Taming the Monster Every Context: Complexity Measure and Unified Framework for Offline-Oracle Efficient Contextual Bandits

Hao Qin HQIN@ARIZONA.EDU
University of Arizona
Chicheng Zhang CHICHENGZ@CS.ARIZONA.EDU
University of Arizona

Editors: Steve Hanneke and Tor Lattimore

Abstract

We propose an algorithmic framework, Offline Estimation to Decisions (OE2D), that reduces contextual bandit learning with general reward function approximation to offline regression. The framework allows near-optimal regret for contextual bandits with large action spaces with $O(\log T)$ calls to an offline regression oracle over T rounds, and makes $O(\log \log T)$ calls when T is known. The design of OE2D algorithm generalizes FALCON (Simchi-Levi and Xu, 2022) and its linear reward version (Xu and Zeevi, 2020, Section 4) in that it chooses an action distribution that we term "exploitative F-design" that simultaneously guarantees low regret and good coverage that trades off exploration and exploitation. Central to our regret analysis is a new complexity measure, the Decision-Offline Estimation Coefficient (DOEC), which we show is bounded in bounded Eluder dimension per-context and smoothed regret settings. We also establish a relationship between DOEC and Decision Estimation Coefficient (DEC) (Foster et al., 2021a), bridging the design principles of offline- and online-oracle efficient contextual bandit algorithms for the first time.

Keywords: offline regression oracle, contextual bandits, optimal experimental design

1. Introduction and Related Work

The online contextual bandit learning problem, a one-step version of online reinforcement learning (RL), has garnered a lot of attention due to its usage in modern applications such as online advertising, recommendations, mobile health (Tewari and Murphy, 2017; Li et al., 2010). In this problem, a learning agent at each time step t receives a context x_t from the context space $\mathcal{X}$, takes an action a_t from the action space $\mathcal{A}$, and receives the reward r_t of the action taken. The goal of the learning agent is to take actions adaptively based on its historical information, so as to learn to maximize its total reward over a time horizon of T. At the beginning, the true reward distribution associated with each context and action is unknown to the learning agent, and thus it needs to take informative actions for learning the reward function (exploration) while reaping high rewards (exploitation).

Early research in contextual bandits mainly focused on designing algorithms that search over a class of policies (Auer et al., 2003; Langford and Zhang, 2007; Agarwal et al., 2014), aiming to learn the optimal policy in the class based on data collected adaptively online. Recently, regression-based contextual bandit algorithms (Foster and Rakhlin, 2020; Simchi-Levi and Xu, 2022) emerges as a practical alternative that allows computationally-efficient implementation with impressive empirical performance (Bietti et al., 2018; Foster et al., 2020b; Foster and Krishnamurthy, 2021). Herein, the learner has access to a class of regression functions $\mathcal{F}$ that predicts reward from context and action (which approximates the ground truth reward function $f^*(x, a) = \mathbb{E}[r_t \mid x_t = x, a_t = a]$), maintains some reward function estimate $\hat{f}_t$ based on historical data, and uses $\hat{f}_t$ to guide the selection of

action a_t to balance between exploration and exploitation. A standard measure of the performance of a contextual bandit algorithm is its *regret*, i.e., the difference between the best cumulative reward achievable had we known f^* ahead of time, and the cumulative reward of the learning agent.

To design efficient contextual bandit algorithms, the research community adopted the *oracle-efficiency* framework: assuming access to some computational oracles that can solve basic regression problems and aim to design online contextual bandit algorithms that makes a small number of calls to them. Such regression oracles can be implemented by standard machine learning libraries in practice (Varoquaux et al., 2015; Paszke et al., 2019). Two types of regression oracles are of main interest. First, an *online regression oracle* receives a stream of (context, action, reward) tuples and maintains a reward predictor on the fly, such that its online prediction error is small, e.g., $o(T)$. Second, an *offline regression oracle* takes a batch of iid (context, action, reward) tuples and outputs a reward predictor that approximately minimizes its out-of-sample prediction error. Recent research has designed contextual bandit algorithms whose regret would be small as long as the error of the online or offline regression problem is small. Comparing these two types of oracles, assuming access to an offline regression oracle with low error is milder, due to: (1) the richer availability of guarantees for offline regression from statistical learning (Yang and Barron, 1999; Rakhlin et al., 2017); (2) the simplicity of algorithms in implementing an offline regression oracle (Foster et al., 2024); on the other hand, guarantees for offline-oracle efficient contextual bandit algorithms have been established in the restricted setting that the contexts are iid (Simchi-Levi and Xu, 2022; Xu and Zeevi, 2020).

Despite impressive progress, there still remain quite a few open questions in the design and analysis of oracle-efficient contextual bandit learning algorithms with general function approximation:

- There lacks a unified framework in designing offline oracle-efficient contextual bandit algorithms, especially in the presence of large action spaces. It is known that the elegant optimism principle (Abbasi-Yadkori et al., 2011; Dani et al., 2008; Russo and Van Roy, 2013) may fail to achieve a sublinear regret under general function approximation (Foster and Rakhlin, 2022, Example 3.1). Although Xu and Zeevi (2020) provides a unified framework for offline oracle-efficient contextual bandits, their method makes $O(T)$ calls to the offline regression oracle, which is known to be suboptimal – for example, in the finite action space setting, Simchi-Levi and Xu (2022) achieves the same optimal regret by making $O(\log T)$ or even $O(\log \log T)$ calls to the regression oracle. How can we design general offline oracle-efficient contextual bandit algorithms that only makes a few calls to the offline regression oracle?

- Connections between the design principles underlying online- and offline-oracle efficient algorithms is currently missing. For example, when the action space $\mathcal{A}$ is small and discrete, the inverse gap weighting (IGW) exploration strategy has been analyzed in both the online-oracle efficient algorithm of Foster and Rakhlin (2020) and the offline-oracle efficient algorithm of Simchi-Levi and Xu (2022). But their analyses look drastically different: Foster and Rakhlin (2020) views IGW as an instance of the Estimation-to-Decision (E2D) principle (Foster et al., 2021a), while Simchi-Levi and Xu (2022)'s analysis draws parallel to a policy search-based algorithm "Taming the Monster" (Agarwal et al., 2014). This raises the question: are these two facets of IGW a coincidence, or do the principles driving online- and offline-oracle efficient contextual bandit exploration actually have some connections?

In this paper, we make significant progress in answering these questions:

- We design a unified algorithm, OE2D (Offine-Estimation-to-Decision, Alg. 1), for offline regression oracle-efficient contextual bandits with general reward function classes, which generalizes FALCON (Simchi-Levi and Xu, 2022) and its linear-reward extension (Xu and Zeevi, 2020, Section 4). Key to our algorithm design is the computation of an exploration distribution that satisfies both low-regret and good-coverage properties, implicitly defined for each context x. It generalizes the pure-exploration-oriented F-design (Agarwal et al., 2024) to accommodate exploitation, and thus we name our exploration distribution "exploitative F-design".
- We establish regret guarantees of OE2D (Theorem 4), and show that it not only recovers existing guarantees of offline-oracle-efficient contextual bandit algorithms (Simchi-Levi and Xu, 2022; Xu and Zeevi, 2020) but also obtains many new guarantees. To highlight: we obtain the first offline-oracle-efficient algorithm in the h-smoothed-regret (Krishnamurthy et al., 2020) and per-context generalized linear reward settings, with $O(\log T)$ calls to the offline regression oracle, using the OE2D framework. In our analysis, we reveal a regret bound of OE2D that *simultaneously holds for every context* x, which may be of independent interest. We also show that our modular theorem statement allows OE2D to handle variants of the contextual bandits problem, such as model misspecification, reward corruption, and context distribution shifts (See Appendix D for the exact assumptions).
- Key to our regret analysis is a new complexity measure, Decision-Offline Estimation Coefficient (DOEC, Definition 2), that characterizes the statistical cost of reducing online contextual bandit learning to offline regression. Different from the Decision Estimation Coefficient (DEC, Foster et al., 2021a) which refers to the square error of the central model, DOEC's definition does not refer to the square error of the central model, which enables reduction to offline estimation. We establish new structural results of DOEC (Theorem 7), specifically relating it to a modification of the Sequential Extrapolation Coefficient (SEC) (Xie et al., 2022), which we name relaxed ε-SEC (Definition 6). We show that relaxed ε-SEC is bounded in many examples. In view that ε-SEC is a "passive" measure of exploration complexity, while DOEC allows active experimental design, we also present an example that demonstrates ε-SEC alone may not be adequate to characterize when regret-efficient contextual bandit with offline oracle is feasible.
- We establish a general theorem that links DOEC to DEC (Theorem 9): any exploration distribution p that certifies small DOEC also certifies small DEC, and DEC is at most DOEC, up to lower order terms. We believe this is an interesting observation, as this bridges the design principles of offline- and online- oracle efficient contextual bandit algorithms for the first time.

