda) \le w_+$. By Lemma 1, for every fixed λ,

$$\mathbb{E}\Big[e^{t\sqrt{w_{q'}(\lambda)}\,\eta_{q'}} \mid \mathcal{H}_{q'}\Big] \le e^{t^2 C_\eta^2/2} \qquad (t \in \mathbb{R}). \tag{11}$$

The scalar $\lambda_{\mathrm{n},q}$ may be obtained by marginal-likelihood optimization on the retained meta data. Its reward dependence is handled below by making the concentration event uniform over the whole admissible interval.

Aggregated full-history posterior. Because the configuration pool is finite, all past observations can be grouped by configuration. For episode q, $\ell_{q,j}$ below is the total current likelihood precision accumulated from earlier observations of configuration m_j, while D_q collects these coordinate-wise precisions into a diagonal matrix. Define

$$\ell_{q,j} := \sum_{q'<q:m_{q'}=m_j} w_{q',q}, \qquad D_q := \mathrm{diag}(\ell_{q,1},\dots,\ell_{q,K}),$$

with Moore–Penrose pseudoinverse $D_q^\dagger$. The vectors s_q^b and s_q^η are the corresponding precision-weighted accumulations of stability drift and centered episode noise:

$$(s_q^b)_j := \sum_{q'<q:m_{q'}=m_j} w_{q',q} b_{q'},$$
$$(s_q^\eta)_j := \sum_{q'<q:m_{q'}=m_j} w_{q',q} \eta_{q'}.$$

Let e_j be the jth coordinate vector and $j(q')$ the pool index of the configuration played in episode q', so $m_{q'} = m_{j(q')}$. The precision-weighted observation statistic is

$$s_q^y := \sum_{q'<q} w_{q',q} \bar{y}_{q'} e_{j(q')} = D_q \mathbf{g} + s_q^b + s_q^\eta.$$

Set $A_q := K_q^{-1} + D_q$ and $\Sigma_q := A_q^{-1}$. The finite-grid latent posterior is

$$\boldsymbol{\mu}_{q-1} := \Sigma_q(K_q^{-1}\boldsymbol{\mu}_q^0 + s_q^y),$$
$$\mu_{q-1}(m_j) := e_j^\top \boldsymbol{\mu}_{q-1}, \qquad \sigma_{q-1}^2(m_j) := e_j^\top \Sigma_q e_j.$$

Substitution gives the exact posterior-error decomposition

$$\boldsymbol{\mu}_{q-1} - \mathbf{g} = -\Sigma_q K_q^{-1}(\mathbf{g} - \boldsymbol{\mu}_q^0) + \Sigma_q s_q^b + \Sigma_q s_q^\eta. \tag{12}$$

The three terms are, respectively, prior mismatch, stability drift, and stochastic episode noise.

Reference information gain. For a fixed $\lambda \in [\underline{\lambda}_{\mathrm{n}}, \overline{\lambda}_{\mathrm{n}}]$, define

$$D_q(\lambda) := \sum_{q'<q} w_{q'}(\lambda) e_{j(q')} e_{j(q')}^\top, \qquad B_q(\lambda) := K_0^{-1} + D_q(\lambda),$$

and

$$\Gamma_q^{(1)}(\lambda) := \frac{1}{2}\log\frac{\det B_q(\lambda)}{\det K_0^{-1}}.$$

The actual noise geometry before episode q is $B_q := B_q(\lambda_{\mathrm{n},q})$.

For the variance budget, let $C_q := \sum_{q'<q} e_{j(q')} e_{j(q')}^\top$ be the configuration visit-count matrix: each retained episode adds one rank-one coordinate matrix for the configuration it played. Define the lower-reference precision $\underline{B}_q := K_0^{-1} + \chi w_- C_q$, which assigns every past observation the worst-case lower precision w_-, and let $\underline{\Gamma}_q := \frac{1}{2}\log[\det \underline{B}_q / \det K_0^{-1}]$ be its log-determinant information gain. Unlike the current likelihood weights, C_q changes by exactly one rank-one term after every retained episode.

Let $\overline{N}_T$ be any deterministic upper bound on the number of analyzed episodes, $N \le \overline{N}_T \le T'$. Taking $\overline{N}_T = T'$ is always valid; for a data-independent power schedule $H_q \asymp (\tau_q + 1)^\beta$, one may take $\overline{N}_T = O(T^{1-\beta})$. Define

$$\overline{\Gamma}_T^{\mathrm{ref}} := \max_{\substack{0 \le n \le \overline{N}_T \\ j_1, \dots, j_n \in \{1, \dots, K\}}} \frac{1}{2} \log\det(I_n + \chi w_+ K_0[j_{1:n}, j_{1:n}]), \tag{13}$$

where repeated indices are allowed and the value at $n = 0$ is zero.

Lemma 6 (Reference information-gain envelope)**.** *For every $q \le N + 1$ and every $\lambda \in [\underline{\lambda}_{\mathrm{n}}, \overline{\lambda}_{\mathrm{n}}]$,*

$$\Gamma_q^{(1)}(\lambda) \le \overline{\Gamma}_T^{\mathrm{ref}}, \qquad \underline{\Gamma}_q \le \overline{\Gamma}_T^{\mathrm{ref}}.$$

Proof. For fixed λ, $w_{q'}(\lambda) \le w_+$, so

$$D_q(\lambda) \preceq w_+ \sum_{q' < q} e_{j(q')} e_{j(q')}^\top.$$

PSD monotonicity of $\log\det(I + \cdot)$ and Sylvester's determinant identity therefore give the first inequality; the factor $\chi \ge 1$ in Eq. (13) only enlarges the envelope. The second follows in the same way from $w_- \le w_+$ and the definition of $\underline{B}_q$. □

Lemma 7 (Reference self-normalized noise event)**.** *The stochastic-noise bound can be made simultaneous over every admissible value of the learned scalar likelihood noise. Let*

$$\bar{\sigma}_\eta^2 := R^2 + \frac{1}{4}, \qquad K_\star := \|K_0\|_{\mathrm{op}}, \qquad M_{\eta,T} := \sqrt{2\bar{\sigma}_\eta^2 \log\left(\frac{4\overline{N}_T}{\delta_\eta}\right)}.$$

Define

$$L_T^{\mathrm{lik}} := 1 + 2\overline{N}_T^2 K_\star w_+^3 M_{\eta,T}^2 + \overline{N}_T^3 K_\star^2 w_+^4 M_{\eta,T}^2,$$
$$J_T^{\mathrm{lik}} := 1 + \left\lceil (\overline{\lambda}_{\mathrm{n}} - \underline{\lambda}_{\mathrm{n}}) L_T^{\mathrm{lik}} \right\rceil,$$

and

$$\beta_{\eta,T}^2 := 2C_\eta^2 \left(\overline{\Gamma}_T^{\mathrm{ref}} + \log \frac{2J_T^{\mathrm{lik}}}{\delta_\eta} \right) + 1.$$

Here $M_{\eta,T}$ is a high-probability envelope for the largest episode noise, L_T^{lik} is a uniform Lipschitz bound in the scalar likelihood parameter, J_T^{lik} is the resulting one-dimensional grid size, and $\beta_{\eta,T}$ is the final uniform stochastic-noise radius. Then there is an event $\mathcal{E}_\eta$ with $\Pr(\mathcal{E}_\eta) \ge 1 - \delta_\eta$ such that, simultaneously for every $q \le N + 1$ and every $\lambda \in [\underline{\lambda}_{\mathrm{n}}, \overline{\lambda}_{\mathrm{n}}]$,

$$\left\| \sum_{q' < q} w_{q'}(\lambda) \eta_{q'} e_{j(q')} \right\|_{B_q(\lambda)^{-1}} \le \beta_{\eta,T}. \tag{14}$$

In particular, Eq. (14) holds at the random $\lambda = \lambda_{\mathrm{n},q}$ selected from the retained meta data, so $\|s_q^\eta\|_{B_q^{-1}} \le \beta_{\eta,T}$. Moreover, $\log J_T^{\mathrm{lik}} = O(\log T)$ for fixed problem constants.

Proof. First fix a deterministic $\lambda \in [\underline{\lambda}_{\mathrm{n}}, \overline{\lambda}_{\mathrm{n}}]$. Set

$$x_{q'}(\lambda) := \sqrt{w_{q'}(\lambda)}\, e_{j(q')}, \qquad \xi_{q'}(\lambda) := \sqrt{w_{q'}(\lambda)}\, \eta_{q'}.$$

Because λ is fixed and $(H_{q'}, m_{q'})$ is chosen before $\eta_{q'}$ is realized, $x_{q'}(\lambda)$ is predictable. By Eq. (11), $\xi_{q'}(\lambda)$ is conditionally C_η-sub-Gaussian. The standard vector self-normalized inequality therefore gives, simultaneously over all q,

$$Q_q(\lambda) := \left\| \sum_{q' < q} w_{q'}(\lambda) \eta_{q'} e_{j(q')} \right\|_{B_q(\lambda)^{-1}}^2 \le 2C_\eta^2 \left(\Gamma_q^{(1)}(\lambda) + \log \frac{1}{\delta} \right)$$

with probability at least $1-\delta$.

It remains to make this event uniform in λ. By Lemma 1 and a union bound,

$$\max_{q' \le N} |\eta_{q'}| \le M_{\eta,T}$$

with probability at least $1-\delta_\eta/2$. On this event, $|w'_{q'}(\lambda)| = w_{q'}(\lambda)^2 \le w_+^2$. If $n = q-1 \le \overline{N}_T$, then

$$\left|\frac{d}{d\lambda} Q_q(\lambda)\right| \le 2n^2 K_\star w_+^3 M_{\eta,T}^2 + n^3 K_\star^2 w_+^4 M_{\eta,T}^2 \le L_T^{\mathrm{lik}}.$$

Here we used $B_q(\lambda) \succeq K_0^{-1}$, hence $\|B_q(\lambda)^{-1}\|_{\mathrm{op}} \le K_\star$, together with $\frac{d}{d\lambda} B_q(\lambda)^{-1} = -B_q(\lambda)^{-1} D'_q(\lambda) B_q(\lambda)^{-1}$.

Cover $[\underline{\lambda}_{\mathrm{n}}, \overline{\lambda}_{\mathrm{n}}]$ by J_T^{lik} deterministic grid points with mesh at most $(L_T^{\mathrm{lik}})^{-1}$. Apply the fixed-$\lambda$ self-normalized inequality at each grid point with failure probability $\delta_\eta/(2J_T^{\mathrm{lik}})$ and union-bound over the grid. By Lemma 6, every grid-point determinant term is at most $\overline{\Gamma}_T^{\mathrm{ref}}$. The Lipschitz bound then increases Q_q by at most 1 between a point and its nearest grid point, proving Eq. (14). Since $\overline{N}_T \le T$ and $M_{\eta,T}^2 = O(\log T)$, the covering number is polynomial in T, so $\log J_T^{\mathrm{lik}} = O(\log T)$. □

Lemma 8 (Learned-kernel confidence under reference domination)**.** *This lemma combines prior mismatch, stability drift, and stochastic episode noise into a single confidence interval for the adaptively relearned GP. The fixed reference kernel is used only to control the noise term and compare it back to the learned posterior standard deviation. On $\mathcal{E}_{\mathrm{stab}} \cap \mathcal{E}_\eta$, simultaneously for all $q \le N+1$ and $m \in \mathcal{M}$, $|\mu_{q-1}(m) - g(m)| \le \rho_q \sigma_{q-1}(m)$, where $B_{b,q} := \|s_q^b\|_{D_q^\dagger}$ and $\rho_q := B_m + B_{b,q} + \sqrt{\chi}\,\beta_{\eta,T}$.*

Proof. Take one coordinate of Eq. (12). The three error terms are controlled in three geometries: K_q^{-1} for prior mismatch, D_q for drift, and B_q for noise. Reference domination converts the last geometry back to the learned posterior standard deviation.

Fix (q, j) and set $v := \Sigma_q e_j$. Because $\Sigma_q(K_q^{-1} + D_q)\Sigma_q = \Sigma_q$,

$$v^\top K_q^{-1} v + v^\top D_q v = e_j^\top \Sigma_q e_j = \sigma_{q-1}^2(m_j).$$

Taking coordinate j in Eq. (12) gives

$$\mu_{q-1}(m_j) - g(m_j) = -v^\top K_q^{-1}(\mathbf{g} - \boldsymbol{\mu}_q^0) + v^\top s_q^b + v^\top s_q^\eta.$$

The prior and drift terms satisfy

$$|v^\top K_q^{-1}(\mathbf{g} - \boldsymbol{\mu}_q^0)| \le B_m \sigma_{q-1}(m_j),$$
$$|v^\top s_q^b| \le B_{b,q} \sigma_{q-1}(m_j),$$

by Cauchy–Schwarz and $s_q^b \in \mathrm{Range}(D_q)$. From $K_q \preceq \chi K_0$ we obtain $B_q \preceq \chi A_q$. Hence

$$|v^\top s_q^\eta| \le \sqrt{v^\top B_q v}\, \|s_q^\eta\|_{B_q^{-1}} \le \sqrt{\chi}\, \sigma_{q-1}(m_j) \beta_{\eta,T}.$$

Add the three bounds in Eq. (12). □

Lemma 9 (Drift contribution to the confidence radius)**.** *The accumulated signed stability bias is controlled by its current weighted drift energy, where $\Delta_{b,q} := (\sum_{q'<q} w_{q',q} d_{q'}^2)^{1/2}$. On $\mathcal{E}_{\mathrm{stab}}$, $B_{b,q} \le \Delta_{b,q}$, and this energy satisfies*

$$\sup_q \Delta_{b,q}^2 = O(\mathcal{D}_{2\alpha}(T)). \tag{15}$$

Consequently, there is a fixed $C_b < \infty$ such that

$$\bar{\rho}_T := B_m + C_b \sqrt{\mathcal{D}_{2\alpha}(T)} + \sqrt{\chi}\,\beta_{\eta,T}$$

satisfies $\rho_q \le \bar{\rho}_T$ for all q.

Proof. Within each configuration coordinate, weighted Cauchy–Schwarz converts the accumulated signed bias into its weighted energy. The current likelihood weight satisfies $w_{q',q} \le H_{q'}/\sigma_m^2$, so the remaining sum is exactly the episode drift sum from Lemma 5 with exponent 2α.