  Due to space limits, we discuss additional related work in Appendix A.

2. Preliminaries

Basic Notations Denote by $[N] := \{1, \dots, N\}$. We say $f = O\big(g\big)$ if $f \le cg$ for some constant c. We say $f = \widetilde{O}\big(g\big)$ if $f = O\big(g \cdot \mathrm{polylog}(g)\big)$. When $f = \widetilde{O}\big(g\big)$ we also write $f \lesssim g$ or $g \gtrsim f$. The convex hull of a set $\mathcal{S}$ is defined as $\mathrm{co}(\mathcal{S}) = \{\sum_i \alpha_i s_i \mid s_i \in \mathcal{S}, \alpha_i \ge 0, \sum_i \alpha_i = 1\}$. Denote $\Delta(\mathcal{Y})$ the space of all probability distributions over $\mathcal{Y}$. For nonnegative measures p, q on a common measurable space, we write $p \succeq q$ (equivalently $q \preceq p$) if $p(S) \ge q(S)$ for every measurable set S.

Basic Assumptions The learning agent has access to a class of reward functions $\mathcal{F}$ where each $f \in \mathcal{F}$ is a mapping from context-action pairs to rewards, i.e., $f : \mathcal{X} \times \mathcal{A} \to [0, 1]$. Throughout the

paper, unless otherwise specified, we assume *realizability* and no context-distribution shift unless specified otherwise, such that the ground truth reward function f^* is in $\mathcal{F}$, and the contexts are drawn iid from a distribution $\mathcal{D}_X$. We assume $\mathcal{F}$ does not have any specific structure but we give some examples of structured reward function classes after presenting the main theoretical results, including per-context linear reward model (Demirer et al., 2019; Zhu et al., 2022) and per-context generalized linear model (Xu and Zeevi, 2020), which generalizes the globally linear and generalized linear reward models (Filippi et al., 2010; Abbasi-Yadkori et al., 2011).

Main Performance Measure: Regret To provide a unified treatment on standard regret and smooothed regret (Zhu and Mineiro, 2022) in the literature, we consider a general notion of regret called Λ-Regret, which measures the performance of a contextual bandit algorithm ALG against the best action distribution per context in a predefined benchmark space of distributions $\Lambda \subset \Delta(\mathcal{A})$: $\mathrm{Regret}_\Lambda(T, \mathrm{ALG}) = \sum_{t=1}^T \mathrm{Reg}(p_t \mid x_t)$, where $\mathrm{Reg}(p \mid x) = \max_{\lambda \in \Lambda} \mathbb{E}_{a\sim\lambda}\left[f^*(x,a)\right] - \mathbb{E}_{a\sim p}\left[f^*(x,a)\right]$ is the instantaneous regret of p on context x. [1]

Running Examples. To help illustrate our results, we will frequently refer to the following three examples throughout this paper:

1. Discrete action space, standard regret (Foster and Rakhlin, 2020; Simchi-Levi and Xu, 2022). Λ is the set of all point mass distributions over the action space $\mathcal{A}$, i.e., $\Lambda = \left\{\delta_a : a \in \mathcal{A}\right\}$.

2. Per-context generalized linear reward structure (Xu and Zeevi, 2020). Let $\Lambda = \left\{\delta_a : a \in \mathcal{A}\right\}$, and assume that $\mathcal{F} = \left\{f_\theta(x,a) = \sigma\left(\phi(x,a)^\top \theta(x)\right) \mid \theta \in \Theta\right\}$, where σ is a known link function with derivative in $[\underline{L}, \overline{L}]$ for constants $0 < \underline{L} \le \overline{L}$ (and $\kappa := \overline{L}/\underline{L}$), $\phi : \mathcal{X} \times \mathcal{A} \to \mathbb{R}^d$ is a known feature map with $||\phi(x,a)||_2 \le 1$, and Θ is a class of maps $\mathcal{X} \to \mathbb{R}^d$ with per-context diameter $\sup_{\theta,\theta'\in\Theta} ||\theta(x) - \theta'(x)||_2 \le B$ for all $x \in \mathcal{X}$. Taking σ to be the identity recovers the per-context linear reward structure (Demirer et al., 2019; Zhu et al., 2022; Zhang, 2022).

3. h-smoothed regret (Krishnamurthy et al., 2020; Zhu and Mineiro, 2022) $\Lambda = \Delta_h^\mu(\mathcal{A}) := \left\{\lambda \in \Delta(\mathcal{A}) : \frac{\mathrm{d}\lambda}{\mathrm{d}\mu}(a) \le 1/h, \forall a \in \mathcal{A}\right\}$ denotes the set of all h-smoothed distributions w.r.t. μ.

Note that Examples 2 and 3 above can be viewed as orthogonal ways to generalize Example 1: Example 2 with σ being the identity function, $d = |\mathcal{A}|$, and $\phi(x,a) = e_a$, the a-th canonical basis, recovers Example 1; Example 3 with μ being the uniform distribution over $\mathcal{A}$ and $h = 1/|\mathcal{A}|$ also gives back Example 1.

Regression Oracles We consider two types of regression oracles that have been widely used in contextual bandit literature: 1) an offline regression oracle $\mathcal{O}_{\mathrm{off}}(\mathcal{F})$ takes into (context, action, reward) tuples drawn i.i.d. from distribution $\mathcal{D}$ and outputs a function $\hat{f} \in \mathcal{F}$, s.t., with probability at least $1-\delta$, $\mathbb{E}_{\mathcal{D}}\left[\left(\hat{f}(x,a) - f^*(x,a)\right)^2\right] \lesssim \mathrm{Reg}_{\mathrm{off}}(\mathcal{F}, T, \delta)$. 2) An online regression oracle $\mathcal{O}_{\mathrm{on}}(\mathcal{F})$ takes as input a sequence of (context, action, reward) tuples generated in an online manner and outputs a sequence of functions $\{f_t\}_{t=1}^T$ s.t., with probability at least $1-\delta$, $\mathbb{E}\left[\sum_{t=1}^T (\hat{f}_t(x_t,a_t) - r_t)^2 - \inf_{f\in\mathcal{F}} \sum_{t=1}^T (f^*(x_t,a_t) - r_t)^2\right] \le \mathrm{Reg}_{\mathrm{on}}(\mathcal{F}, T, \delta)$. It is well-known that when $|\mathcal{F}|$ is finite, $\mathrm{Reg}_{\mathrm{off}}(\mathcal{F}, T, \delta)$ and $\mathrm{Reg}_{\mathrm{on}}(\mathcal{F}, T, \delta)$ can be $O\left(\log |\mathcal{F}|/T\right)$ (Agarwal et al., 2012) and $O\left(\log |\mathcal{F}|\right)$ (Cesa-Bianchi and Lugosi, 2006), respectively. Since most learning algorithms have better guarantees with larger sample size, we assume that $\mathrm{Reg}_{\mathrm{off}}$ is monotonically decreasing in T and increasing in δ in general.

1. Our framework also handles the setting where the benchmark distributions are context-dependent; we present relevant results in Appendix D.4.

Decision-Estimation Coefficient (DEC) The Decision-Estimation Coefficient (Foster et al., 2021a) is a key measure to characterize the complexity of exploration in online decision making problems. Here we present a version using square loss and benchmark distribution space (Zhu and Mineiro, 2022):

Definition 1 *Given a reward function class $\mathcal{G}$ mapping from $\mathcal{A}$ to $\mathbb{R}$, benchmark distribution space Λ, $\gamma > 0$, reward function $\hat{g}$, their Decision-Estimation Coefficient (DEC) is defined as*

$$\mathsf{dec}_\gamma(\mathcal{G}, \hat{g}, \Lambda) = \inf_{p \in \Delta(\mathcal{A})} \sup_{g^* \in \mathcal{G}} \mathbb{E}_{a \sim p} \left[\max_{\lambda' \in \Lambda} \mathbb{E}_{a' \sim \lambda'}[g^*(a')] - g^*(a) - \gamma \left(\hat{g}(a) - g^*(a)\right)^2 \right] \quad (1)$$

and $\mathsf{dec}_\gamma(\mathcal{G}, \Lambda) = \max_{\hat{g} \in \mathcal{G}} \mathsf{dec}_\gamma(\mathcal{G}, \hat{g}, \Lambda)$.