For each coordinate j,

$$\frac{((s_q^b)_j)^2}{\ell_{q,j}} \le \sum_{q'<q:m_{q'}=m_j} w_{q',q} b_{q'}^2.$$

Summing over j proves $B_{b,q}^2 \le \sum_{q'<q} w_{q',q} b_{q'}^2$. Since $w_{q',q} \le H_{q'}/\sigma_m^2$ and $|b_{q'}| \le d_{q'} \le C_d(\tau_{q'}+1)^{-\alpha}$, Lemma 5 gives Eq. (15). Substitute this result into Lemma 8. □

Lemma 10 (Reference-kernel posterior-variance budget)**.** *Let $L_- := \log(1+\chi w_- \kappa_0^2) > 0$ be the fixed chord constant used to convert log-determinant gain back into posterior variance. Reference domination converts the fixed lower-reference information gain into a budget on the learned posterior variance accumulated along the played configurations:*

$$V_N = \sum_{q=1}^{N} \sigma_{q-1}^2(m_q) \le \frac{2\chi\kappa_0^2}{L_-} \overline{\Gamma}_T^{\mathrm{ref}}.$$

Combining this with Lemma 4 gives

$$\mathcal{P}_T \le \left(\frac{2\chi\kappa_0^2}{L_-}\right)^{1/2} \sqrt{\overline{\Gamma}_T^{\mathrm{ref}} S_T}.$$

Proof. The current likelihood fit changes the weights of all retained observations, so we do not telescope the realized precision D_q. Instead, the deterministic lower bound $w_{q',q} \ge w_-$ gives

$$D_q \succeq w_- C_q.$$

Together with $K_q \preceq \chi K_0$,

$$A_q = K_q^{-1} + D_q \succeq \chi^{-1} K_0^{-1} + w_- C_q = \chi^{-1} \underline{B}_q,$$

and therefore

$$\Sigma_q \preceq \chi \underline{B}_q^{-1}.$$

If $m_q = m_{j_q}$, set $x_q^\star := e_{j_q}^\top \underline{B}_q^{-1} e_{j_q}$. Then

$$\sigma_{q-1}^2(m_q) \le \chi x_q^\star, \qquad 0 \le x_q^\star \le \kappa_0^2.$$

The count-based reference precision has the exact rank-one update

$$\underline{B}_{q+1} = \underline{B}_q + \chi w_- e_{j_q} e_{j_q}^\top.$$

Hence the matrix determinant lemma gives

$$\log \frac{\det \underline{B}_{q+1}}{\det \underline{B}_q} = \log(1+\chi w_- x_q^\star).$$

For $0 \le x \le \kappa_0^2$, concavity of $x \mapsto \log(1+\chi w_- x)$ gives

$$\log(1+\chi w_- x) \ge \frac{x}{\kappa_0^2} L_-.$$

Summing and telescoping yields

$$V_N \le \frac{2\chi\kappa_0^2}{L_-} \underline{\Gamma}_{N+1} \le \frac{2\chi\kappa_0^2}{L_-} \overline{\Gamma}_T^{\mathrm{ref}},$$

where the last step is Lemma 6. Apply Lemma 4 for the final claim. □

Theorem 1 (Finite-grid GP-UCB regret)**.** *Under Assumptions 1-4 and the conditional R-sub-Gaussian reward-noise assumption, suppose $\alpha \ge 1/2$. Then, for any $\delta_\eta, \delta_z \in (0,1)$, with probability at least $1 - \delta_{\mathrm{stab}} - \delta_\eta - \delta_z$, $\mathfrak{R}_T(\pi) \le \mathfrak{R}_{\mathrm{cal}} + M_T^{\mathrm{pool}} + \widetilde{O}\left(\left[B_m\sqrt{\chi\overline{\Gamma}_T^{\mathrm{ref}}} + \chi\overline{\Gamma}_T^{\mathrm{ref}}\right]\sqrt{S_T} + \sqrt{T}\right)$. Thus, beyond calibration and pool mismatch, the cost of online configuration selection is governed by the reference information gain $\overline{\Gamma}_T^{\mathrm{ref}}$, kernel/prior mismatch (B_m, χ), and the episode-length term S_T. The general $\alpha > 0$ bound is given in Appendix B.5.*

Proof. Episode-level performance changes for two reasons: the meta-controller may deploy different configurations across episodes, and the policy of a fixed configuration may change because it is repeatedly refitted on the growing shared dataset. Lemma 2 separates these effects into pool mismatch, configuration-selection regret, shared-dataset drift, and a martingale fluctuation, in addition to calibration. The GP argument controls configuration-selection regret: Lemma 9 gives a simultaneous confidence radius, Lemma 3 reduces UCB regret to posterior width, and Lemma 10 bounds the accumulated width using the fixed reference kernel. The stability and schedule bounds control the remaining drift term, after which the decomposition returns to the original contextual-bandit regret.

On $\mathcal{E}_{\mathrm{stab}} \cap \mathcal{E}_\eta$, $\rho_q \le \bar{\rho}_T$. Under Assumptions 4, Lemmas 3 and 10 give

$$\mathfrak{R}_T^g \le 2C_u \left[B_m + C_b\sqrt{\mathcal{D}_{2\alpha}(T)} + \sqrt{\chi}\,\beta_{\eta,T}\right] \times \left(\frac{2\chi\kappa_0^2}{L_-}\right)^{1/2} \sqrt{\overline{\Gamma}_T^{\mathrm{ref}}\, S_T}.$$

Combining this with Lemmas 2 and 5 yields, for general $\alpha > 0$,

$$\begin{aligned}\mathfrak{R}_T(\pi) \le\ & \mathfrak{R}_{\mathrm{cal}} + M_T^{\mathrm{pool}} \\ & + \widetilde{O}\left(\left[B_m\sqrt{\chi\overline{\Gamma}_T^{\mathrm{ref}}} + \sqrt{\mathcal{D}_{2\alpha}(T)}\sqrt{\chi\overline{\Gamma}_T^{\mathrm{ref}}} + \chi\overline{\Gamma}_T^{\mathrm{ref}}\right]\sqrt{S_T} + \mathcal{D}_\alpha(T) + \sqrt{T}\right).\end{aligned}$$

If $\alpha \ge 1/2$, then $\mathcal{D}_{2\alpha}(T)$ is at most logarithmic and $\mathcal{D}_\alpha(T) = \widetilde{O}(\sqrt{T})$, giving Theorem 1. □

Why kernel and likelihood relearning do not break the argument. The algorithm may relearn both K_q and the bounded scalar $\lambda_{\mathrm{n},q}$. The proof never telescopes either learned object. Predictable kernel changes are controlled by the fixed dominating matrix K_0. Reward-dependent likelihood relearning is handled by making the self-normalized event uniform over $[\underline{\lambda}_{\mathrm{n}}, \overline{\lambda}_{\mathrm{n}}]$, while the variance budget telescopes the count-based lower reference $\underline{B}_q = K_0^{-1} + \chi w_- C_q$. Thus old observation weights may change between GP fits without changing the stated $\widetilde{O}$ rate; the extra one-dimensional covering factor contributes only logarithmic terms.

Finite-pool and descriptor-space information gain. A simple cardinality bound is

$$\overline{\Gamma}_T^{\mathrm{ref}} \le \frac{K}{2}\log\left(1 + \frac{\chi w_+ \overline{N}_T \kappa_0^2}{K}\right), \tag{16}$$

obtained by applying AM–GM to the nonzero eigenvalues after a trace bound. This makes explicit that the meta information gain depends on the number of meta observations, not directly on the number of base-bandit rounds.

The finite-pool bound does not use smoothness of the descriptor kernel. The next result gives a second branch that does. To avoid confusion with the likelihood variance, write $\mathcal{I}_n(k;\mathcal{Z})$ for the standard maximum information gain of a kernel k after n observations:

$$\mathcal{I}_n(k;\mathcal{Z}) := \sup_{z_1,\dots,z_n \in \mathcal{Z}} \frac{1}{2}\log\det(I_n + K_k[z_{1:n}, z_{1:n}])\,.$$

Lemma 11 (Descriptor-space information-gain branch)**.** *The finite-pool information-gain bound depends explicitly on pool cardinality. This complementary branch instead bounds the same reference complexity through the descriptor-space kernel, exposing when smooth covariance sharing can yield a smaller complexity term. Suppose the reference matrix K_0 is the restriction of a bounded kernel k_0 on the descriptor domain $\mathcal{Z}$. Then*

$$\overline{\Gamma}_T^{\mathrm{ref}} \le \min\left\{\frac{K}{2}\log\left(1 + \frac{\chi w_+ \overline{N}_T \kappa_0^2}{K}\right),\ \mathcal{I}_{\overline{N}_T}(\chi w_+ k_0; \mathcal{Z})\right\}. \tag{17}$$

If, as in the finite backend, the reference kernel is chosen with additive family/continuous structure

$$k_0 = k_{\mathrm{fam},0} + k_{\mathrm{cont},0},$$

the family label takes only M values, and the family-kernel diagonal is bounded, then

$$\mathcal{I}_n(\chi w_+ k_0; \mathcal{Z}) \le O\big(M\log(n+1)\big) + \mathcal{I}_n(\chi w_+ k_{\mathrm{cont},0}; \mathcal{V}), \tag{18}$$

where $\mathcal{V}$ is the continuous-descriptor domain. If $\mathcal{V}$ is a compact subset of $\mathbb{R}^{d_v}$, the second term has the standard rates $\widetilde{O}(n^{d_v/(2\nu+d_v)})$ for a Matérn-ν kernel (Vakili et al., 2021) and $O((\log(n+1))^{d_v+1})$ for a squared-exponential kernel (Srinivas et al., 2010). For a single-family pool, the family contribution is absent.

Proof. The first branch is Eq. (16). For the second, enlarge the finite candidate pool to the whole descriptor domain and maximize over all possible observation locations. For an additive kernel, information gain is subadditive across the two positive-semidefinite kernel components. The family component has rank at most M, while the continuous component uses the standard smooth-kernel information-gain bounds.

Every sequence of configurations allowed in Eq. (13) is also a sequence of points in $\mathcal{Z}$, so

$$\overline{\Gamma}_T^{\mathrm{ref}} \le \mathcal{I}_{\overline{N}_T}(\chi w_+ k_0; \mathcal{Z}).$$

Combining this with Eq. (16) proves Eq. (17). For positive-semidefinite Gram matrices A and B,

$$\log\det(I + A + B) \le \log\det(I + A) + \log\det(I + B),$$

which gives Eq. (18). Because $k_{\mathrm{fam},0}$ is evaluated on only M distinct family labels, its Gram matrix has rank at most M; a trace bound and AM–GM therefore give $O(M\log(n+1))$. The stated Matérn and squared-exponential rates are the standard maximum-information-gain bounds for the continuous kernel. A fixed positive scaling by χw_+ changes only their constants. □

Across growing pools, comparing either branch with a diagonal reference is meaningful only when B_m, χ, and the diagonal mismatch constant remain pool-uniformly bounded. Under this condition, the second branch shows that $\overline{\Gamma}_T^{\mathrm{ref}} = o(\sqrt{K})$ is a non-vacuous asymptotic regime when a fixed compact descriptor domain is discretized increasingly finely. This is an asymptotic pool-size comparison and does not imply a crossover at any particular finite K.

B.6 Diagonal meta-kernel reference: no covariance sharing

This reference uses the same episode observations, bounded learned scalar likelihood noise, retention rule, and UCB selector as the full GP, but treats configurations as independent at the meta level. It is therefore a comparison of *cross-configuration covariance sharing*, not a comparison of whether base learners share the dataset $\mathcal{D}_t$.

For the current pre-episode fit, use the same weights $w_{q',q}$ as in Appendix B.5. For configuration m_j, let $\ell_{q,j} := \sum_{q'<q:m_{q'}=m_j} w_{q',q}$ be its accumulated current precision and $S_{q,j} := \sum_{q'<q:m_{q'}=m_j} w_{q',q}\eta_{q'}$ its corresponding precision-weighted noise sum. For a fixed test value $\lambda \in [\underline{\lambda}_{\mathrm{n}}, \overline{\lambda}_{\mathrm{n}}]$, write $\ell_{q,j}(\lambda) := \sum_{q'<q:m_{q'}=m_j} w_{q'}(\lambda)$ and $S_{q,j}(\lambda) := \sum_{q'<q:m_{q'}=m_j} w_{q'}(\lambda)\eta_{q'}$.

Set $K_q = \mathrm{diag}(\kappa_{q,1}^2, \dots, \kappa_{q,K}^2)$ with $0 < \underline{\kappa} \le \kappa_{q,j} \le \overline{\kappa} < \infty$, and assume $\sup_{q,j} |g(m_j) - \mu_{q,j}^0|/\kappa_{q,j} \le B_0 < \infty$. Let $a_{q,j} := \kappa_{q,j}^{-2} + \ell_{q,j}$ be the scalar posterior precision at configuration m_j; its posterior variance is $a_{q,j}^{-1}$ and

$$\mu_{q-1}(m_j) - g(m_j) = a_{q,j}^{-1}\left[-\kappa_{q,j}^{-2}(g(m_j) - \mu_{q,j}^0) + (s_q^b)_j + S_{q,j}\right]. \tag{19}$$

Lemma 12 (Diagonal simultaneous confidence)**.** *With a diagonal meta-kernel, each configuration is estimated from its own observations. Let*

$$M_{\eta,T}^{\mathrm{diag}} := \sqrt{2\bar{\sigma}_\eta^2 \log\left(\frac{4\overline{N}_T}{\delta_{\mathrm{diag}}}\right)},$$

where $\bar{\sigma}_\eta^2 := R^2 + 1/4$, and define

$$L_T^{\mathrm{diag,lik}} := 1 + 2\overline{N}_T^2\overline{\kappa}^2 w_+^3 (M_{\eta,T}^{\mathrm{diag}})^2 + \overline{N}_T^3\overline{\kappa}^4 w_+^4 (M_{\eta,T}^{\mathrm{diag}})^2,$$
$$J_T^{\mathrm{diag,lik}} := 1 + \left\lceil (\overline{\lambda}_{\mathrm{n}} - \underline{\lambda}_{\mathrm{n}}) L_T^{\mathrm{diag,lik}} \right\rceil,$$

and

$$(\beta_{\eta,T}^{\mathrm{diag}})^2 := 2C_\eta^2 \left[\frac{1}{2} \log\bigl(1 + \overline{\kappa}^2 w_+ \overline{N}_T\bigr) + \log \frac{2K J_T^{\mathrm{diag,lik}}}{\delta_{\mathrm{diag}}} \right] + 1.$$

Here $M_{\eta,T}^{\mathrm{diag}}$ bounds the largest episode noise, $L_T^{\mathrm{diag,lik}}$ controls variation with the scalar likelihood parameter, $J_T^{\mathrm{diag,lik}}$ is the corresponding grid size, and $\beta_{\eta,T}^{\mathrm{diag}}$ is the simultaneous diagonal noise radius. Then, with probability at least $1 - \delta_{\mathrm{diag}}$, simultaneously for all q, j,

$$|\mu_{q-1}(m_j) - g(m_j)| \le \rho_q^{\mathrm{diag}} a_{q,j}^{-1/2}, \qquad \rho_q^{\mathrm{diag}} := B_0 + \Delta_{b,q} + \beta_{\eta,T}^{\mathrm{diag}}.$$

Consequently, $\rho_q^{\mathrm{diag}} = \widetilde{O}(1 + \sqrt{\mathcal{D}_{2\alpha}(T)})$.

Proof. Because the diagonal model separates coordinates, the stochastic term can be controlled one configuration at a time. For fixed λ, apply the scalar self-normalized inequality to $S_{q,j}(\lambda)$ with initial precision $\overline{\kappa}^{-2}$. Since $\ell_{q,j}(\lambda) \le \overline{N}_T w_+$, this gives the displayed log-determinant term. The one-dimensional likelihood parameter is then handled by the same deterministic covering argument as in Lemma 7.

More explicitly, on the event $\max_{q' \le N} |\eta_{q'}| \le M_{\eta,T}^{\mathrm{diag}}$, define

$$Q_{q,j}(\lambda) := \frac{S_{q,j}(\lambda)^2}{\overline{\kappa}^{-2} + \ell_{q,j}(\lambda)}.$$

Using $|w'_{q'}(\lambda)| \le w_+^2$ gives

$$\left| \frac{d}{d\lambda} Q_{q,j}(\lambda) \right| \le L_T^{\mathrm{diag,lik}}.$$

A grid of $J_T^{\mathrm{diag,lik}}$ points and a union bound over the K coordinates therefore imply, simultaneously for every q, j and every admissible λ,

$$\frac{|S_{q,j}(\lambda)|}{\sqrt{\overline{\kappa}^{-2} + \ell_{q,j}(\lambda)}} \le \beta_{\eta,T}^{\mathrm{diag}}.$$

Evaluate this at $\lambda = \lambda_{\mathrm{n},q}$. Since $\kappa_{q,j}^{-2} \ge \overline{\kappa}^{-2}$,

$$|S_{q,j}| \le \beta_{\eta,T}^{\mathrm{diag}} \sqrt{a_{q,j}}.$$

The prior term in Eq. (19) is at most $B_0 a_{q,j}^{-1/2}$, while weighted Cauchy–Schwarz gives the drift term at most $\Delta_{b,q} a_{q,j}^{-1/2}$. This proves the confidence bound. Finally, $\log J_T^{\mathrm{diag,lik}} = O(\log T)$ and Lemma 9 gives the stated rate. □

Lemma 13 (Diagonal posterior-variance budget)**.** *Without cross-configuration covariance sharing, uncertainty at one configuration can shrink only when that configuration is visited. The resulting variance budget therefore scales with the number of pool elements. The diagonal posterior variances satisfy*

$$\sum_{q=1}^{N} \sigma_{q-1}^2(m_q) \le K\overline{\kappa}^2 + \frac{K}{w_-} [1 + \log(N+1)] = \widetilde{O}(K). \tag{20}$$

Proof. The first visit to a configuration costs at most its prior variance. After it has been visited $s-1$ times, every current likelihood weight is at least w_-, so the current accumulated precision is at least $(s-1)w_-$ and the remaining variances form a harmonic series.

For the first visit to coordinate j, $a_{q,j}^{-1} \le \overline{\kappa}^2$. Before its sth visit, $s \ge 2$, $\ell_{q,j} \ge (s-1)w_-$ and hence $a_{q,j}^{-1} \le [(s-1)w_-]^{-1}$. Summing this harmonic bound over the visits to each of the K coordinates gives Eq. (20). □

Proposition 1 (Diagonal configuration-selection rate)**.** *Combining the diagonal confidence and variance budgets gives the corresponding configuration-selection rate. This provides the no-covariance-sharing reference against which the full meta-GP is compared. Under the same coverage condition as in Lemma 3,*

$$\mathfrak{R}_T^g = \widetilde{O}\Big(\sqrt{KS_T}\,[1+\sqrt{\mathcal{D}_{2\alpha}(T)}]\Big).$$

If $\alpha \geq 1/2$*, then*

$$\mathfrak{R}_T^g = \widetilde{O}(\sqrt{KS_T}).$$

Proof. Combine the diagonal confidence radius with the UCB reduction, then apply Cauchy–Schwarz and Lemma 13.

Lemma 3 gives $\mathfrak{R}_T^g \leq 2U_N\mathcal{P}_T$, while Lemma 4 and Lemma 13 give $\mathcal{P}_T = \widetilde{O}(\sqrt{KS_T})$. Lemma 12 supplies the stated radius dependence. When $\alpha \geq 1/2$, $\mathcal{D}_{2\alpha}(T)$ is at most logarithmic. □

If $g(m) \in [0,1]$, the diagonal prior means are uniformly bounded, and $\underline{\kappa}$ is fixed, then B_0 is bounded independently of K. This is the diagonal-side regularity condition used in the growing-pool comparison in the main text.

B.7 Illustrative finite-rank covariance-sharing model

This result is not needed for Theorem 1. It gives a simple finite-dimensional example in which covariance sharing has an explicit rank-p complexity. The feature map below is assumed known, so this is an interpretive model rather than a claim about the learned Matérn kernel used in the experiments.

Proposition 2 (Finite-rank adaptive covariance sharing)**.** *This proposition isolates covariance sharing in a finite-dimensional model with known features. Its complexity is governed by feature rank rather than pool cardinality, making the sharing mechanism explicit without being required by the main theorem. Assume Assumptions 1, cumulative retention, Eq. (10), the episode schedule in Assumptions 3, the bounded learned scalar likelihood-noise model of Appendix B.5, and the UCB rule Eq. (2).*

Suppose there are an integer $p \geq 1$*, a known feature map* $\varphi : \mathcal{M} \to \mathbb{R}^p$*, an unknown stable-value parameter* $\vartheta^\star \in \mathbb{R}^p$*, and a uniform feature bound* $L_\varphi < \infty$ *such that*

$$g(m) = \varphi(m)^\top \vartheta^\star, \qquad \|\varphi(m)\|_2 \leq L_\varphi.$$

Before episode q*, let the predictable matrix* $M_q \in \mathbb{R}^{p\times p}$ *specify the feature-space prior covariance and let* $\vartheta_{0,q} \in \mathbb{R}^p$ *be its predictable prior center. Assume both are* $\mathcal{H}_q$*-measurable, with*

$$\underline{m} I \preceq M_q \preceq \overline{m} I, \qquad \|\vartheta^\star - \vartheta_{0,q}\|_{M_q^{-1}} \leq B_\varphi.$$

Use $\mu_q^0(m) := \varphi(m)^\top \vartheta_{0,q}$ *and* $k_q(m,m') := \varphi(m)^\top M_q \varphi(m')$*.*

For fixed $\lambda \in [\underline{\lambda}_{\mathrm{n}}, \overline{\lambda}_{\mathrm{n}}]$*, define the feature-space reference precision*

$$G_q(\lambda) := \overline{m}^{-1} I + \sum_{q'<q} w_{q'}(\lambda)\varphi(m_{q'})\varphi(m_{q'})^\top,$$

and define the count-based lower-reference precision

$$\underline{G}_q := \overline{m}^{-1} I + w_- \sum_{q'<q} \varphi(m_{q'})\varphi(m_{q'})^\top.$$

Set $\Lambda_N := \log[\det \underline{G}_{N+1} / \det(\overline{m}^{-1} I)]$*, the realized log-determinant complexity of this lower reference. Then*

$$\Lambda_N \leq p \log\left(1 + \frac{\overline{m} w_- N L_\varphi^2}{p}\right) \leq \overline{\Lambda}_T, \tag{21}$$

where

$$\overline{\Lambda}_T := p \log\left(1 + \frac{\overline{m} w_+ \overline{N}_T L_\varphi^2}{p}\right).$$

Let

$$M_{\eta,T}^{\mathrm{str}} := \sqrt{2\bar{\sigma}_\eta^2 \log\left(\frac{4\overline{N}_T}{\delta_\varphi}\right)},$$

where $\bar{\sigma}_\eta^2 := R^2 + 1/4$*, and define*

$$L_T^{\mathrm{str,lik}} := 1 + 2\overline{N}_T^2 \overline{m}\, w_+^3 L_\varphi^2 (M_{\eta,T}^{\mathrm{str}})^2 + \overline{N}_T^3 \overline{m}^2 w_+^4 L_\varphi^4 (M_{\eta,T}^{\mathrm{str}})^2,$$
$$J_T^{\mathrm{str,lik}} := 1 + \left\lceil (\overline{\lambda}_{\mathrm{n}} - \underline{\lambda}_{\mathrm{n}}) L_T^{\mathrm{str,lik}} \right\rceil,$$

and

$$(\beta_{\eta,T}^{\mathrm{str}})^2 := C_\eta^2 \left(\overline{\Lambda}_T + 2\log \frac{2J_T^{\mathrm{str,lik}}}{\delta_\varphi} \right) + 1.$$

As above, $M_{\eta,T}^{\mathrm{str}}$ *controls the largest episode noise,* $L_T^{\mathrm{str,lik}}$ *the likelihood-parameter sensitivity,* $J_T^{\mathrm{str,lik}}$ *the covering-grid size, and* $\beta_{\eta,T}^{\mathrm{str}}$ *the resulting simultaneous stochastic radius. With probability at least* $1 - \delta_\varphi$*, the simultaneous confidence radius is*

$$\rho_q = B_\varphi + \Delta_{b,q} + \beta_{\eta,T}^{\mathrm{str}} = \widetilde{O}\left(\sqrt{1 + \overline{\Lambda}_T} + \sqrt{\mathcal{D}_{2\alpha}(T)} \right). \tag{22}$$

If the UCB coefficients cover this radius up to a fixed multiplicative constant, then

$$\mathfrak{R}_T^g = \widetilde{O}\left(\left[\sqrt{\overline{\Lambda}_T(1 + \overline{\Lambda}_T)} + \sqrt{\overline{\Lambda}_T \mathcal{D}_{2\alpha}(T)} \right] \sqrt{S_T} \right).$$

In particular, if $\alpha \geq 1/2$ *and* p *is fixed, then* $\mathfrak{R}_T^g = \widetilde{O}(p\sqrt{S_T})$*.*

Proof. The proof is the p-dimensional analogue of the reference-kernel argument: self-normalization gives confidence, a log determinant controls the accumulated variance, and the UCB reduction converts that variance budget into regret.

At episode q, let

$$A_q := M_q^{-1} + \sum_{q' < q} w_{q',q} \varphi(m_{q'}) \varphi(m_{q'})^\top.$$

The posterior parameter estimate and variance are

$$\widehat{\vartheta}_q := A_q^{-1} \left[M_q^{-1} \vartheta_{0,q} + \sum_{q' < q} w_{q',q} \varphi(m_{q'}) \bar{y}_{q'} \right],$$
$$\sigma_{q-1}^2(m) := \varphi(m)^\top A_q^{-1} \varphi(m).$$

Hence

$$\widehat{\vartheta}_q - \vartheta^\star = -A_q^{-1} M_q^{-1} (\vartheta^\star - \vartheta_{0,q}) + A_q^{-1} \sum_{q' < q} w_{q',q} \varphi(m_{q'}) b_{q'} + A_q^{-1} \sum_{q' < q} w_{q',q} \varphi(m_{q'}) \eta_{q'}.$$

The prior and drift terms are bounded by $B_\varphi \sigma_{q-1}(m)$ and $\Delta_{b,q} \sigma_{q-1}(m)$ exactly as in the fixed-likelihood argument.

For the stochastic term, first fix deterministic λ and apply vector self-normalization to $x_{q'}(\lambda) := \sqrt{w_{q'}(\lambda)} \varphi(m_{q'})$. The corresponding reference precision is $G_q(\lambda)$. Since

$$\log \frac{\det G_q(\lambda)}{\det(\overline{m}^{-1} I)} \leq \overline{\Lambda}_T,$$

the fixed-λ stochastic term has the usual $C_\eta \sqrt{\overline{\Lambda}_T + 2\log(1/\delta)}$ envelope.

To make the event uniform in λ, use the same one-dimensional covering argument as in Lemma 7. On $\max_{q' \leq N} |\eta_{q'}| \leq M_{\eta,T}^{\mathrm{str}}$, the squared self-normalized norm has derivative at most $L_T^{\mathrm{str,lik}}$. A

deterministic grid of $J_T^{\mathrm{str,lik}}$ points and a union bound therefore give simultaneously for every q and every admissible λ,

$$\left\| \sum_{q'<q} w_{q'}(\lambda)\varphi(m_{q'})\eta_{q'} \right\|_{G_q(\lambda)^{-1}} \le \beta_{\eta,T}^{\mathrm{str}}.$$

Evaluate at $\lambda = \lambda_{\mathrm{n},q}$. Since $M_q^{-1} \succeq \overline{m}^{-1} I$, we have $A_q \succeq G_q(\lambda_{\mathrm{n},q})$, which yields the stochastic part of Eq. (22). Also, $\log J_T^{\mathrm{str,lik}} = O(\log T)$.

For the variance budget, the current weights satisfy $w_{q',q} \ge w_-$, so

$$A_q \succeq \underline{G}_q.$$

Set $x_q^\star := \varphi(m_q)^\top \underline{G}_q^{-1} \varphi(m_q)$ and $x_{\max}^{\mathrm{str}} := \overline{m} L_\varphi^2$. Then $\sigma_{q-1}^2(m_q) \le x_q^\star \le x_{\max}^{\mathrm{str}}$, while the count-based reference has the exact update

$$\underline{G}_{q+1} = \underline{G}_q + w_- \varphi(m_q)\varphi(m_q)^\top.$$

Hence

$$\Lambda_N = \sum_q \log(1 + w_- x_q^\star).$$

The same chord argument as in Lemma 10 gives

$$\sum_q \sigma_{q-1}^2(m_q) \le \frac{x_{\max}^{\mathrm{str}}}{\log(1 + w_- x_{\max}^{\mathrm{str}})} \Lambda_N. \tag{23}$$

Finally, the trace bound and AM–GM give Eq. (21); combine Eq. (23) with Lemmas 4 and 3. □

B.8 Shared-dataset stabilization under a margin condition

We now justify Assumptions 1 from properties of the base learners trained on the common dataset $\mathcal{D}_t$. The key idea is simple: if a fitted score is uniformly close to a limiting score, the fitted policy can disagree with the limiting policy only on contexts where the best and second-best limiting actions are already close. A margin condition makes such contexts rare.