It is well-known that the E2D algorithm (Foster et al., 2021a) has a small regret whenever the $\mathsf{dec}(\mathcal{F}_x, \Lambda)$ is small for every x, and the online regression oracle has a low regret: $\mathrm{Reg}(T, \mathrm{E2D}) \leq \min_{\gamma > 0}(T \max_x \mathsf{dec}_\gamma(\mathcal{F}_x, \Lambda) + \gamma \mathrm{Reg}_{\mathrm{on}}(\mathcal{F}, T, \delta))$. For the above three examples, $\max_x \mathsf{dec}_\gamma(\mathcal{F}_x, \Lambda)$ are $\lesssim \frac{|\mathcal{A}|}{\gamma}$, $\frac{d}{\gamma}$, and $\frac{1}{\gamma h}$ respectively (Foster et al., 2021a; Zhu and Mineiro, 2022). If the class $\mathcal{F}$ is finite so that setting online regression oracle to be exponential weight has $\mathrm{Reg}_{\mathrm{on}}(\mathcal{F}, T, \delta) = O(\log |\mathcal{F}|)$, E2D has regret bounds $\lesssim \sqrt{|\mathcal{A}| T \log |\mathcal{F}|}$, $\sqrt{dT \log |\mathcal{F}|}$, and $\sqrt{\frac{T}{h} \log |\mathcal{F}|}$ respectively.

Coverage between Distributions Our work will use basic tools in offline policy evaluation (OPE) for contextual bandits. In the simplest (non-contextual) structured bandit setting, OPE aims to estimate the expectation of reward function g^* over a *target action distribution* $q \in \Delta(\mathcal{A})$, by drawing noisy measurements of $g^*(a)$ for a's sampled from some *behavior action distribution* $p \in \Delta(\mathcal{A})$. Following prior works in offline reinforcement learning with function approximation (Song et al., 2022; Xie et al., 2022) and experimental design (Agarwal et al., 2024), we define the coverage of p over q with respect to function class $\mathcal{G}$ and a constant $\varepsilon > 0$ as

$$\mathsf{Coverage}_\varepsilon(p, q; \mathcal{G}) = \sup_{g, g' \in \mathcal{G}} \frac{(\mathbb{E}_{a \sim q}[g(a) - g'(a)])^2}{\varepsilon + \mathbb{E}_{a \sim p}[(g(a) - g'(a))^2]} \quad (2)$$

A smaller $\mathsf{Coverage}_\varepsilon(p, q; \mathcal{G})$ indicates that samples from p provides more information in evaluating the expected reward of distribution q. Given function class $\mathcal{G}$ and benchmark distribution Λ, The nonlinear F-design (Agarwal et al., 2024) aims to find a distribution p such that it minimizes the worst-case coverage over all possible q's in Λ: $p^* = \mathrm{argmin}_{p \in \mathrm{co}(\Lambda)} \max_{q \in \Lambda} \mathsf{Coverage}_\varepsilon(p, q; \mathcal{G})$ and we denote the optimal objective as $\mathcal{V}^*_\varepsilon(\mathcal{G}, \Lambda)$ [2].

3. The OE2D contextual bandit algorithm and its guarantees

We present our main algorithm, OE2D (Algorithm 1) that efficiently deals with large action spaces and general reward function classes in contextual bandit problems using a few calls to offline regression oracles. In addition to standard inputs, it also takes in a relaxed coverage function $\overline{\mathsf{Coverage}}$, an upper bound of Coverage; this is for computational efficiency considerations, as we will discuss shortly. The algorithm proceeds in epochs. At the beginning of each epoch $m \geq 2$, we call the offline regression oracle to obtain a reward function estimate $\hat{f}_m$ by minimizing the square loss over data collected in the previous epoch $m - 1$ (lines 4-7); we use the convention that $\hat{f}_1 \equiv 0$. During epoch

2. Our definition slightly generalizes Agarwal et al. (2024) in that we incorporate a benchmark distribution class Λ; when $\Lambda = \{\delta_a : a \in \mathcal{A}\}$ our definition become theirs.

m, at each step t, we use $\hat{f}_m$ together with the observed context x_t to construct an action distribution p_t that solves a minimax optimization problem (Eq. (3)) (line 9); we will discuss its rationale in the next paragraph. We then sample an action a_t from p_t and observe its reward r_t (line 9). This process is repeated for each round within the epoch, and the algorithm proceeds to the next epoch until the time horizon is reached.

Relaxed Exploitative F-design The two terms in the objective function in Eq. (3) encourage exploitation and exploration, respectively. Specifically, minimizing $\mathbb{E}_{a\sim\lambda}\left[\hat{f}_m(x,a)\right] - \mathbb{E}_{a\sim p}\left[\hat{f}_m(x,a)\right]$ encourages p_t to be greedy with respect to reward function $\hat{f}_m$; minimizing $\frac{1}{\gamma_m}\overline{\mathsf{Coverage}}_{\varepsilon_m}(p,\lambda;\mathcal{F}_{x_t})$ encourages p_t to provide decent coverage to all distributions $\lambda \in \Lambda$. Parameter γ_m serves as a tuning hyperparameter: a larger (resp. smaller) γ_m implies a higher degree of exploitation (resp. exploration). When $\gamma_m \to 0$, minimizing Eq. (3) amounts to minimizing $\max_{\lambda\in\Lambda}\overline{\mathsf{Coverage}}_{\varepsilon_m}(p,\lambda;\mathcal{F}_{x_t})$, which is equivalent to finding a relaxed variant of the F-design (Agarwal et al., 2024), previously proven useful for pure exploration in contextual bandits. In light of this connection, we name our optimization problem (3) "relaxed exploitative F-design" due to its additional encouragement of exploitation. Its unrelaxed counterpart, the *exploitative F-design*, in which the original coverage $\mathsf{Coverage}$ takes the place of $\overline{\mathsf{Coverage}}$, is recovered as the special case of the trivial relaxation $\overline{\mathsf{Coverage}} = \mathsf{Coverage}$. Finally, the constraint that $p_t \in \mathrm{co}(\Lambda)$ allows p_t's expected reward to be evaluated with good accuracy.

Relaxed Coverage and Computational Efficiency Recall $\mathsf{Coverage}_\varepsilon$ defined in Eq. (2): obtaining its value at even a single pair (p,λ) requires solving an optimization problem, namely a maximization over pairs of functions $g, g' \in \mathcal{G}$. For general function classes, this is a nonconcave maximization problem that can be computationally costly to solve, let alone the minimax problem (Eq. (3)) built on top of it. This motivates our definition of *relaxed coverage*, $\overline{\mathsf{Coverage}}$: a tractable upper bound of the original coverage, i.e., $\overline{\mathsf{Coverage}}_\varepsilon(p,\lambda;\mathcal{G}) \geq \mathsf{Coverage}_\varepsilon(p,\lambda;\mathcal{G})$ for all $p \in \mathrm{co}(\Lambda)$ and $\lambda \in \Lambda$. An appropriate choice of $\overline{\mathsf{Coverage}}$ allows us to design a computationally efficient algorithm while not inflating the coverage value by too much. Such relaxations exist for many settings of interest. For example, in our three running examples, we can define:

1. Discrete action space, $\overline{\mathsf{Coverage}}_\varepsilon(p,\lambda;\mathcal{F}_x) := \sum_{a\in\mathcal{A}}\frac{\lambda(a)}{p(a)}$;

2. Per-context generalized linear reward, $\overline{\mathsf{Coverage}}_\varepsilon(p,\lambda;\mathcal{F}_x) := \kappa^2\,\mathrm{tr}\left(\Sigma_p^{-1}\Sigma_\lambda\right)$, where $\Sigma_p := \mathbb{E}_{a\sim p}\left[\phi(x,a)\phi(x,a)^\top\right]$, Σ_λ is defined analogously;

3. h-smoothed regret, $\overline{\mathsf{Coverage}}_\varepsilon(p,\lambda;\mathcal{F}_x) := \frac{1}{h}\mathbb{E}_{a\sim\mu}\left[\frac{\lambda(a)}{p(a)}\right]$, where $\lambda(a)$ and $p(a)$ denote the densities of λ and p with respect to the base measure μ.