Assume the contexts c_t are i.i.d. with law P_C. For a generic context c, write $x(c,a) := \varphi(c,a)$ and let $\mathcal{A}(c)$ be its feasible action set. For each configuration $m \in \mathcal{M}$, let $\mathsf{U}_t^m(c,a)$ be its fitted action score from the shared history before round t, and let $\pi_t^m(c) \in \arg\max_{a\in\mathcal{A}(c)} \mathsf{U}_t^m(c,a)$ be the resulting fitted policy. Suppose there is a deterministic limiting score $s_m^\infty(c,a)$ with unique greedy policy $\pi_m^\infty(c) := \arg\max_{a\in\mathcal{A}(c)} s_m^\infty(c,a)$. Its action gap is $\Delta_m(c) := s_m^\infty(c,\pi_m^\infty(c)) - \sup_{a\neq\pi_m^\infty(c)} s_m^\infty(c,a)$, namely the limiting-score separation between the best and second-best actions; the supremum over an empty set is interpreted as $-\infty$.

Lemma 14 (Uniform score error implies disagreement only near the margin)**.** *Uniform score convergence alone does not immediately imply policy convergence. This lemma supplies the bridge: the fitted policy can differ from its limiting policy only on contexts whose limiting action gap is small enough to be reversed by the score error. If for some $\omega_t \ge 0$,*

$$\sup_{m,c,\, a\in\mathcal{A}(c)} |\mathsf{U}_t^m(c,a) - s_m^\infty(c,a)| \le \omega_t,$$

then for every m,

$$\{c : \pi_t^m(c) \neq \pi_m^\infty(c)\} \subseteq \{c : \Delta_m(c) \le 2\omega_t\}.$$

Proof. If the fitted policy chooses a different action, that action must overcome the limiting winner. Since each fitted score can move by at most ω_t, the limiting gap can be at most $2\omega_t$.

Fix m, c and suppose $\pi_t^m(c) \neq \pi_m^\infty(c)$. Optimality under the fitted score gives

$$s_m^\infty(c,\pi_m^\infty(c)) \le \mathsf{U}_t^m(c,\pi_m^\infty(c)) + \omega_t \le \mathsf{U}_t^m(c,\pi_t^m(c)) + \omega_t \le s_m^\infty(c,\pi_t^m(c)) + 2\omega_t.$$

Therefore $\Delta_m(c) \le 2\omega_t$. □

Proposition 3 (Shared-dataset stability under a margin condition)**.** *This proposition turns score convergence into the played-path stability condition Assumptions 1. The margin condition controls how often score errors can change the selected action, and bounded rewards convert that disagreement probability into value drift. Let $\omega_t \downarrow 0$ be deterministic and define*

$$\mathcal{E}_{\mathrm{sh},t} := \left\{ \sup_{m,c,\, a\in\mathcal{A}(c)} |\mathsf{U}_t^m(c,a) - s_m^\infty(c,a)| \le \omega_t \right\} \in \mathcal{J}_{t-1}. \tag{24}$$

Assume

$$\Pr\left(\bigcap_{t=T_{\mathrm{cal}}+1}^{T} \mathcal{E}_{\mathrm{sh},t} \right) \ge 1 - \delta_{\mathrm{sh}},$$

and, for some $C_{\mathrm{mar}}, \kappa > 0$,

$$\Pr_{C\sim P_C}(\Delta_m(C) \le u) \le C_{\mathrm{mar}} u^\kappa, \qquad m \in \mathcal{M},\ u \ge 0. \tag{25}$$

Define

$$g(m) := \mathbb{E}_{C\sim P_C}\left[f\big(x(C, \pi_m^\infty(C))\big) \right].$$

Then Assumptions 1 holds on $\mathcal{E}_{\mathrm{sh}} := \cap_t \mathcal{E}_{\mathrm{sh},t}$ with $\delta_{\mathrm{stab}} = \delta_{\mathrm{sh}}$ and

$$d_q := \frac{C_{\mathrm{mar}}}{H_q} \sum_{h=0}^{H_q-1} (2\omega_{t_{q,h}})^\kappa. \tag{26}$$

If $\omega_t \le C_\omega (t+1)^{-r}$, then the Assumptions 1 decay rate holds with $\alpha = r\kappa$.

Proof. Lemma 14 turns score error into an action disagreement event. The margin condition bounds the probability of that event, and bounded rewards turn disagreement probability into value error. I.i.d. contexts are used only to evaluate the next context under the fixed law P_C.

On $\mathcal{E}_{\mathrm{sh},t}$, Lemma 14 and Eq. (25) give

$$\Pr_{C\sim P_C}\big(\pi_t^m(C) \ne \pi_m^\infty(C)\big) \le C_{\mathrm{mar}}(2\omega_t)^\kappa.$$

Because $f \in [0,1]$, changing the selected action changes expected reward by at most this disagreement probability. Since c_t is independent of $\mathcal{J}_{t-1}$ and $\mathcal{E}_{\mathrm{sh},t}$ is measurable with respect to that history,

$$|\mathbb{E}[f(x(c_t, \pi_t^m(c_t))) \mid \mathcal{J}_{t-1}] - g(m)| \le C_{\mathrm{mar}}(2\omega_t)^\kappa.$$

Apply the bound to each round of episode q and average to obtain Eq. (26). Since $t_{q,h} \ge \tau_q$, a polynomial score rate $\omega_t = O(t^{-r})$ gives $d_q = O((\tau_q+1)^{-r\kappa})$. □

Assumption 1 is about expected reward along the *played path*, which can be difficult to verify directly. Proposition 3 reduces it to two easier questions: does the fitted score converge uniformly, and how much context probability lies near an action tie?

B.8.1 Well-specified ridge configurations

For a well-specified configuration, the shared dataset is informative about one true linear parameter. Persistent excitation then turns standard ridge concentration into a uniform score-convergence rate, which feeds the margin argument for Assumptions 1.

Let $\emptyset \ne \mathcal{M}_{\mathrm{ws}} \subseteq \mathcal{M}$ be the well-specified subset. A configuration m may transform the raw feature vector $x \in \mathcal{X}$ using its own model feature map $\psi_m : \mathcal{X} \to \mathbb{R}^{d_m}$. All configurations are trained on the same raw shared dataset $\mathcal{D}_t$; only this representation and the fitted parameters differ across configurations.

Assume, for every $m \in \mathcal{M}_{\mathrm{ws}}$,

$$\sup_{x\in\mathcal{X}} \|\psi_m(x)\|_2 \le L_\psi < \infty, \tag{27}$$

and that there exists $\theta_m^\star$ with $\|\theta_m^\star\|_2 \le S_\theta$ such that

$$f(x) = \psi_m(x)^\top \theta_m^\star \qquad \text{for every } x \in \mathcal{X}. \tag{28}$$

Let $\lambda_m \geq \lambda_{\min} > 0$ and define

$$V_t^m := \lambda_m I + \sum_{s<t} \psi_m(x_{s,a_s})\psi_m(x_{s,a_s})^\top,$$
$$\widehat{\theta}_t^m := (V_t^m)^{-1} \sum_{s<t} \psi_m(x_{s,a_s}) y_s.$$

Assume that on an event $\mathcal{E}_{\mathrm{PE}}$ of probability at least $1 - \delta_{\mathrm{PE}}$, for some $\nu > 0$, and sufficiently large t,

$$\sum_{s<t} \psi_m(x_{s,a_s})\psi_m(x_{s,a_s})^\top \succeq \nu(t-1)I \qquad (m \in \mathcal{M}_{\mathrm{ws}}). \tag{29}$$

We take the calibration prefix long enough that this lower bound holds for every post-calibration round used in the analysis; any fixed earlier prefix can be absorbed into calibration. In a disjoint-action encoding, this requires every action block to receive a linear number of samples and its conditional feature second moment to be nondegenerate. This is a covariate-diversity-type sufficient condition. The proof below uses only the one-sided matrix lower bound in Eq. (29); it does not require the empirical design matrix to converge to a deterministic limit.

Lemma 15 (Well-specified ridge parameter rate under persistent excitation)**.** *For a well-specified ridge configuration, persistent excitation makes the shared-dataset design informative in every parameter direction. This converts the standard self-normalized regression bound into a Euclidean parameter rate of order $t^{-1/2}$ up to logarithmic factors. For every $\delta_{\mathrm{r}} \in (0,1)$, with probability at least $1 - \delta_{\mathrm{PE}} - \delta_{\mathrm{r}}$, simultaneously over $m \in \mathcal{M}_{\mathrm{ws}}$ and all post-calibration $t \leq T$,*

$$\|\widehat{\theta}_t^m - \theta_m^\star\|_2 = O\left(\sqrt{\frac{\log(T|\mathcal{M}_{\mathrm{ws}}|/\delta_{\mathrm{r}})}{t}}\right),$$

where the hidden constant depends only on the fixed dimensions, $R, L_\psi, S_\theta, \lambda_{\min}$, and ν.

Proof. Self-normalized linear regression controls the estimation error in the V_t^m-norm. Persistent excitation makes the smallest eigenvalue of V_t^m grow linearly in t, converting that ellipsoidal bound into an ordinary Euclidean $t^{-1/2}$ rate.

Realizability gives

$$\widehat{\theta}_t^m - \theta_m^\star = (V_t^m)^{-1}\left[\sum_{s<t} \psi_m(x_{s,a_s})\varepsilon_s - \lambda_m \theta_m^\star\right].$$

The standard self-normalized linear-regression inequality and a union bound over the finite set $\mathcal{M}_{\mathrm{ws}}$ give

$$\|\widehat{\theta}_t^m - \theta_m^\star\|_{V_t^m} \leq R\sqrt{2\log\left(\frac{|\mathcal{M}_{\mathrm{ws}}|}{\delta_{\mathrm{r}}}\frac{\det(V_t^m)^{1/2}}{\det(\lambda_m I)^{1/2}}\right)} + \sqrt{\lambda_m} S_\theta. \tag{30}$$

Bounded model features make the determinant ratio logarithmic in T, while Eq. (29) gives $\lambda_{\min}(V_t^m) \geq \lambda_m + \nu(t-1)$. Divide Eq. (30) by $\sqrt{\lambda_{\min}(V_t^m)}$. □

Proposition 4 (Well-specified shared ridge under persistent excitation)**.** *This proposition converts the well-specified parameter rate into the uniform score convergence required by the margin argument. The limiting score is the true reward function, so the resulting stability rate follows directly from Proposition 3. Suppose the deployed score is $\mathsf{U}_t^m(c,a) := \psi_m(x(c,a))^\top \widehat{\theta}_t^m + \mathrm{bonus}_t^m(c,a)$, and*

$$\sup_{m \in \mathcal{M}_{\mathrm{ws}}, c,\, a \in \mathcal{A}(c)} |\mathrm{bonus}_t^m(c,a)| = \widetilde{O}(t^{-1/2}). \tag{31}$$

Then, on the event from Lemma 15, for every $m \in \mathcal{M}_{\mathrm{ws}}$, $s_m^\infty(c,a) = f(x(c,a))$ and the score error in Eq. (24) is $\omega_t = \widetilde{O}(t^{-1/2})$. Under the margin condition Eq. (25), Assumptions 1 therefore holds for episodes using $m \in \mathcal{M}_{\mathrm{ws}}$ with $\alpha = \kappa/2$ up to logarithmic factors.

Proof. Lemma 15 controls the fitted parameter. Bounded model features convert parameter error into uniform prediction error, and the bonus has the same order. Proposition 3 then converts the score rate into the Assumptions 1 drift rate.

By Eq. (27) and Lemma 15, uniformly over $m \in \mathcal{M}_{\mathrm{ws}}$ and feasible (c, a),

$$|\psi_m(x(c,a))^\top(\widehat{\theta}_t^m - \theta_m^\star)| \le L_\psi \|\widehat{\theta}_t^m - \theta_m^\star\|_2 = \widetilde{O}(t^{-1/2}).$$

Add Eq. (31) and use Eq. (28); hence the limiting score is the true reward and $\omega_t = \widetilde{O}(t^{-1/2})$. Apply Proposition 3 with $r = 1/2$. □

Interpretation of the well-specified route. A single $m \in \mathcal{M}_{\mathrm{ws}}$ can be enough for oracle alignment: its limiting score is the true reward function, so its limiting greedy policy selects an optimal action up to ties. Other configurations in the same heterogeneous pool may remain misspecified. This alone does not verify Assumptions 1 for the full candidate pool; every configuration that the meta-controller may deploy must be covered by this route or by another stability argument.

B.8.2 Misspecified ridge configurations

For a misspecified configuration, the fitted linear predictor need not converge to the true reward function. Its limiting target depends on the distribution of features actually collected by the shared-dataset policy. We therefore make the needed design stability explicit rather than deriving it from persistent excitation.

For each $m \in \mathcal{M}$, let $\psi_m : \mathcal{X} \to \mathbb{R}^{d_m}$ satisfy

$$\sup_{m,x} \|\psi_m(x)\|_2 \le L_\psi < \infty, \tag{32}$$

and define

$$V_t^m := \lambda_m I + \sum_{s<t} \psi_m(x_{s,a_s})\psi_m(x_{s,a_s})^\top, \qquad q_t^m := \sum_{s<t} \psi_m(x_{s,a_s}) y_s, \tag{33}$$

Let $\widehat{\theta}_t^m := (V_t^m)^{-1} q_t^m$ and $n_t := t - 1$.

Lemma 16 (Deterministic moment limits imply a pseudo-true parameter limit)**.** *Under misspecification, the ridge estimator need not converge to the true reward parameter. This lemma shows that stable empirical design and cross moments are instead sufficient for convergence to a deterministic pseudo-true parameter. Suppose that, on an event $\mathcal{E}_{\lim}$ with probability at least $1 - \delta_{\lim}$, there exist deterministic $G_m \succ 0$, vectors g_m, and a deterministic sequence $\zeta_t \downarrow 0$ such that, uniformly over m and all sufficiently large t,*

$$\left\| \frac{1}{n_t} \sum_{s<t} \psi_m(x_{s,a_s})\psi_m(x_{s,a_s})^\top - G_m \right\|_{\mathrm{op}} \le \zeta_t, \tag{34}$$

$$\left\| \frac{1}{n_t} \sum_{s<t} \psi_m(x_{s,a_s}) y_s - g_m \right\|_2 \le \zeta_t. \tag{35}$$

Define $\theta_m^\infty := G_m^{-1} g_m$. Then

$$\max_{m \in \mathcal{M}} \|\widehat{\theta}_t^m - \theta_m^\infty\|_2 = O\left(\zeta_t + \frac{1}{n_t}\right). \tag{36}$$

Proof. After dividing the ridge normal equations by n_t, both the empirical matrix and vector converge to deterministic limits. Positive definiteness of G_m then makes matrix inversion locally Lipschitz, so the estimator converges at the same order.

From Eq. (33),

$$\left\| \frac{1}{n_t} V_t^m - G_m \right\|_{\mathrm{op}} = O\left(\zeta_t + \frac{1}{n_t}\right), \qquad \left\| \frac{1}{n_t} q_t^m - g_m \right\|_2 \le \zeta_t.$$

Because $\mathcal{M}$ is finite and every G_m is positive definite, for all large t the empirical matrices are uniformly invertible. The standard identity $A^{-1} - B^{-1} = A^{-1}(B - A)B^{-1}$ therefore gives a uniform $O(\zeta_t + 1/n_t)$ inverse perturbation bound. Multiplying by the convergent cross-moment vector proves Eq. (36). □

Proposition 5 (Misspecified shared ridge under design limits)**.** *This proposition converts convergence to the pseudo-true parameter into uniform score convergence. Once the exploration bonus also vanishes, the same margin argument yields Assumptions 1 for misspecified configurations. Assume the conditions of Lemma 16. Suppose the deployed score is* $\mathsf{U}_t^m(c,a) := \psi_m(x(c,a))^\top \widehat{\theta}_t^m + \mathrm{bonus}_t^m(c,a)$*, with* $\sup_{m,c,\, a\in\mathcal{A}(c)} |\mathrm{bonus}_t^m(c,a)| \le \omega_t^{\mathrm{bon}}$ *and* $\omega_t^{\mathrm{bon}} \downarrow 0$*. Let* $s_m^\infty(c,a) := \psi_m(x(c,a))^\top \theta_m^\infty$*. Then, for a constant* $C_{\mathrm{sh}} < \infty$*,*

$$\omega_t := C_{\mathrm{sh}}\left(\zeta_t + \frac{1}{n_t \vee 1} + \omega_t^{\mathrm{bon}}\right) \tag{37}$$

satisfies Eq. (24). Consequently, if the margin condition holds and $\zeta_t, \omega_t^{\mathrm{bon}} = \widetilde{O}(t^{-1/2})$*, then Assumptions 1 holds with* $\alpha = \kappa/2$ *up to logarithmic factors.*

Proof. Lemma 16 controls the distance to the pseudo-true parameter. Bounded model features and the vanishing exploration bonus turn this into uniform score convergence, after which the same margin proposition applies.

By Eq. (32) and Lemma 16,

$$|\mathsf{U}_t^m(c,a) - s_m^\infty(c,a)| \le L_\psi \|\widehat{\theta}_t^m - \theta_m^\infty\|_2 + |\mathrm{bonus}_t^m(c,a)|,$$

which is bounded by Eq. (37) after enlarging the fixed constant C_{sh}. Proposition 3 gives the final Assumptions 1 rate. □

Failure-probability bookkeeping for Assumptions 1. When Appendix B.8 is used to verify Assumptions 1 over the full candidate pool, δ_{stab} is the union-bound failure probability of the events used to establish the uniform score-convergence event $\mathcal{E}_{\mathrm{sh}}$. For example, the well-specified ridge route contributes at most $\delta_{\mathrm{PE}} + \delta_{\mathrm{r}}$, while the misspecified design-limit route contributes δ_{lim} when it is used. Thus $\delta_{\mathrm{stab}} = \delta_{\mathrm{sh}}$ in Proposition 3; δ_{sh} should not be added again as a separate failure term.

Why misspecification needs the stronger assumption. For a well-specified model, every sufficiently informative design estimates the same true parameter; only the one-sided lower bound Eq. (29) is needed. Under misspecification, the pseudo-true target $G_m^{-1} g_m$ itself depends on the action distribution induced by the learner. Persistent excitation does not imply the two-sided convergence of the empirical design and cross moments in Eq. (34)–Eq. (35). Proving those limits from the coupled configuration-selection dynamics would require a separate fixed-point argument and is not part of the present theorem. The two routes can nevertheless be combined configuration-wise in one heterogeneous finite pool.

B.9 Comparator mismatch and an end-to-end corollary

The exact decomposition separates configuration-selection regret from the external-oracle/pool mismatch. The GP proof controls the former; the following result gives one simple condition under which the latter is also sublinear.

Proposition 6 (Oracle-aligned pool comparator)**.** *The main theorem leaves pool mismatch explicit because configuration selection cannot repair a pool that misses the contextual oracle. Under oracle alignment, this proposition shows that the remaining mismatch is only a centered sampling fluctuation of the context sequence. Assume* c_t *are i.i.d. and the feasible action set depends only on the current context,* $\mathcal{A}_t = \mathcal{A}(c_t)$*. Let* $r^\star(c) := \max_{a\in\mathcal{A}(c)} f(x(c,a))$ *be the context-wise oracle mean reward and let* $\mu^\star := \mathbb{E}_{C\sim P_C}[r^\star(C)]$ *be its expectation over the context distribution. If the configuration pool is oracle aligned in the sense that*

$$g^\star = \mu^\star, \tag{38}$$

then for every $\delta_{\mathrm{pool}} \in (0,1)$*, with probability at least* $1 - \delta_{\mathrm{pool}}$*,*

$$M_T^{\mathrm{pool}} \le \sqrt{\frac{T'}{2}\log\frac{1}{\delta_{\mathrm{pool}}}}. \tag{39}$$

Proof. Under oracle alignment, M_T^{pool} is simply the centered sum of the per-context oracle values $r^\star(c_t)$. These variables are i.i.d. and lie in $[0,1]$, so Hoeffding's inequality applies directly.

By Eq. (38),

$$M_T^{\mathrm{pool}} = \sum_{t=T_{\mathrm{cal}}+1}^{T} \left[r^\star(c_t) - \mu^\star\right].$$

The summands are independent, mean zero, and lie in an interval of length one. Hoeffding's inequality gives Eq. (39). □

Condition Eq. (38) is a property of the candidate pool, not an automatic consequence of the meta-controller. It holds, for example, when $\mathcal{M}_{\mathrm{ws}} \neq \emptyset$ and at least one well-specified configuration has a limiting greedy policy that is optimal almost surely, modulo ties.

Corollary 1 (End-to-end sublinear regime)**.** *This corollary closes the remaining comparator term in the main theorem. With oracle alignment, sufficiently fast shared-dataset stabilization, and the finite-pool information-gain bound, the full contextual-bandit regret is sublinear. Assume the conditions of Theorem 1 for a fixed finite pool and fixed regularity constants, the i.i.d.-context and oracle-alignment conditions of Proposition 6, and a calibration prefix with $\mathfrak{R}_{\mathrm{cal}} = O(1)$. If shared-dataset stability gives $\alpha \geq 1/2$ uniformly over all configurations that may be deployed, then on the intersection of the corresponding high-probability events,*

$$\mathfrak{R}_T(\pi) = \widetilde{O}\Big(T^{(1+\beta)/2}\Big).$$

Proof. For fixed K, the worst-case reference information gain grows only logarithmically with T. Lemma 5 gives $S_T = O(T^{1+\beta})$, and Proposition 6 controls the remaining pool-comparator term by $O(\sqrt{T})$.

For fixed K and fixed problem constants, Eq. (16) gives $\overline{\Gamma}_T^{\mathrm{ref}} = O(\log T)$. Lemma 5 gives $S_T = O(T^{1+\beta})$, and Proposition 6 gives $M_T^{\mathrm{pool}} = O(\sqrt{T\log(1/\delta_{\mathrm{pool}})})$. Substitute these bounds into Theorem 1. □

Remark. For uniformly bounded episode lengths ($\beta = 0$), Corollary 1 reduces to $\mathfrak{R}_T(\pi) = \widetilde{O}(\sqrt{T})$. For any $0 < \beta < 1$, the same corollary remains sublinear, with the larger exponent arising through $S_T = O(T^{1+\beta})$ rather than through the shared-dataset drift term.

B.10 Continuous and mixed configuration spaces

Lemma 1 and the exact decomposition in Lemma 2 use only episodic deployment and Assumptions 1; they do not use finiteness of $\mathcal{M}$. Their observation-model interpretation therefore continues to apply when configurations contain continuous variables, discrete variables, or both.

The finite-grid UCB proof relies on a finite Gram matrix, finite-pool reference information gain, and exact maximization over the candidate pool. Extending the selection-regret bound to continuous or mixed spaces requires accounting for optimization and approximation errors. Our mixed backend uses TuRBO-style trust-region Bayesian optimization (Eriksson et al., 2019). We therefore use the finite-pool result to motivate experiments in denser configuration spaces and the episode decomposition to interpret the mixed experiments. We do not claim a continuous-space selection-regret bound.

C Additional Experimental Details and Results

Unless stated otherwise, results follow the protocol of Sec. 5: curves show the median over seeds with the IQR shaded, and tables report the median ± standard deviation of final regret.

C.1 Implementation Details

Environments. Tab. 1 summarizes the environments. For the classification environments, the input dimension of the base learner is nd_c because of the disjoint encoding $x_{t,a} = e_a \otimes c_t$, with e_a the a-th basis vector of $\mathbb{R}^n$: the context fills the a-th of n blocks and the other blocks are zero.

- **MNIST.** Each 28×28 image is scaled to $[0, 1]$, replicated across three channels, resized to 64×64 by bilinear interpolation, and normalized with the ImageNet channel statistics. It is then passed through a frozen ImageNet-pretrained ResNet-18 from `torchvision` (He et al., 2016), truncated after its second residual stage (`layer2`, 128 channels), and the output feature map is averaged over its spatial dimensions and normalized to unit ℓ_2 norm, giving $c_t \in \mathbb{R}^{128}$.
- **DBPEDIA.** Each text is embedded with the `all-MiniLM-L6-v2` sentence encoder (Wang et al., 2020; Reimers & Gurevych, 2019), normalized to unit ℓ_2 norm, giving $c_t \in \mathbb{R}^{384}$; all 14 classes are used as arms.
- **MOVIELENS.** Following Ding et al. (2022), we factorize the full MovieLens-100K rating matrix (943 users, 1,682 movies) with LIBPMF (Yu et al., 2012) at rank 20 and regularization 0.05, giving 20-dimensional user and movie factors. An arm is a movie, represented by its factor vector $x_{t,a} \in \mathbb{R}^{20}$, and each round offers $n = 30$ movies drawn uniformly without replacement. Each seed fixes one user vector $u \in \mathbb{R}^{20}$, the mean of 100 randomly chosen user factors. The reward is Bernoulli with mean $f(x_{t,a}) = \mathrm{sigmoid}(z_{t,a})$, where $z_{t,a}$ is $x_{t,a}^\top u$ standardized by its mean and standard deviation over all rounds and arms; without this standardization the means saturate near 1 and the best arm is barely distinguishable from the second best. The arm sets are generated once per seed and shared by all methods.
- **Synthetic.** Each round draws a context from a fixed set of 10^4 contexts and offers $n = 10$ arms with features $x_{t,a} \in \mathbb{R}^8$; the reward is $f(x_{t,a})$ plus Gaussian noise with standard deviation 0.1 on Cosine and 0.01 on Square, where $f(x) = \cos(3x^\top u)$ on Cosine and $f(x) = (x^\top u)^2$ on Square for a fixed $u \in \mathbb{R}^8$, rescaled to $[0, 1]$.

Table 1: Environments. n: number of arms; d_c: context dimension (for the synthetic environments and MOVIELENS, the dimension of $x_{t,a}$); K: size of the discrete configuration pool.

Environment	Type	Context	n	d_c	K
Cosine	synthetic	–	10	8	30
Square	synthetic	–	10	8	30
MOVIELENS	recommendation	matrix-factorization factors	30	20	27
SHUTTLE	classification	raw features	7	9	28
MNIST	classification	ResNet-18 features	10	128	33
DBPEDIA	classification	MiniLM embedding	14	384	33

Configuration spaces. *Linear* learners are ridge regressions on $x_{t,a}$ with LinUCB; under the disjoint encoding, they decompose into independent per-arm ridge regressions. *Polynomial* learners are ridge regressions on a fixed monomial feature map of degree 2 or 3, with nothing learned in the map. *Neural* learners are fully connected ReLU networks of constant width with NeuralUCB (Zhou et al., 2020), using a diagonal approximation of the design matrix and $\nu = 1$. They are retrained every round with learning rate 0.01 and batch size 64 for at most 100 epochs, stopping early at loss tolerance 10^{-3} (checked every 5 epochs). All learners select actions by UCB with confidence parameter $\delta = 0.05$, and the exploration scale ζ multiplies the confidence width. Tab. 2 gives the discrete pool of each environment: within a family, every architecture is combined with every regularization λ and exploration scale ζ. In the continuous setting, λ ranges over $[10^{-8}, 10^{-3}]$ and ζ over $[10^{-3}, 1]$ ($[10^{-2}, 1]$ on MNIST and DBPEDIA), both on a log scale. To maximize the acquisition over the continuous space, we partition it into sub-regions and draw Sobol samples from each. The selected ζ has a median of about 0.01 to 0.26 across environments.

COMAS settings. *Episode schedule:* episode lengths satisfy $H_q \le C_H(\tau_q + 1)^\beta$ with $C_H = 1$ and $\beta = 0.3$. *Meta-GP:* a Matérn-5/2 kernel with automatic relevance determination (ARD) over the configuration descriptors and a separate learner-family length-scale, implemented in GPyTorch. Its hyperparameters are relearned every 5 episodes with 50 iterations of marginal-likelihood optimization. The likelihood follows Assumption 3, with the residual noise $\lambda_{\mathrm{n},q}$ relearned within a bounded interval. *Acquisition:* COMAS (TS) implements the Thompson sampling rule of Eq. (3) with independent draws from the posterior marginals of the K configurations,

$$\widetilde{g}_{q,j} = \mu_{q-1}(m_j) + \alpha_q \sigma_{q-1}(m_j)\xi_{q,j}, \quad \xi_{q,j} \overset{\mathrm{iid}}{\sim} \mathcal{N}(0,1), \tag{40}$$

Table 2: Discrete configuration pools. Neural architectures are given as width $\times$ number of hidden layers. Last column: number of configurations.