The validity of these relaxations is straightforward and we defer the detailed proof to Lemma 22 in Appendix C.2. In all three running examples, $\overline{\mathsf{Coverage}}$ is convex in p and linear in λ, which makes the induced minimax problem (Eq. (3)) efficiently solvable. Furthermore, the solution to Eq. (3) in each of the three examples coincides with that of a convex optimization problem (see Lemma 23); similar observations have been made in prior works (Kiefer and Wolfowitz, 1960; Xu and Zeevi, 2020; Levy et al., 2023). Meanwhile, the trivial relaxation $\overline{\mathsf{Coverage}} = \mathsf{Coverage}_\varepsilon$ is always valid, and remains interesting whenever the original coverage is itself tractable; for instance, in the "tabular" structured bandit setting (Lattimore and Munos, 2014; Jun and Zhang, 2020a; Tirinzoni et al., 2020) , where $\mathcal{G}$ is represented by a matrix whose rows are functions in $\mathcal{G}$ and columns are actions in $\mathcal{A}$, any coverage can be computed in time polynomial in the size of the matrix.

Algorithm 1 OE2D: Offline Estimation to Decision

Input: learning parameter $\{\gamma_m\}_{m=1}^M$, epoch schedule $0 = \tau_0 < \tau_1 < \cdots < \tau_M$, benchmark distributions Λ, reward function class $\mathcal{F}$, offline regression oracle $\mathcal{O}_{\text{off}}$, relaxed coverage $\overline{\mathsf{Coverage}}$.

for $t = 1$ **to** τ_1 /* No historical data for the first epoch, pure exploration only */ **do**

  Observe context $x_t \in \mathcal{X}$, and sample action a_t from a relaxed F-design distribution:

$$p_t = \operatorname*{argmin}_{p \in \mathrm{co}(\Lambda)} \max_{\lambda \in \Lambda} \overline{\mathsf{Coverage}}_\varepsilon(p, \lambda; \mathcal{F}_{x_t}).$$

for $m = 2$ **to** M /* Construct reward estimator via offline regression oracle */ **do**

  Compute $\hat{f}_m \leftarrow \mathcal{O}_{\text{off}}(\mathcal{F})(\{(x_i, a_i, r_i)\}_{i=\tau_{m-2}+1}^{\tau_{m-1}})$.

  for $t = \tau_m + 1$ **to** τ_{m+1} /* Construct distribution p_t which is the solution of relaxed exploitative F-design */ **do**

    Observe context $x_t \in \mathcal{X}$. Let p_t be the solution of the following problem:

$$p_t = \operatorname*{argmin}_{p \in \mathrm{co}(\Lambda)} \max_{\lambda \in \Lambda} \left(\mathbb{E}_{a \sim \lambda} \left[\hat{f}_m(x_t, a) \right] - \mathbb{E}_{a \sim p} \left[\hat{f}_m(x_t, a) \right] + \frac{\overline{\mathsf{Coverage}}_\varepsilon(p, \lambda; \mathcal{F}_{x_t})}{\gamma_m} \right). \tag{3}$$

    Sample action $a_t \sim p_t$ and observe reward r_t.

Decision-Offline Estimation Coefficient (DOEC): a New Complexity Measure The minimax value of the (relaxed) exploitative F-design (Eq. (3)) turns out to be important for establishing the regret guarantee of OE2D, which we formally introduce as follows:

Definition 2 (DOEC and Relaxed DOEC) *Given a class of functions $\mathcal{G} : \mathcal{A} \to [0, 1]$, a class of benchmark distributions $\Lambda \subset \Delta(\mathcal{A})$, a $\hat{g} \in \mathcal{G}$, their* Decision-Offline-Estimation Coefficient (DOEC) *is:*

$$\mathsf{doec}_{\gamma,\varepsilon}(\hat{g}, \mathcal{G}, \Lambda) = \min_{p \in \mathrm{co}(\Lambda)} \max_{\lambda \in \Lambda} \left(\mathbb{E}_{a \sim \lambda} \left[\hat{g}(a) \right] - \mathbb{E}_{a \sim p} \left[\hat{g}(a) \right] + \tfrac{1}{\gamma} \mathsf{Coverage}_\varepsilon(p, \lambda; \mathcal{G}) \right),$$

and $\mathsf{doec}_{\gamma,\varepsilon}(\mathcal{G}, \Lambda) = \max_{\hat{g} \in \mathcal{G}} \mathsf{doec}_{\gamma,\varepsilon}(\hat{g}, \mathcal{G}, \Lambda)$. *Given a choice of relaxed coverage $\overline{\mathsf{Coverage}}$, we define its associated relaxed DOEC, $\overline{\mathsf{doec}}_{\gamma,\varepsilon}(\hat{g}, \mathcal{G}, \Lambda)$ and $\overline{\mathsf{doec}}_{\gamma,\varepsilon}(\mathcal{G}, \Lambda)$ to be their unrelaxed counterparts with* Coverage *replaced by* $\overline{\mathsf{Coverage}}$.

With $\mathcal{G} = \mathcal{F}_{x_t}$ and $\hat{g} = \hat{f}_m(x_t, \cdot)$, $\overline{\mathsf{doec}}_{\gamma_m,\varepsilon_m}(\hat{g}, \mathcal{G}, \Lambda)$ is exactly the minimax value of the relaxed exploitative F-design (Eq. (3)). Since any valid relaxed coverage bounds the original coverage, their associated relaxed DOEC upper bounds the original DOEC as well. And thus, any minimizer p of the relaxed DOEC optimization problem *certifies* an upper bound of the original DOEC.

Our name "Decision-Offline-Estimation Coefficient" is inspired by DEC (Definition 1). Similar to DEC, the DOEC objective consists of a suboptimality term of decision p, $\mathbb{E}_{a \sim \lambda} \left[\hat{g}(a) \right] - \mathbb{E}_{a \sim p} \left[\hat{g}(a) \right]$, and an estimation term $\frac{1}{\gamma} \mathsf{Coverage}_\varepsilon(p, \lambda; \mathcal{G})$, measuring the usefulness of p in estimating the ground truth reward function. We also observe some key differences: DEC takes an additional maximum over all possible ground truth reward functions g^*, and the estimation cost is measured by the central model $\hat{g}$'s error $\mathbb{E}_{a \sim p} \left[(\hat{g}(a) - g^*(a))^2 \right]$. This error term enables a reduction from contextual bandits

to *online* regression, and it is not clear how we can use it to reduce contextual bandits to offline regression. In contrast, DOEC measures the estimation overhead in terms of coverage, which obviates the reference to the central model $\hat{g}$, enabling reduction to offline estimation. For the three running examples, with the relaxed coverages introduced above, we show in Lemma 23 that $\overline{\mathsf{doec}}_{\gamma,\varepsilon}(\mathcal{F}_x, \Lambda)$ is equal to $\frac{|\mathcal{A}|}{\gamma}$, $\frac{\kappa^2 d}{\gamma}$, and $\frac{1}{\gamma h}$, respectively.

We now come back to explain the utility of the relaxed exploitative F-design (Eq. (3)). Define $\widehat{\mathrm{Reg}}_m(p \mid x) = \max_{\lambda' \in \Lambda} \mathbb{E}_{a' \sim \lambda'}\left[\hat{f}_m(x, a')\right] - \mathbb{E}_{a \sim p}\left[\hat{f}_m(x, a)\right]$ to be the *empirical regret* of distribution p in context x, according to reward estimate $\hat{f}_m$. With this notation, the objective function in Eq. (3) can be equivalently written as the sum of two nonnegative terms [3]:

$$V_m(x_t, p) = \widehat{\mathrm{Reg}}_m(p \mid x_t) + \max_{\lambda \in \Lambda} \left(\frac{1}{\gamma_m} \overline{\mathsf{Coverage}}(p, \lambda; \mathcal{F}_{x_t}) - \widehat{\mathrm{Reg}}_m(\lambda \mid x_t) \right). \tag{4}$$

The choice of p_t makes the objective bounded by $\overline{\mathsf{doec}}_{\gamma_m, \varepsilon_m}(\mathcal{F}_{x_t}, \Lambda)$, and thus each of the two terms is also bounded by the same number. Since relaxed coverage upper bounds the original coverage, this suggests that p_t achieves an exploration-exploitation tradeoff in a precise sense:

Lemma 3 *$p_t \in \mathrm{co}(\Lambda)$ satisfies the following two properties simultaneously:*

$$\widehat{\mathrm{Reg}}_m(p_t \mid x_t) \le \overline{\mathsf{doec}}_{\gamma_m, \varepsilon_m}(\mathcal{F}_{x_t}, \Lambda) \qquad \textit{Low Regret (LR)}$$

$$\mathsf{Coverage}_{\varepsilon_m}(p_t, \lambda; \mathcal{F}_{x_t}) \le \gamma_m \overline{\mathsf{doec}}_{\gamma_m, \varepsilon_m}(\mathcal{F}_{x_t}, \Lambda) + \gamma_m \widehat{\mathrm{Reg}}_m(\lambda \mid x_t), \forall \lambda \in \Lambda, \quad \textit{Good Coverage (GC)}$$

Specifically, LR asserts that p_t nearly maximizes the reward estimate $\hat{f}_m(x_t, \cdot)$ among all distributions in Λ. Meanwhile, GC states that p_t "covers" all distributions λ in the benchmark class Λ well in a heterogeneous manner, with "greedier" distributions covered better. Simchi-Levi and Xu (2022) and Xu and Zeevi (2020, Section 4) were the first to observe the existence of such a distribution for discrete action spaces and action spaces with linear reward structure, respectively, given access to a reward regressor, which enables subsequent regret analysis similar to analysis of policy-search-based algorithms (Agarwal et al., 2014). Here we generalize their findings to general reward function classes and benchmark distribution classes. We now present the regret guarantee for OE2D:

Theorem 4 *Suppose $\tau_m \ge 2\tau_{m-1}$ for all m. Let $\delta_m = \frac{\delta}{m(m+1)}$. For any $\delta \in (0,1)$, $\Lambda \subseteq \Delta(\mathcal{A})$, with probability at least $1-\delta$, the Λ-Regret of OE2D is bounded as: $\mathbb{E}\left[\mathrm{Regret}_\Lambda(T, \textit{OE2D})\right] \le$*

$$\widetilde{O}\left(\tau_1 + \max_{m \in \{2,\dots,M\}} \frac{\tau_m}{\gamma_m} \cdot \left(\begin{array}{l} \max_{n \in [M]} \gamma_n \mathbb{E}\left[\overline{\mathsf{doec}}_{\gamma_n, \varepsilon_n}(\mathcal{F}_x, \Lambda)\right] \\ \qquad + \max_{n \in \{2,\dots,M\}} \gamma_n^2 \left(\mathrm{Reg}_{\mathrm{off}}(\mathcal{F}, \tau_{n-1}/2, \delta_n) + \varepsilon_{n-1}\right) \end{array} \right) \right),$$

where we use the convention that $\gamma_1 = 0$ and $\gamma_1 \overline{\mathsf{doec}}_{\gamma_1, \varepsilon_1}(\mathcal{F}_x, \Lambda)$ is $\lim_{\gamma \to 0} \gamma \overline{\mathsf{doec}}_{\gamma, \varepsilon_1}(\mathcal{F}_x, \Lambda) = \overline{\mathcal{V}}^*_{\varepsilon_1}(\mathcal{F}_x, \Lambda)$, the value of the relaxed F-design problem solved in the first epoch.

The regret bound can be understood as follows: in the first epoch, OE2D does pure exploration, and thus incurs a regret of τ_1 at most; in the subsequent epochs $m \ge 2$, at every time step, OE2D has an instantaneous regret of

$$\widetilde{O}\left(\frac{1}{\gamma_m} \left(\max_{n \in [m]} \gamma_n \mathbb{E}\left[\overline{\mathsf{doec}}_{\gamma_n, \varepsilon_n}(\mathcal{F}_x, \Lambda)\right] \right) + \max_{n \in \{2,\dots,m\}} \gamma_n^2 \left(\mathrm{Reg}_{\mathrm{off}}(\mathcal{F}, \tau_{n-1} - \tau_{n-2}, \delta_n) + \varepsilon_{n-1}\right) \right)$$

3. To see why the second term is nonnegative, note that we can choose $\lambda = \mathrm{argmax}_{\lambda' \in \Lambda} \mathbb{E}_{a' \sim \Lambda} \hat{f}_m(x, a')$, so that $\widehat{\mathrm{Reg}}_m(\lambda \mid x_t) = 0$.

which depends on the average context-wise complexity of exploration $\mathbb{E}\left[\overline{\mathsf{doec}}_{\gamma_n,\varepsilon_n}(\mathcal{F}_x,\Lambda)\right]$ and historical reward function estimates' qualities $\mathrm{Reg}_{\mathrm{off}}(\mathcal{F},\tau_{n-1}/2,\delta)$. The $\max_{n\in[m]}$ and $\max_{n\in\{2,\dots,m\}}$ operators come from the interdependence of the reward estimator and data collection in historical epochs: $\hat{f}_n$ is trained from data collected in epoch $n-1$, which in turn is determined by $\hat{f}_{n-1}$, etc. [4]

Remark 5 (OE2D with inexact minimizers) *Theorem 4 continues to hold when the relaxed exploitative F-design problem is solved inexactly: as long as for all t, $V_m(x_t,p_t)\le \overline{V}_{\gamma_m,\varepsilon_m}(\mathcal{F}_{x_t},\Lambda)$ for some function $\overline{V}$, its regret guarantee will still hold with $\overline{\mathsf{doec}}$ replaced by $\overline{V}$. The proof of Theorem 4 carries over verbatim by replacing $\overline{\mathsf{doec}}$ by $\overline{V}$. We will use this fact in Section 5.*

We now discuss the implication of this theorem in our three running examples. For simplicity, we assume a finite function class $\mathcal{F}$ and the offline regression oracle performs ERM; in this case, $\mathrm{Reg}_{\mathrm{off}}(\mathcal{F},\tau_{m-1}/2,\delta)=\widetilde{O}\left(\frac{\log|\mathcal{F}|}{\tau_{m-1}}\right)$. In addition, all three running examples satisfy that $\max_x \overline{\mathsf{doec}}_{\gamma,\varepsilon}(\mathcal{F}_x,\Lambda)=O\left(\frac{D}{\gamma}\right)$, using the relaxed coverages introduced at the beginning of this section, for $D=|\mathcal{A}|,\kappa^2 d,1/h$ respectively (see Appendix C.3 for the relaxed DOEC bounds).

And thus:

- Setting $\tau_m=2^m$, $\gamma_m=\sqrt{D/\mathrm{Reg}_{\mathrm{off}}(\mathcal{F},\tau_{m-1}/2,\delta_m)}$, $\varepsilon_m=1/T$ gives a regret bound of $\widetilde{O}\left(\sqrt{DT\log|\mathcal{F}|}\right)$ (Theorem 17).
- If the total horizon T is known, we use a *small-epoch schedule* (Theorem 18) with $M=\log\log(T)$, $\tau_m=\left\lfloor 2T^{1-2^{-m}}\right\rfloor$, $\gamma_m=\sqrt{D/\mathrm{Reg}_{\mathrm{off}}(\mathcal{F},\tau_{m-1}-\tau_{m-2},\delta_m)}$, $\varepsilon_m=1/T$ to achieve the same regret guarantee (Theorem 19).

See Appendix B.6 for the detailed calculation of these regret bounds.

Specifically, instantiating the relaxed coverage earlier in this section in each running example turns OE2D to a tractable and offline-oracle-efficient algorithm, recovering and extending a range of known guarantees:

- In the discrete action space setting, we recover FALCON (Simchi-Levi and Xu, 2022) with regret $\widetilde{O}\left(\sqrt{|\mathcal{A}|T\log|\mathcal{F}|}\right)$.
- In the per-context linear reward setting, we recover LINEAR-FALCON (Xu and Zeevi, 2020, Section 4) with regret $\widetilde{O}\left(\sqrt{dT\log|\mathcal{F}|}\right)$. For the per-context generalized linear reward setting, we obtain a new algorithm, which we name GLM-OE2D. It has a regret bound of $\widetilde{O}\left(\kappa\sqrt{dT\log|\mathcal{F}|}\right)$, which matches that of UCCB (Xu and Zeevi, 2020); however, its number of calls to the offline regression oracle is $O(\log T)$, which is much smaller.
- In the h-smoothed setting, we obtain SMOOTHED-OE2D, which has regret $\widetilde{O}\left(\sqrt{T/h\log|\mathcal{F}|}\right)$; this matches the regret guarantee of SmoothIGW (Zhu and Mineiro, 2022), an online-oracle-efficient algorithm and is information-theoretically optimal (Krishnamurthy et al., 2020, Remark under Theorem 3). This is the first time that an optimal h-smoothed regret bound is attained by an offline regression oracle-efficient algorithm. In Appendix G, we empirically show that SMOOTHED-OE2D has competitive performance with SmoothIGW.