Environment	Family	Architecture	λ / ζ	#
Cosine, Square ($K = 30$)	Linear	–	$\{10^{-3}, 10^{-2}, 10^{-1}\}$ / $\{0.1, 1\}$	6
	Polynomial	degree 2, 3	$\{10^{-3}, 10^{-2}, 10^{-1}\}$ / $\{0.1, 1\}$	12
	Neural	16×2, 32×2	$\{10^{-3}, 10^{-2}, 10^{-1}\}$ / $\{0.1, 1\}$	12
MovieLens ($K = 27$)	Linear	–	$\{10^{-3}, 10^{-2}, 10^{-1}\}$ / $\{0.01, 0.1, 1\}$	9
	Polynomial	degree 2	$\{10^{-3}, 10^{-2}, 10^{-1}\}$ / $\{0.01, 0.1, 1\}$	9
	Neural	8×2	$\{10^{-3}, 10^{-2}, 10^{-1}\}$ / $\{0.01, 0.1, 1\}$	9
Shuttle ($K = 28$)	Linear	–	$\{10^{-4}, 10^{-2}\}$ / $\{0.1, 1\}$	4
	Polynomial	degree 2, 3	$\{10^{-4}, 10^{-3}, 10^{-2}\}$ / $\{0.1, 1\}$	12
	Neural	16×2, 32×2, 64×2	$\{10^{-4}, 10^{-2}\}$ / $\{0.1, 1\}$	12
MNIST, DBpedia ($K = 33$)	Linear	–	$\{10^{-3}, 10^{-2}, 10^{-1}\}$ / $\{0.01, 0.1, 1\}$	9
	Neural	64×1, 64×2, 128×1, 128×2	$\{10^{-5}, 10^{-4}, 10^{-3}\}$ / $\{0.1, 1\}$	24

and selects $m_q = m_{j^\star}$ with $j^\star \in \operatorname{argmax}_{j \in [K]} \widetilde{g}_{q,j}$. Both variants use the fixed coefficient $\alpha_q = 1.5$, as the posterior scale of COMAS (TS) and in Eq. (2) for COMAS (UCB); for COMAS (UCB), this constant replaces the theoretical coefficient of Assumption 4.

Baselines and offline references. Offline-best and offline-worst are the configurations with the lowest and highest median final regret over seeds; offline-average is the mean over all configurations of the pool, computed per seed. Following Zhang et al. (2021a), offline neural configurations stop training after 3000 rounds and keep their weights fixed afterward.

C.2 Final Regret

Table 3: **Final regret, discrete configuration pool** (Fig. 2). The last two rows give the learner family of offline-best and offline-worst.

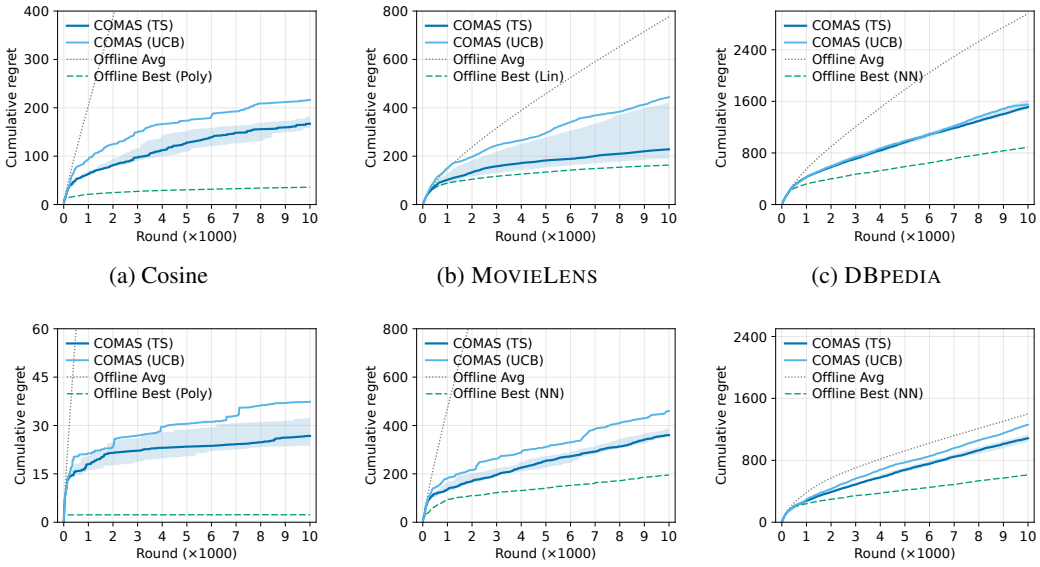
Method	Cosine	Square	MovieLens	Shuttle	MNIST	DBpedia
COMAS (TS)	119.1 ± 161.6	26.9 ± 10.7	226.4 ± 55.0	348.0 ± 50.3	1248.5 ± 721.8	1460.0 ± 165.1
COMAS (UCB)	108.7 ± 362.1	22.3 ± 55.3	230.7 ± 149.4	326.0 ± 56.5	959.0 ± 774.0	1446.0 ± 2505.5
EXP3.P	1707.4 ± 59.2	895.0 ± 27.9	1769.8 ± 91.9	2519.5 ± 63.9	4480.5 ± 237.8	5625.5 ± 365.7
Smooth-CORRAL	1511.5 ± 39.2	784.9 ± 46.1	1630.6 ± 71.8	2288.5 ± 208.9	4300.0 ± 445.7	4748.0 ± 493.2
Offline-best	35.7 ± 6.2	2.3 ± 1.3	163.2 ± 23.7	195.0 ± 176.3	614.0 ± 62.7	892.0 ± 51.6
Offline-average	1819.2 ± 49.9	1046.9 ± 70.0	767.2 ± 122.2	3376.4 ± 162.6	1384.2 ± 48.8	2989.9 ± 130.7
Offline-worst	3760.2 ± 528.6	3069.8 ± 260.8	2018.2 ± 965.7	8288.0 ± 19.7	5374.0 ± 27.2	9065.5 ± 638.5
Family of offline-best	Poly	Poly	Lin	NN	NN	NN
Family of offline-worst	NN	NN	NN	Poly	Lin	NN

C.3 Why COMAS Works

This section summarizes the evidence detailed in Sections C.6 and C.7. *Shared interaction data are essential.* Refitting each configuration only on the rounds in which it was selected raises the final regret of COMAS (UCB) on Cosine from 149 to 229, worse in $10/10$ seeds (paired Wilcoxon $p = 0.002$): configurations that are not played early receive little data and are rarely selected again (Sec. C.6.1). *Played-path stability holds empirically.* Episode reward shows no trend with the order in which configurations are first evaluated, and the expected reward of the most-played configurations drifts by a median of 0.0015 per 1000 rounds, consistent with Assumption 1 (Sec. C.6.2). COMAS *settles on the right family early.* The most-selected configuration belongs to the offline-best family in 9–$10/10$ seeds for COMAS (TS) and in 5–$10/10$ for COMAS (UCB). Within about 2000 rounds, COMAS (TS) uses configurations whose offline per-round regret is within 1.1–$1.4\times$ that of offline-best on MNIST, DBpedia and MovieLens; the realized regret of COMAS also includes the cost of continued exploration (Sec. C.7).

C.4 CONTINUOUS CONFIGURATION SPACE

Fig. 4 shows all six environments for the continuous setting of Sec. 5.2, and Tab. 4 gives the final regret. COMAS (UCB) is higher than in the discrete setting on all six environments.



(a) Cosine (b) MOVIELENS (c) DBPEDIA

(d) Square (e) SHUTTLE (f) MNIST

Figure 4: **Continuous configuration space, all environments.** Cumulative regret of COMAS (TS) and COMAS (UCB) with continuous regularization and exploration scale. Offline references are those of the discrete pool (Fig. 2).

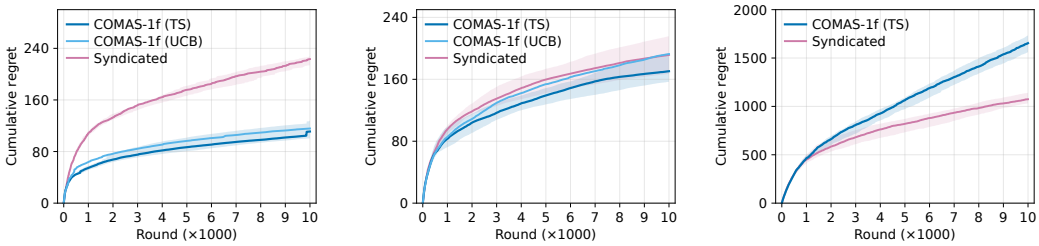
Table 4: **Final regret, continuous configuration space** (Fig. 4).

Method	Cosine	Square	MOVIELENS	SHUTTLE	MNIST	DBPEDIA
COMAS (TS)	167.0 ± 39.7	26.8 ± 5.3	228.3 ± 136.5	360.5 ± 52.6	1085.0 ± 122.5	1510.5 ± 109.5
COMAS (UCB)	216.1 ± 73.2	37.3 ± 9.4	444.1 ± 196.8	459.5 ± 104.4	1237.5 ± 128.9	1551.4 ± 61.6
Offline-best	35.7 ± 6.2	2.3 ± 1.3	163.2 ± 23.7	195.0 ± 176.3	614.0 ± 62.7	892.0 ± 51.6
Offline-average	1819.2 ± 49.9	1046.9 ± 70.0	767.2 ± 122.2	3376.4 ± 162.6	1384.2 ± 48.8	2989.9 ± 130.7

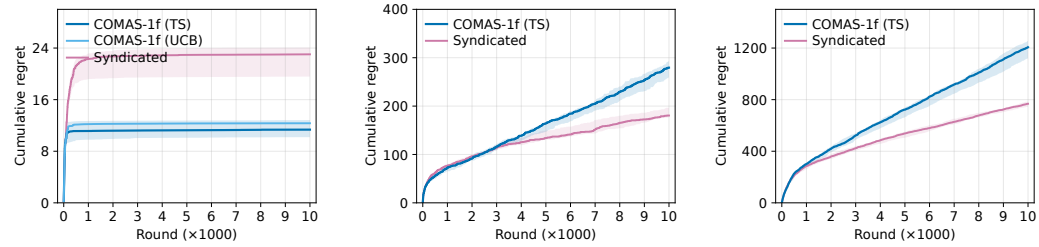
C.5 SINGLE-FAMILY TUNING BASELINES

Syndicated Bandits (Ding et al., 2022) tunes the regularization λ and exploration scale ζ of one fixed learner family, so it does not apply to the heterogeneous pools of Fig. 2. Syndicated Bandits uses as its base learner the family of the offline-best configuration in each environment: polynomial on Cosine and Square, linear on MOVIELENS, and neural on SHUTTLE, MNIST, and DBPEDIA (Tab. 3). This family is not known in advance, so Syndicated Bandits receives hindsight information. For a like-for-like comparison, COMAS-1f runs COMAS on the sub-pool of Tab. 2 restricted to the same family, on the same seeds. Fig. 5 shows the cumulative regret and Tab. 5 the final regret.

COMAS-1f ends lower than Syndicated Bandits on Cosine (111.1 vs. 223.4), Square (11.4 vs. 23.0) and MOVIELENS (170.5 vs. 191.5). Syndicated Bandits ends lower on SHUTTLE (180.5 vs. 279.5), MNIST (768.0 vs. 1206.5) and DBPEDIA (1074.0 vs. 1654.0). These are the three classification environments, whose disjoint-encoded inputs have dimension $nd_c = 63$, 1280, and 5376. When the family is fixed in advance, COMAS therefore needs further evaluation on such higher-dimensional tasks.



(a) Cosine (b) MOVIELENS (c) DBPEDIA



(d) Square (e) SHUTTLE (f) MNIST

Figure 5: **Single-family tuning.** Cumulative regret of COMAS restricted to one learner family (COMAS-1f) and Syndicated Bandits on the same family.

Table 5: **Final regret, single-family tuning** (Fig. 5). –: not run.

Method	Cosine	Square	MOVIELENS	SHUTTLE	MNIST	DBPEDIA
COMAS-1f (TS)	111.1 ± 18.2	11.4 ± 2.7	170.5 ± 25.8	279.5 ± 25.5	1206.5 ± 87.3	1654.0 ± 173.0
COMAS-1f (UCB)	115.7 ± 7.7	12.3 ± 2.8	192.7 ± 16.0	–	–	–
Syndicated Bandits	223.4 ± 11.7	23.0 ± 2.6	191.5 ± 32.2	180.5 ± 20.4	768.0 ± 39.2	1074.0 ± 66.0

C.6 ABLATIONS

Setup. The ablations use the discrete Cosine pool ($K = 30$) and change one component at a time. They differ from the runs of Fig. 2 only in that the residual likelihood noise $\lambda_{\mathrm{n},q}$ of Assumption 3 is held fixed instead of relearned. Because of this difference, ablation numbers are not directly comparable with Tab. 3: for example, COMAS (UCB) with shared interaction data in Sec. C.6.1 ends at 148.9 on Cosine, against 108.7 in Tab. 3. We report medians with IQR over seeds and paired Wilcoxon signed-rank tests across seeds.

C.6.1 SHARED INTERACTION DATA VERSUS PER-CONFIGURATION DATA

We compare COMAS (UCB) with a learned ARD kernel, refitting each selected configuration on the shared interaction data $\mathcal{D}_t$, with a variant in which a configuration's learner is refit only on the rounds collected while that configuration was selected. The meta-GP and the episode schedule are unchanged. Fig. 6 and Tab. 6 show the results.

The per-configuration variant is worse in $10/10$ seeds (median paired difference $+78$, $p = 0.002$), and in two seeds its regret blows up (1318 and 890). The cause is data starvation. The configuration played early accumulates data and improves, while every other configuration is judged on little data and is rarely selected again. The per-configuration runs spend 98% of the second-half rounds on one configuration, whereas the runs with shared interaction data spread them over about nine. In the two blow-up seeds, the locked configuration is poor, with offline regret 1120 and 366 against an offline-best of 36.

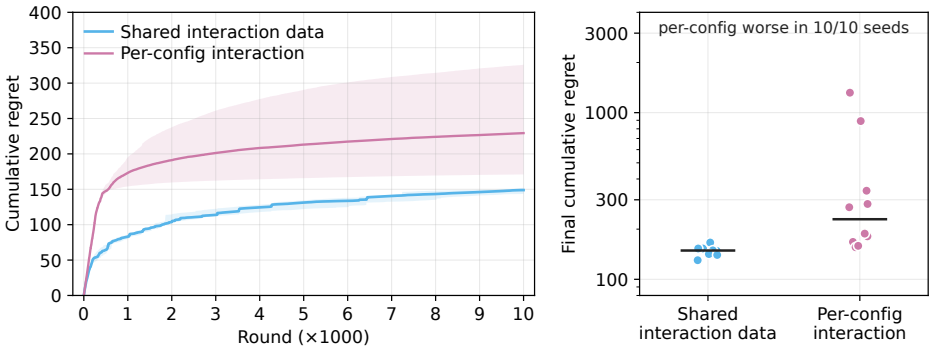


Figure 6: **Shared interaction data versus per-configuration data (Cosine).** Left: cumulative regret of COMAS (UCB). Right: final regret per seed; lines pair the two variants on the same seed, and horizontal bars mark the medians.