Proof Idea of Theorem 4. Our analysis takes a "virtual policy" view of OE2D: at epoch m, OE2D implicitly uses a policy π_m to do exploration, whose definition is based on the estimated reward function $\hat{f}_m$. We then show that the cumulative regret of the sequence of policies on every context x

4. For simplicity of presentation, we further relax $\max_{n\in[m]}$ to $\max_{n\in[M]}$ and $\max_{n\in\{2,\dots,m\}}$ to $\max_{n\in\{2,\dots,M\}}$ when presenting the regret bound in Theorem 4.

can be precisely controlled (Lemma 15). Theorem 4 follows immediately by taking expectation over $x \sim \mathcal{D}_X$ and using the linearity of expectation. To this end, we establish per-context concentration bounds on the empirical regret to the true regret, for any action distribution in Λ (Lemma 12). We think our analysis may be of independent interest: this is in contrast to most previous analysis on offline-regression-oracle efficient contextual bandit algorithms (Simchi-Levi and Xu, 2022), which establish low-regret and good-coverage guarantees at the policy and population level (averaged across all x's). We speculate that this lemma may have practical usages, e.g., for guaranteeing treatment quality for every patient. We defer the full proof to Appendix B.

Extensions: Misspecification, Corruption, and Distribution Shifts Beyond the realizable and iid context setting, we show that our general regret theorem can be used to handle several variants, with all details defered to Appendix D:

1. In the model misspecification setting (Krishnamurthy et al., 2021) with a misspecification level B defined as $\sup_{p:\mathcal{X}\to\Delta(\mathcal{A})} \inf_{f\in\mathcal{F}} \mathbb{E}_{x\sim\mathcal{D},a\sim p}[(f(x,a)-f^\star(x,a))^2]$, with appropriate setting of its parameters, the regret bound of Algorithm 1 generalizes the state-of-the-art discrete action space result of Krishnamurthy et al. (2021) to per-context linear model and smoothed regret settings (Theorem 26).

2. In the corruption setting (e.g., Kapoor et al., 2019), an adversary is allowed to corrupt the observed reward in at most C rounds. In this case, the regret bound of Algorithm 1 achieves $\widetilde{O}\big(\sqrt{TD(C+\log(|\mathcal{F}|/\delta)}\big)$ (Theorem 27). This matches the results of deploying Foster et al. (2020a)'s approach to this corruption setting using online regression oracles.

3. In the context-distribution shift setting, Algorithm 1 achieves $\widetilde{O}\big(\sqrt{A^3TD\log(|\mathcal{F}|/\delta)}\big)$ regret (Theorem 28), where $A \geq 1$ measures the worst-case density ratio between the context distribution at every time step and some base context distribution (see Assumption 4). To the best of our knowledge, this is the first result showing offline-oracle efficient algorithms can work beyond the iid context setting.

4. Efficient Algorithms for Finding Exploration Strategies Certifying Low Relaxed DOEC

So far, we have shown that a small relaxed DOEC leads to offline-regression-oracle efficient algorithms with low regret. This naturally raises the question: under what conditions is the relaxed DOEC small, and how do we efficiently find exploration strategies that certify this? In this section, we develop structural results that provide general tools for identifying problem settings in which the relaxed DOEC admits favorable bounds, along with an efficient algorithm for computing exploration strategies certifying small relaxed DOEC.

Our key finding in this section is that, a new complexity measure of the reward function class $\mathcal{G}$ and the benchmark class of distributions Λ named *ε-sequential extrapolation coefficient (ε-SEC)* (together with its relaxed counterpart) can be used to bound the (relaxed) DOEC, and the boundedness of ε-SEC is grounded in several useful examples, including our three running examples above. Moreover, our proof is constructive: we give an efficient coordinate descent-based algorithm (Algorithm 2) that finds a distribution certifying it. The guarantee holds for any relaxed coverage satisfying a short list of structural properties (Assumption 5), which we distill from the convergence analysis of Algorithm 2 and ground in our three running examples as well as the original coverage.

Definition 6 (Relaxed ε-SEC) *For a class of functions $\mathcal{G}$ from $\mathcal{A}$ to $[0,1]$, and a family of benchmark distributions Λ, and $\varepsilon > 0$, and a relaxed coverage $\overline{\mathsf{Coverage}}_\varepsilon$ (Section 3; extended to take an unnormalized nonnegative measure as its first argument with cushion parameter ε), their* relaxed ε-sequential extrapolation coefficient, $\overline{\mathsf{SEC}}_\varepsilon(\mathcal{G}, \Lambda)$ *is defined as (here $\lambda_{1:i} := \sum_{j=1}^{i} \lambda_j$):*

$$\overline{\mathsf{SEC}}_\varepsilon(\mathcal{G}, \Lambda) = \sup_{N \in \mathbb{N}} \sup_{\lambda_1, \dots, \lambda_N \in \Lambda} \sum_{i=1}^{N} \overline{\mathsf{Coverage}}_{N\varepsilon}(\lambda_{1:i}, \lambda_i; \mathcal{G}) \tag{5}$$

Under the trivial relaxation $\overline{\mathsf{Coverage}}_\varepsilon = \mathsf{Coverage}_\varepsilon$, we call the resulting quantity the (unrelaxed) ε-SEC, *denoted by* $\mathsf{SEC}_\varepsilon(\mathcal{G}, \Lambda)$.

Since $\overline{\mathsf{Coverage}}_{N\varepsilon} \geq \mathsf{Coverage}_{N\varepsilon}$ pointwise, $\overline{\mathsf{SEC}}_\varepsilon(\mathcal{G}, \Lambda) \geq \mathsf{SEC}_\varepsilon(\mathcal{G}, \Lambda)$ for any relaxed coverage. Our definition of ε-SEC is inspired by SEC (Xie et al., 2022), which was useful to provide a unified analysis of optimism-based RL algorithms with low coverability and small Bellman Eluder dimension (Jin et al., 2021). There are a few important differences: first, our "cushion parameter" $N\varepsilon$ for coverage is proportional to N, the number of terms, whereas SEC has a fixed regularization parameter 1; second, we measure coverage of $\lambda_{1:i}$ on λ_i, whereas SEC measures the coverage of $\lambda_{1:i-1}$ to λ_i. Such "off-by-1" adjustment turns out to be important for constructing a useful exploration distribution that certifies a low DOEC, which can provide regret guarantees where optimism-based approaches are not adequate (Foster and Rakhlin, 2022, Example 3.1).

Examples of small relaxed ε-SEC We instantiate the relaxed ε-SEC in our three running examples. It turns out that our relaxed coverages defined before were insufficient, and we need to slightly adjust them with suitable "cushions":

1. Discrete action space: $\overline{\mathsf{Coverage}}_\varepsilon(p, \lambda; \mathcal{G}) = \sum_{a \in \mathcal{A}} \frac{\lambda(a)}{p(a) + \varepsilon/|\mathcal{A}|}$ gives $\overline{\mathsf{SEC}}_\varepsilon(\mathcal{G}, \Lambda) \leq |\mathcal{A}| \log(1 + |\mathcal{A}|/\varepsilon)$;
2. Per-context (generalized) linear reward: with $\varepsilon' := \varepsilon/(\underline{L}^2 B^2)$, $\overline{\mathsf{Coverage}}_\varepsilon(p, \lambda; \mathcal{G}) = \kappa^2 \operatorname{tr}((\Sigma_p + \varepsilon' I)^{-1} \Sigma_\lambda)$ gives $\overline{\mathsf{SEC}}_\varepsilon(\mathcal{G}, \Lambda) \leq \kappa^2 d \log(1 + 1/\varepsilon')$; here, $\Sigma_p := \sum_{a \in \mathcal{A}} p(a) \phi(a) \phi(a)^\top$.
3. h-smoothed regret: $\overline{\mathsf{Coverage}}_\varepsilon(p, \lambda; \mathcal{G}) = \frac{1}{h} \mathbb{E}_{a \sim \mu}\left[\frac{\lambda(a)}{p(a) + \varepsilon}\right]$ gives $\overline{\mathsf{SEC}}_\varepsilon(\mathcal{G}, \Lambda) \leq \frac{1}{h} \log(1 + 1/(h\varepsilon))$.