Table 6: **Shared interaction data versus per-configuration data (Cosine).** Final regret of COMAS (UCB), median [IQR] and per seed.

Data	Final regret	Per seed
Shared interaction data $\mathcal{D}_t$	148.9 [143.4, 152.0]	149, 153, 153, 130, 149, 148, 166, 149, 142, 140
Per-configuration	229.4 [171.0, 325.9]	340, 271, 181, 1318, 188, 168, 283, 890, 156, 159

C.6.2 Played-Path Stability and Time Confounding

Because all configurations are refitted on the shared interaction data, a configuration evaluated later sees more data than one evaluated earlier. If this mattered, later evaluations would look better regardless of the configuration, and Assumption 1 would fail. To test this, we force one episode per configuration in a random order, using the 30-configuration Cosine pool, 5 environment seeds, and 3 random orders (15 runs, 450 forced episodes). The time effect is estimated only from these randomized episodes. After the forced episodes, each run continues with COMAS (TS) with a learned ARD kernel under the ablation setup.

Table 7: **Time confounding (Cosine).** Effect of the position of a configuration's forced episode on its episode reward, for the observed reward $\bar{y}_q$ and its noiseless counterpart $\bar{f}_q$, the mean of f over the played arms, whose conditional expectation is the average of $r_{q,h}$ in Assumption 1. Slopes are given $\pm$ one standard error.

Quantity	Observed $\bar{y}_q$	Noiseless $\bar{f}_q$
Fixed-effects slope on $\log(\tau_q + 1)$	$+0.013 \pm 0.019$	$+0.023 \pm 0.018$
Per-configuration slopes: median (% positive)	$+0.033$ (60%)	$+0.022$ (63%)
Centred mean: early / middle / late	-0.010 / $+0.007$ / $+0.003$	-0.017 / $+0.009$ / $+0.008$

No detectable time confounding. Neither slope is significant; both are within 1.3 standard errors of zero (Tab. 7). The differences between terciles are about 0.01, against a within-configuration residual spread of about ± 0.5. Configurations evaluated later, therefore, do not look better because more shared interaction data are available (Fig. 7, left).

Played-path stability. Over the second half of the run, the median absolute drift of $\bar{f}_q$ is 0.0015 per 1000 rounds, over 98 configuration–run pairs with at least 10 episodes after the forced ones. The most-played configurations settle within about 1000 rounds and then fluctuate around a stable level of 0.97–0.99 (Fig. 7, right). This is consistent with Assumption 1.

Sensitivity to the evaluation order. The order of the forced episodes matters: the variance of final regret across orders is $0.57\times$ that across environment seeds, and the mean total-variation distance

between the selection frequencies after the forced episodes of different orders is 0.34. The median final regret over the three orders is 139.0, 125.8, and 125.2, respectively.

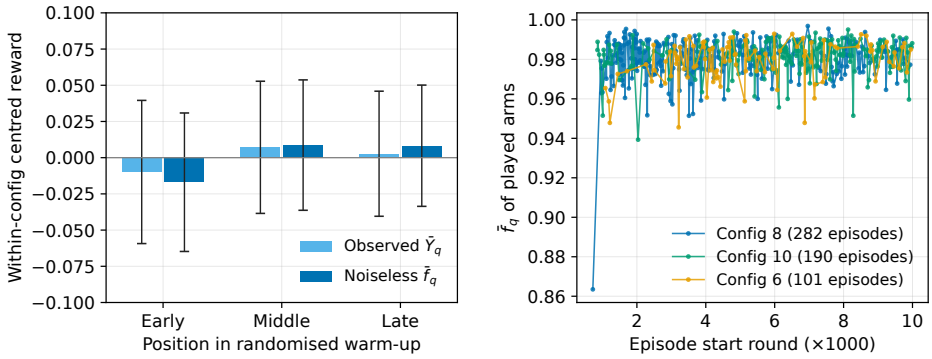


Figure 7: **Time confounding and played-path stability (Cosine).** Left: mean within-configuration centered reward of the forced episodes in the early, middle, and late thirds of the random order, for the observed $\bar{y}_q$ (light) and the noiseless $\bar{f}_q$ (dark), labeled $\bar{\mu}_q$ in the plot. Right: noiseless episode reward $\bar{f}_q$ of the three most-played configurations of one run against the start round of the episode.

C.7 Configurations Selected by COMAS

Definitions. For each seed, the *most-selected* configuration is the one with the largest share of the T rounds, $\sum_q H_q \mathbb{1}_{\{m_q=m\}}/T$. We match it by family, λ, degree, architecture, and ζ to the offline sweep and take its median final offline regret over the offline seeds. Δ is its offline regret minus that of offline-best. Configurations are written as (family, λ, ζ), with neural architectures as width $\times$ number of hidden layers and polynomial learners as Poly-d for degree d; in the per-seed tables, * marks offline-best and † offline-best up to ζ. Tab. 8 summarizes all environments, Figs. 8 and 9 show the per-seed spread for COMAS (TS), and Tables 11 to 16 list every seed.

Table 8: **Most-selected configuration.** For each seed, the most-selected configuration is the one with the largest share of the T rounds. Share: median [min, max] over seeds; "merged" counts configurations that differ only in the exploration scale as one. The next three columns count the seeds whose most-selected configuration is the offline-best configuration, match it up to the exploration scale, or belong to the offline-best family. Last column: median final regret of COMAS.

Environment	Method	Share (%)	Merged (%)	= best	= best up to ζ	Best family	Final regret
Cosine	TS	41 [29, 48]	80	3/10	6/10	9/10	119.1
	UCB	90 [30, 98]	93	1/10	1/10	9/10	108.7
Square	TS	20 [18, 51]	38	3/10	5/10	10/10	26.9
	UCB	70 [27, 93]	87	1/10	1/10	9/10	22.3
MovieLens	TS	28 [12, 35]	79	1/10	5/10	9/10	226.4
	UCB	97 [70, 99]	97	1/10	2/10	7/10	230.7
Shuttle	TS	46 [26, 50]	91	1/10	2/10	10/10	348.0
	UCB	99 [96, 99]	99	1/10	3/10	10/10	326.0
MNIST	TS	46 [22, 52]	90	0/10	2/10	10/10	1248.5
	UCB	99 [98, 100]	99	0/10	1/10	6/10	959.0
DBpedia	TS	35 [26, 50]	70	0/10	0/10	10/10	1460.0
	UCB	99 [69, 100]	99	0/10	0/10	5/10	1446.0

COMAS (UCB) commits; COMAS (TS) spreads over ζ-twins. COMAS (UCB) puts a median of 70–99% of the rounds on a single configuration. COMAS (TS) puts only 20–46% on its most-selected configuration and splits its rounds among configurations that differ only in ζ (ζ-*twins*). With ζ-twins merged, COMAS (TS) concentrates 70–91% of the rounds on five environments (Fig. 8). On Square it stays spread (38%), because several degree-2 polynomial configurations are nearly equivalent there (offline regret 2.3–10.4).

COMAS finds the family, not the exact configuration. The most-selected configuration is the offline-best configuration in at most $3/10$ seeds for either variant. Counting the ζ-twin of offline-best as a match, COMAS (TS) reaches 5–$6/10$ on Cosine, Square and MovieLens, but 0–$2/10$ on the three neural environments. The most-selected configuration belongs to the offline-best family in 9–$10/10$ seeds for COMAS (TS) on every environment, and in 5–$10/10$ for COMAS (UCB). The gap to offline-best thus comes mostly from settling on a neighboring λ or ζ.

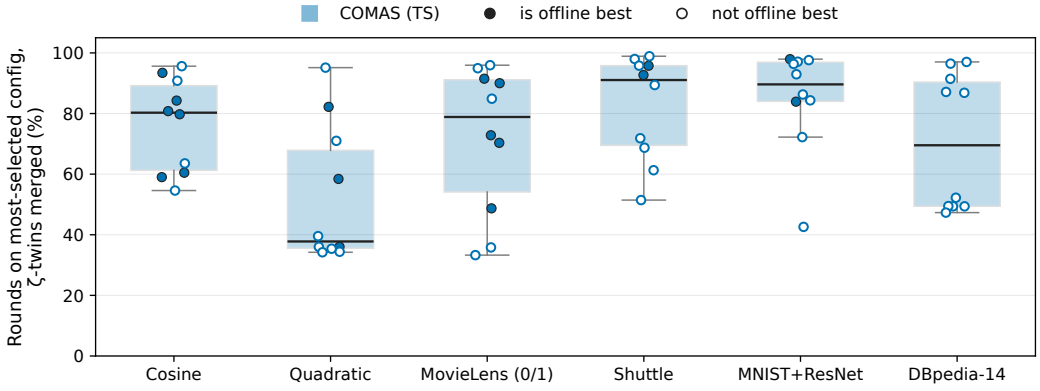


Figure 8: **Concentration of COMAS (TS).** Share of rounds on each seed's most-selected configuration, with ζ-twins merged. Filled markers: the configuration is offline-best (up to ζ); hollow: it is not.

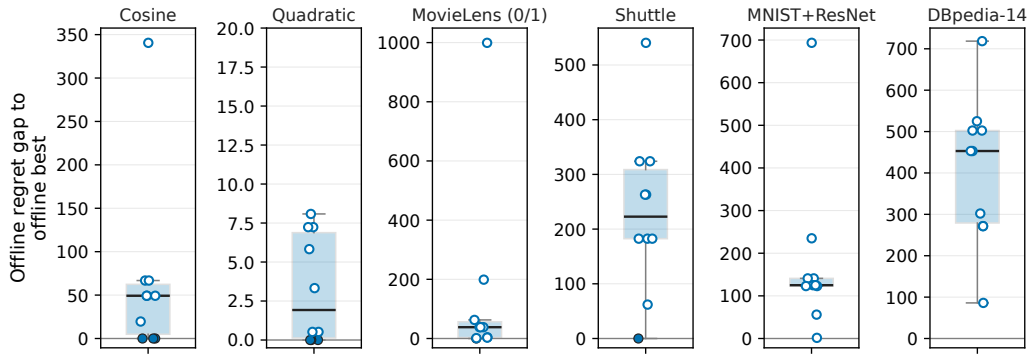
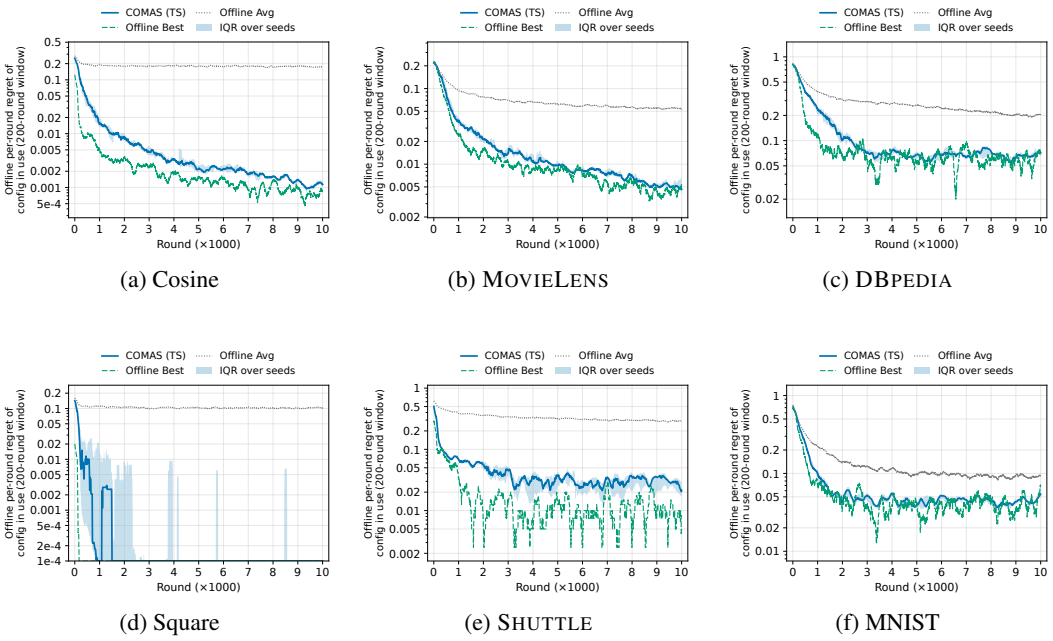


Figure 9: **Distance to offline-best.** Offline regret of each seed's most-selected configuration under COMAS (TS) minus that of offline-best; 0 means COMAS (TS) settled on offline-best.

Offline regret is not the cost inside COMAS. Inside COMAS, a configuration is refit on the shared interaction data each time it is selected, so it does not pay the learning transient that dominates a standalone offline run. For example, on Square one COMAS (UCB) seed settles on a neural configuration with offline regret 1047.9 yet ends at 196.0, and on MovieLens two COMAS (UCB) seeds settle on neural configurations with offline regret above 1000 yet end at 335.1 and 363.5. Across seeds, the offline regret of the most-selected configuration correlates strongly with the final regret of COMAS on Cosine ($+0.96$ for both variants), MNIST ($+0.89$ / $+0.83$ for TS / UCB), and MovieLens ($+0.89$ / $+0.62$), but weakly or negatively on Shuttle (-0.35 / $+0.14$) and for COMAS (TS) on DBpedia (-0.42). Δ should therefore be read as a distance between configurations, not as the regret COMAS paid. The largest final regrets of COMAS (UCB) arise when it settles on a poor configuration: on DBpedia, seeds 1 and 9 settle on configurations with offline regret 8738.5 and 2257.5 and end at 8748 and 5079, which produces its large standard deviation in Tab. 3.

Quality of the configuration in use over time. Fig. 10 tracks, for COMAS (TS), the offline per-round regret of the configuration in use at each round, $(\mathfrak{R}^{\text{off}}_{t+100}(m) - \mathfrak{R}^{\text{off}}_{t-100}(m))/200$, where $\mathfrak{R}^{\text{off}}_t(m)$ is the median offline cumulative regret of configuration m. Each seed's series is averaged

over a centered 200-round window, then the median and IQR are taken over seeds, because COMAS (TS) switches configuration about every 13 rounds. In every environment, COMAS (TS) moves from configurations of offline-average quality to near-best ones within about 1000–2000 rounds and stays there. Tab. 9 gives the second-half values. On MNIST, DBPEDIA and MOVIELENS, the per-round offline regret of the configurations in use is within 1.1–1.4× that of offline-best. Their final offline regret is higher mainly because they learn more slowly early on. SHUTTLE is the one environment with a clear late-run gap (2.9×).