We verify the validity of these relaxed coverages (Lemma 38, Appendix E.3) and prove the relaxed SEC bounds above (Lemma 36, Appendix E.2.2).

An efficient algorithm for certifying small relaxed DOEC We give a coordinate descent-based algorithm, Algorithm 2, to find a distribution p^* certifying that $\overline{\mathsf{doec}}_{\gamma,\varepsilon}(\hat{g}, \mathcal{G}, \Lambda) \lesssim \frac{1}{\gamma} \overline{\mathsf{SEC}}_\varepsilon(\mathcal{G}, \Lambda)$, and show its termination guarantees. To this end, we first use our observation before that the relaxed DOEC objective (Definition 2) is equivalent to $\widehat{R}(p) + \max_{\lambda \in \Lambda} \left(\frac{1}{\gamma} \overline{\mathsf{Coverage}}_\varepsilon(p, \lambda; \mathcal{G}) - \widehat{R}(\lambda) \right)$, where $\widehat{R}(p) := \max_{\lambda' \in \Lambda} \mathbb{E}_{a \sim \lambda'}[\hat{g}(a)] - \mathbb{E}_{a \sim p}[\hat{g}(a)]$. It suffices to find p such that both terms are $\lesssim \frac{1}{\gamma} \overline{\mathsf{SEC}}_\varepsilon(\mathcal{G}, \Lambda)$ simultaneously; following the convention in the preceding section, we name the two requirements (LR) and (GC).

Algorithm 2 first computes the greedy distribution $\widehat{\lambda}$ that maximizes the estimated reward function $\hat{g}$ over Λ (line 1) for later use in the final output. It maintains a nonnegative measure p_t over time, with p_0 initialized to be the zero measure (line 2). Then, at each iteration t, it first finds the distribution λ_t that maximizes the amount of violation of the (GC) property. We extend the definition of Coverage (and likewise $\overline{\mathsf{Coverage}}$) so that it can take any nonnegative p (not necessarily normalized) as the

Algorithm 2 Finding p that certifies small relaxed doec

Input: benchmark distributions Λ, reward function class $\mathcal{G}$, estimated reward function $\hat{g} : \Lambda \to [0,1]$, relaxed coverage $\overline{\mathsf{Coverage}}$ satisfying Assumption 5 with step-size threshold $\bar{\Delta}$, step sizes $\Delta_t \in (0, \bar{\Delta}]$, parameters $\gamma > 0$ and $\varepsilon \in (0,1)$.

Output: a distribution $p^* \in \Lambda$ that certifies $\overline{\mathsf{doec}}_{\gamma,\varepsilon}(\mathcal{G}, \Lambda) \le \frac{10}{\gamma}\overline{\mathsf{SEC}}_{\varepsilon}(\mathcal{G}, \Lambda)$, i.e., Eq. (6) holds.

1 Compute greedy distribution $\widehat{\lambda} = \operatorname{argmax}_{\lambda \in \Lambda} \mathbb{E}_{a\sim\lambda}[\hat{g}(a)]$, and denote $\widehat{R}(p) := \mathbb{E}_{a\sim\widehat{\lambda}}[\hat{g}(a)] - \mathbb{E}_{a\sim\lambda}[\hat{g}(a)]$

2 Let $p_0 := 0$. /* Initialize with zero distribution */

3 **for** $t = 1, 2, \ldots$ **do**

4 Compute $\lambda_t = \operatorname{argmax}_{\lambda\in\Lambda} \left[\mathbb{E}_{a\sim\lambda}[\hat{g}(a)] + \frac{1}{\gamma}\overline{\mathsf{Coverage}}_{\varepsilon}(p_{t-1}, \lambda; \mathcal{G})\right]$,

5 **if** $-\widehat{R}(\lambda_t) + \frac{1}{\gamma}\overline{\mathsf{Coverage}}_{\varepsilon}(p_{t-1}, \lambda_t; \mathcal{G}) > \frac{8}{\gamma}\overline{\mathsf{SEC}}_{\varepsilon}(\mathcal{G}, \Lambda)$ **then**

6 Run a coordinate descent step: $p_t \leftarrow p_{t-1} + \Delta_t \lambda_t$, /* Update p_{t-1} when not satisfying Eq.(6) */

7 **else**

8 $t_0 \leftarrow t-1$, **return** $p^* = p_{t_0} + (1 - \|p_{t_0}\|_1)\widehat{\lambda}$ /* Terminate and return the final distribution */

covering measure[5]. Then it checks whether the violation of the GC property (on λ_t) exceeds a threshold (line 5). If so, it performs a coordinate descent update along the direction of λ_t (line 6). Otherwise, (GC) is satisfied, it terminates the iteration and returns the final distribution p^*, which is a convex combination of p_{t_0} and the greedy distribution $\widehat{\lambda}$ (line 8).

Algorithm 2 is inspired by previous works in computing exploration distributions over a class of policies (Agarwal et al., 2014), and linear reward structure setting (Xu and Zeevi, 2020, Section 4). Different from these works, we make a few key adjustments: 1) we do not scale the distributions during updates to ensure the (LR) property more easily at each iteration; 2) we initialize p_0 as the zero vector rather than a normalized probability measure. A key step in proving Theorem 7 is to show that our iterates p_t's are always sub-probability distributions and can be extended to a valid probability distribution $p^* \in \mathrm{co}(\Lambda)$.

We show that as long as $\overline{\mathsf{Coverage}}_{\varepsilon}$ satisfies a few structural properties, Algorithm 2 can efficiently find a distribution certifying that $\overline{\mathsf{doec}}_{\gamma,\varepsilon}(\mathcal{G}, \Lambda) \lesssim O\big(\frac{1}{\gamma}\overline{\mathsf{SEC}}_{\varepsilon}(\mathcal{G}, \Lambda)\big)$. We call such an $\overline{\mathsf{Coverage}}_{\varepsilon}$ an *admissible* relaxed coverage (formal statement can be found in Assumption 5 in Appendix E). Importantly, the admissibility assumption is grounded in our "cushioned" relaxed coverages defined for the three running examples, as well as the original unrelaxed coverage. We defer these checks to Lemma 38 in Appendix E.3.

The following theorem is the main result of this section; it shows that given any admissible relaxed coverage, Algorithm 2 efficiently finds a distribution certifying that $\overline{\mathsf{doec}}_{\gamma,\varepsilon}(\mathcal{G}, \Lambda)$ is at most $\frac{1}{\gamma}\overline{\mathsf{SEC}}_{\varepsilon}(\mathcal{G}, \Lambda)$ up to a constant factor; we defer its proof to Appendix E.

Theorem 7 *For any reward function class $\mathcal{G} : \mathcal{A} \to [0,1]$, benchmark distribution class Λ, relaxed coverage $\overline{\mathsf{Coverage}}$ satisfying Assumption 5 with step-size threshold $\bar{\Delta}$, $\gamma > 0$, and $\varepsilon \in (0,1)$, Algorithm 2 with step size $\Delta_t = \bar{\Delta}$ terminates within $\lfloor 1/\bar{\Delta} \rfloor$ iterations and outputs a distribution $p^* \in \mathrm{co}(\Lambda)$ such that*

$$\max_{\lambda\in\Lambda}\left(\mathbb{E}_{a\sim\lambda}[\hat{g}(a)] - \mathbb{E}_{a\sim p^*}[\hat{g}(a)] + \frac{1}{\gamma}\overline{\mathsf{Coverage}}_{\varepsilon}(p^*, \lambda; \mathcal{G})\right) \le \frac{10}{\gamma}\overline{\mathsf{SEC}}_{\varepsilon}(\mathcal{G}, \Lambda). \tag{6}$$

5. We still require q to be normalized probability measures.

Applying Theorem 7 with the relaxed coverages with cushion ε of Lemma 38, Algorithm 2 terminates within $O\big(\frac{|\mathcal{A}|}{\varepsilon}\big)$, $O\big(\frac{L^2B^2}{\varepsilon}\big)$, and $O\big(\frac{1}{h\varepsilon}\big)$ iterations in the three running examples, and certifies relaxed DOEC bounds of $O\big(\frac{|\mathcal{A}|}{\gamma}\log\frac{|\mathcal{A}|}{\varepsilon}\big)$, $O\big(\frac{\kappa^2 d}{\gamma}\log\frac{L^2B^2}{\varepsilon}\big)$, and $O\big(\frac{1}{\gamma h}\log\frac{1}{h\varepsilon}\big)$, respectively – matching the bounds given by the certificates obtained via convex optimization (Lemma 23) up to a logarithmic factor. The added value of Theorem 7 is its generality: it provides a certifying procedure for *any* admissible relaxed coverage, including the original coverage, for which no closed-form or convex-optimization-based certificate is available. We also remark that the quantity $\overline{\mathsf{SEC}}_\varepsilon(\mathcal{G},\Lambda)$ in line 5 of Algorithm 2 may be replaced by any upper bound S of it (e.g., the explicit bounds in Lemma 36); the resulting algorithm then outputs p^* that certifies $\overline{\mathsf{doec}}_{\gamma,\varepsilon}(\hat{g},\mathcal{G},\Lambda)\le\frac{10S}{\gamma}$.