(a) Cosine (b) MOVIELENS (c) DBPEDIA

(d) Square (e) SHUTTLE (f) MNIST

Figure 10: **Quality of the configuration in use.** Offline per-round regret of the configuration COMAS (TS) uses at each round (200-round window; log scale), against offline-best and offline-average.

Table 9: **Configuration in use, second half ($t > T/2$).** Medians over seeds and rounds of the final offline regret $\mathfrak{R}_T^{\text{off}}(m)$ and the offline per-round regret of the configuration COMAS (TS) uses, against offline-best.

	Offline-best		COMAS (TS)	
Environment	Final	Per round	Final	Per round
Cosine	35.7	0.0010	67.8	0.0017
Square	2.3	$< 10^{-5}$	6.9	$< 10^{-5}$
MOVIELENS	163.2	0.0053	184.4	0.0073
SHUTTLE	195.0	0.0100	438.0	0.0293
MNIST	614.0	0.0400	712.3	0.0445
DBPEDIA	892.0	0.0600	1207.4	0.0669

Pool quality and the offline-best reference. Tab. 10 estimates, for each environment, the gap between the pool and the contextual-bandit oracle, M_T^{pool} in Theorem 1. For each configuration m, $\hat{g}(m)$ is its mean noiseless reward over rounds $t > T/2$ in the offline sweep, and $\hat{M}^{\text{pool}} = \sum_{t>T/2} \max_a f(x_{t,a}) - (T/2)\,\hat{g}^\star$ with $\hat{g}^\star = \max_m \hat{g}(m)$. The estimate is conservative because offline learners may not have stabilized, and standalone runs do not refit on shared interaction data. On MNIST and DBPEDIA, the configuration with the best second-half value is linear, not the neural offline-best configuration, which wins on total regret because it learns fastest early; its offline training stops at round 3000. COMAS (UCB) settles on a linear configuration in $4/10$ MNIST seeds and

5/10 DBPEDIA seeds, and these seeds end with lower median regret than those that settle on neural configurations (789 vs. 1020 on MNIST, 1086 vs. 1510 on DBPEDIA). In these two environments, offline-best, ranked by total regret, is therefore not the configuration that is best late in the run.

Table 10: **Pool quality.** Offline-best final regret and configuration, estimated pool gap $\hat{M}^{\text{pool}}$ over $t > T/2$, and the configuration maximizing the second-half value $\hat{g}$.

Environment	K	Offline-best	Configuration	$\hat{M}^{\text{pool}}$	$\arg\max_m \hat{g}(m)$
Cosine	30	35.7	Poly-2, 10^{-3}, 0.1	4.9	same
Square	30	2.3	Poly-2, 10^{-3}, 0.1	0.0	Poly-2, 10^{-2}, 0.1
MOVIELENS	27	163.2	Lin, 10^{-3}, 0.1	30.1	same
SHUTTLE	28	195.0	NN 64×2, 10^{-4}, 1	77.8	same
MNIST	33	614.0	NN 128×2, 10^{-5}, 1	176.8	Lin, 10^{-2}, 0.01
DBPEDIA	33	892.0	NN 64×1, 10^{-4}, 1	291.0	Lin, 10^{-1}, 0.01

Per-seed configurations. Tables 11 to 16 list, for every seed, the most-selected configuration of each variant, its share of rounds, Δ, and the final regret of COMAS on that seed.

Table 11: **Cosine: most-selected configuration per seed.** Offline best: Poly-2, 10^{-3}, 0.1 (offline regret 35.7).

	COMAS (TS)				COMAS (UCB)			
Seed	Configuration	Share	Δ	Regret	Configuration	Share	Δ	Regret
0	NN 16×2, 10^{-3}, 1	48%	340.5	626.9	Poly-2, 10^{-2}, 0.1	96%	19.6	51.4
1	Poly-2, 10^{-3}, $1^{\dagger}$	42%	49.2	143.3	Poly-3, 10^{-1}, 0.1	90%	142.0	137.5
2	Poly-2, 10^{-3}, $0.1^{\star}$	47%	0.0	99.1	Poly-2, 10^{-2}, 0.1	90%	19.6	106.8
3	Poly-2, 10^{-3}, $0.1^{\star}$	31%	0.0	119.2	Poly-2, 10^{-2}, 0.1	86%	19.6	80.8
4	Poly-2, 10^{-2}, 1	48%	66.8	106.4	NN 32×2, 10^{-3}, 0.1	90%	1103.9	1239.0
5	Poly-2, 10^{-3}, $1^{\dagger}$	34%	49.2	104.0	Poly-2, 10^{-2}, 1	61%	66.8	142.6
6	Poly-2, 10^{-3}, $0.1^{\star}$	44%	0.0	121.4	Poly-2, 10^{-3}, $0.1^{\star}$	30%	0.0	345.4
7	Poly-2, 10^{-3}, $1^{\dagger}$	41%	49.2	119.0	Poly-3, 10^{-2}, 0.1	98%	57.7	110.7
8	Poly-2, 10^{-2}, 1	29%	66.8	116.4	Poly-2, 10^{-2}, 0.1	98%	19.6	75.4
9	Poly-2, 10^{-2}, 0.1	32%	19.6	130.3	Poly-2, 10^{-2}, 0.1	97%	19.6	71.5

Table 12: **Square: most-selected configuration per seed.** Offline best: Poly-2, 10^{-3}, 0.1 (offline regret 2.3).

	COMAS (TS)				COMAS (UCB)			
Seed	Configuration	Share	Δ	Regret	Configuration	Share	Δ	Regret
0	Poly-2, 10^{-3}, $0.1^{\star}$	30%	0.0	41.2	Poly-3, 10^{-2}, 0.1	58%	22.1	27.5
1	Poly-2, 10^{-1}, 0.1	21%	3.3	30.6	Poly-2, 10^{-1}, 1	73%	8.1	23.1
2	Poly-2, 10^{-3}, $0.1^{\star}$	18%	0.0	22.0	Poly-3, 10^{-3}, 0.1	58%	6.5	21.5
3	Poly-2, 10^{-2}, 0.1	19%	0.5	20.8	Poly-2, 10^{-2}, 1	80%	5.8	20.1
4	Poly-2, 10^{-1}, 1	36%	8.1	32.1	Poly-2, 10^{-1}, 0.1	89%	3.3	15.3
5	Poly-2, 10^{-3}, $1^{\dagger}$	18%	7.2	48.9	Poly-2, 10^{-3}, $0.1^{\star}$	67%	0.0	15.5
6	Poly-2, 10^{-3}, $1^{\dagger}$	18%	7.2	14.5	NN 32×2, 10^{-3}, 0.1	85%	1045.5	196.0
7	Poly-2, 10^{-2}, 0.1	19%	0.5	23.3	Poly-2, 10^{-2}, 0.1	27%	0.5	33.2
8	Poly-2, 10^{-2}, 1	51%	5.8	33.1	Poly-3, 10^{-3}, 0.1	41%	6.5	26.1
9	Poly-2, 10^{-3}, $0.1^{\star}$	42%	0.0	18.9	Poly-2, 10^{-1}, 0.1	93%	3.3	16.3

Table 13: **MOVIELENS: most-selected configuration per seed.** Offline best: Lin, 10^{-3}, 0.1 (offline regret 163.2).

	COMAS (TS)				COMAS (UCB)			
Seed	Configuration	Share	Δ	Regret	Configuration	Share	Δ	Regret
0	Lin, 10^{-1}, 1	12%	198.9	229.0	Lin, 10^{-3}, $0.1^{\star}$	70%	0.0	147.7
1	Lin, 10^{-2}, 0.01	35%	1.2	230.5	Lin, 10^{-3}, $1^{\dagger}$	99%	38.2	241.5
2	Lin, 10^{-3}, $1^{\dagger}$	19%	38.2	230.6	Lin, 10^{-1}, 0.1	98%	20.5	220.0
3	Lin, 10^{-3}, $0.01^{\dagger}$	26%	4.0	169.1	NN 8×2, 10^{-3}, 0.1	97%	898.4	335.1
4	NN 8×2, 10^{-3}, 1	35%	999.2	366.4	Lin, 10^{-2}, 0.1	96%	9.5	167.1
5	Lin, 10^{-3}, $1^{\dagger}$	34%	38.2	205.5	Lin, 10^{-1}, 0.1	98%	20.5	153.5
6	Lin, 10^{-3}, $1^{\dagger}$	31%	38.2	179.1	Lin, 10^{-2}, 0.1	96%	9.5	159.5
7	Lin, 10^{-3}, $0.1^{\star}$	25%	0.0	223.7	Lin, 10^{-1}, 0.1	87%	20.5	247.4
8	Lin, 10^{-2}, 1	29%	62.8	260.8	Poly-2, 10^{-1}, 0.1	97%	466.4	634.4
9	Lin, 10^{-1}, 0.01	13%	3.0	202.7	NN 8×2, 10^{-3}, 0.01	97%	873.3	363.5

Table 14: **SHUTTLE: most-selected configuration per seed.** Offline best: NN 64×2, 10^{-4}, 1 (offline regret 195.0).

	COMAS (TS)				COMAS (UCB)			
Seed	Configuration	Share	Δ	Regret	Configuration	Share	Δ	Regret
0	NN 64×2, 10^{-4}, $0.1^{\dagger}$	50%	540.5	404.0	NN 16×2, 10^{-4}, 1	99%	263.0	298.0
1	NN 16×2, 10^{-4}, 0.1	26%	324.0	333.0	NN 64×2, 10^{-4}, $1^{\star}$	96%	0.0	436.0
2	NN 32×2, 10^{-4}, 1	49%	62.0	373.0	NN 32×2, 10^{-4}, 0.1	98%	182.5	321.0
3	NN 16×2, 10^{-4}, 0.1	50%	324.0	285.0	NN 64×2, 10^{-4}, $0.1^{\dagger}$	99%	540.5	407.0
4	NN 32×2, 10^{-4}, 0.1	37%	182.5	381.0	NN 32×2, 10^{-4}, 0.1	98%	182.5	361.0
5	NN 32×2, 10^{-4}, 0.1	49%	182.5	354.0	NN 64×2, 10^{-4}, $0.1^{\dagger}$	98%	540.5	438.0
6	NN 64×2, 10^{-4}, $1^{\star}$	46%	0.0	459.0	NN 16×2, 10^{-4}, 0.1	99%	324.0	330.0
7	NN 16×2, 10^{-4}, 1	31%	263.0	315.0	NN 16×2, 10^{-4}, 0.1	99%	324.0	285.0
8	NN 16×2, 10^{-4}, 1	45%	263.0	317.0	NN 16×2, 10^{-4}, 0.1	99%	324.0	314.0
9	NN 32×2, 10^{-4}, 0.1	37%	182.5	342.0	NN 16×2, 10^{-4}, 1	99%	263.0	322.0

Table 15: **MNIST: most-selected configuration per seed.** Offline best: NN 128×2, 10^{-5}, 1 (offline regret 614.0).

	COMAS (TS)				COMAS (UCB)			
Seed	Configuration	Share	Δ	Regret	Configuration	Share	Δ	Regret
0	NN 64×1, 10^{-4}, 0.1	36%	141.0	1263.0	Lin, 10^{-1}, 0.01	99%	7.5	581.0
1	NN 128×2, 10^{-5}, $0.1^{\dagger}$	43%	123.0	1019.0	Lin, 10^{-3}, 0.01	99%	316.5	934.0
2	NN 64×2, 10^{-4}, 0.1	45%	125.0	1234.0	NN 64×2, 10^{-5}, 1	99%	40.0	1008.0
3	NN 64×1, 10^{-4}, 0.1	48%	141.0	1476.0	Lin, 10^{-3}, 0.01	99%	316.5	922.0
4	NN 128×1, 10^{-5}, 1	50%	1.5	1129.0	NN 128×2, 10^{-5}, $0.1^{\dagger}$	99%	123.0	1031.0
5	NN 64×1, 10^{-5}, 0.1	42%	235.0	1023.0	NN 64×2, 10^{-3}, 1	98%	693.5	3297.0
6	NN 64×2, 10^{-3}, 1	52%	693.5	3420.0	NN 64×1, 10^{-5}, 0.1	99%	235.0	920.0
7	NN 64×2, 10^{-4}, 0.1	22%	125.0	1364.0	NN 128×1, 10^{-5}, 1	99%	1.5	1120.0
8	NN 128×2, 10^{-5}, $0.1^{\dagger}$	49%	123.0	960.0	NN 64×2, 10^{-5}, 1	99%	40.0	984.0
9	NN 64×1, 10^{-4}, 1	49%	56.0	1536.0	Lin, 10^{-1}, 0.01	100%	7.5	656.0

Table 16: **DBPEDIA: most-selected configuration per seed.** Offline best: NN 64×1, 10^{-4}, 1 (offline regret 892.0).

	COMAS (TS)				COMAS (UCB)			
Seed	Configuration	Share	Δ	Regret	Configuration	Share	Δ	Regret
0	NN 64×2, 10^{-5}, 1	44%	302.0	1414.0	Lin, 10^{-1}, 0.01	98%	188.0	1086.0
1	NN 128×1, 10^{-5}, 0.1	50%	502.5	1475.0	Lin, 10^{-1}, 1	69%	7846.5	8748.0
2	NN 128×1, 10^{-5}, 0.1	49%	502.5	1445.0	NN 128×1, 10^{-5}, 0.1	98%	502.5	1510.0
3	NN 64×1, 10^{-5}, 0.1	46%	718.5	1301.0	Lin, 10^{-3}, 0.01	99%	1819.0	2692.0
4	NN 64×2, 10^{-4}, 0.1	26%	453.0	1637.0	NN 128×1, 10^{-5}, 0.1	99%	502.5	1382.0
5	NN 64×2, 10^{-4}, 0.1	26%	453.0	1640.0	Lin, 10^{-1}, 0.01	99%	188.0	1027.0
6	NN 64×2, 10^{-4}, 1	27%	271.5	1705.0	Lin, 10^{-1}, 0.01	100%	188.0	979.0
7	NN 64×2, 10^{-4}, 1	26%	271.5	1710.0	NN 128×2, 10^{-5}, 0.1	99%	540.0	1361.0
8	NN 128×1, 10^{-5}, 1	48%	86.0	1428.0	NN 128×1, 10^{-5}, 0.1	99%	502.5	1613.0
9	NN 64×2, 10^{-5}, 0.1	26%	525.0	1245.0	NN 64×1, 10^{-3}, 1	98%	1365.5	5079.0