Specialized to the pure-exploration setting $\gamma\to 0$ and the trivial relaxation $\overline{\mathsf{Coverage}}_\varepsilon=\mathsf{Coverage}_\varepsilon$, Theorem 7 implies that $\mathcal{V}_\varepsilon(\mathcal{G},\Lambda)\lesssim\mathsf{SEC}_\varepsilon(\mathcal{G},\Lambda)$; this generalizes (Agarwal et al., 2024, Theorem 4.2) to allow general benchmark distribution class Λ. Our result is more general in that we allow to bound $\mathsf{doec}_{\gamma,\varepsilon}(\mathcal{G},\Lambda)$ with arbitrary $\gamma>0$, making it a good fit for the regret minimization setting.

The proof of Theorem 7 can be found in Appendix E. Its key idea is to construct a potential function that linearly combines a (negative) sequential extrapolation error term and a regret term, and show that the potential function decreases steadily while respecting a lower bound. The sequential extrapolation error serves as the same role as log barrier or log-determinant barrier regularizers in previous works (Agarwal et al., 2014; Xu and Zeevi, 2020). In Appendix E.4, we show additionally that when $\Lambda=\{\delta_a:a\in\mathcal{A}\}$, we can solve Eq. (6) using Algorithm 2 with a more aggressive step size and a much lower $O\left(\mathsf{SEC}_\varepsilon(\mathcal{G},\Lambda)\right)$ iterations; this generalizes (Xu and Zeevi, 2020, Section 4) to nonlinear reward function classes.

As an example, for the per-context generalized linear setting, a single call of Algorithm 2 runs in $O\big(|\mathcal{A}|\,\kappa^2 d^3\big)$ time; see Appendix E.5 for a full account and an end-to-end comparison to the UCCB algorithm of Xu and Zeevi (2020).

4.1. Bounding DEOC beyond ε-SEC: the importance of active exploration

Despite Theorem 7's generality, we observe that the (relaxed) ε-SEC has a worst-case nature: it evaluates the maximum extrapolation error in all sequences. However, the definition of DOEC allows *active experimental design*: we can choose action distribution $p\in\mathrm{co}(\Lambda)$ that balances exploration and exploitation. Does that mean Theorem 7 alone may be insufficient to tightly characterize DOEC? We show in the following proposition that this is indeed the case:

Proposition 8 *Let $\Lambda=\{\delta_a:a\in\mathcal{A}\}$. For any $k\ge 1$, there exists $\mathcal{A}$ and $\mathcal{G}$ such that $\mathsf{doec}_{\gamma,\varepsilon}(\mathcal{G},\Lambda)\lesssim\sqrt{\frac{k}{\gamma}}+\frac{k}{\gamma}$ whereas $\mathsf{SEC}_\varepsilon(\mathcal{G},\Lambda)\ge\min\left(2^{k-2},\frac{1}{2\varepsilon}\right)$.*

We defer the proof and detailed discussions to Appendix E.6 – the proof is based on a "cheating code" structured bandit instance (e.g., Agarwal et al., 2024). For that instance, Theorem 7 at best can give an upper bound on $\mathsf{doec}_{\gamma,\varepsilon}(\mathcal{G},\Lambda)$ of $\Omega\big(\min\big(\frac{2^k}{\gamma},\frac{1}{\varepsilon\gamma}\big)\big)$, much greater than its actual value. When plugging this bound into Theorem 4, this gives a regret bound at least $\min(T,\sqrt{2^k})$ for OE2D, which is vacuous when $k=\Omega\big(\log T\big)$. In contrast, using the tight bound $\mathsf{doec}_{\gamma,\varepsilon}(\mathcal{G},\Lambda)\lesssim\sqrt{\frac{k}{\gamma}}+\frac{k}{\gamma}$ gives a nontrivial regret bound of $\widetilde{O}\big((k\ln|\mathcal{F}|)^{1/3}T^{2/3}\big)$ for OE2D. In summary, this example reveals that using worst-case quantities such as SEC to characterize DOEC may not be adequate, and we leave finer structural characterizations of DOEC as an interesting open question.

5. Connecting DOEC to Contextual Bandits with Online Regression Oracles

We bridge DOEC with DEC, a primary statistical complexity measure for contextual bandit with online regression oracles. Our main finding is that any exploration distribution that certifies small DOEC should also certify a small DEC:

Theorem 9 *For any function class $\mathcal{G} : \mathcal{X} \times \mathcal{A} \to [0,1]$, any set of action distributions $\Lambda \subseteq \Delta(\mathcal{A})$, any $\gamma > 0$ and any $\varepsilon \geq 0$, we have that any distribution p that certifies $\mathsf{doec}_\gamma(\mathcal{G}, \Lambda) \leq V$ also certifies that $\mathsf{dec}_\gamma(\mathcal{G}, \Lambda) \leq V + \frac{1}{\gamma} + \gamma\varepsilon$. As a consequence, $\mathsf{dec}_\gamma(\mathcal{G}, \Lambda) \leq \mathsf{doec}_{\gamma,\varepsilon}(\mathcal{G}, \Lambda) + \frac{1}{\gamma} + \gamma\varepsilon$.*

Combining Theorem 9 with Theorem 7, we obtain that $\mathsf{dec}_\gamma(\mathcal{G}, \Lambda) \lesssim \inf_{\varepsilon > 0} \left(\frac{\mathsf{SEC}_\varepsilon(\mathcal{G}, \Lambda)}{\gamma} + \frac{1}{\gamma} + \gamma\varepsilon \right)$. A similar result was obtained in (Foster et al., 2021a, Theorem 6.1) using a nonconstructive proof, via minimax theorem and bounding the dual (Bayesian) DEC. Our proof can be interpreted as constructive, as we use Algorithm 2 to construct a distribution to certify this upper bound.

Furthermore, when $\mathsf{SEC}_\varepsilon(\mathcal{G}, \Lambda) \leq D \cdot \mathrm{polylog}(\frac{1}{\varepsilon})$ with $D \geq 1$, Theorem 9 implies that $\mathsf{dec}_\gamma(\mathcal{G}, \Lambda) \leq \frac{D}{\gamma}\mathrm{polylog}(\frac{1}{\gamma})$, which enables $\widetilde{O}\big(\sqrt{DT \ln |\mathcal{F}|}\big)$ regret for E2D (Foster et al., 2021b). In Appendix F.2, we present SQUARECB.F, an online oracle-efficient contextual bandit algorithm in light of this observation. SQUARECB.F employs Algorithm 2 as a subroutine, and we speculate that it may sometimes enjoy better computational cost than the original E2D algorithm.

6. Conclusion

We design a unified algorithm, OE2D for contextual bandits with offline regression oracles. Our key algorithmic innovation is to use an exploitative F-design that generalizes prior works (Simchi-Levi and Xu, 2022; Xu and Zeevi, 2020) to achieve exploration-exploitation tradeoff. Central to OE2D's performance guarantee is a new concept, Decision-Offline Estimation Coefficient (DOEC), which we show to be small in many examples. Our results not only recover existing works but also provide new guarantees, while at the same time demonstrating robustness to environmental changes. For future work, we are interested in investigating: Does the lower bound of DOEC also imply information-theoretic lower bounds for online contextual bandit learning? Does generalizations of DOEC exist that enable other guarantees, e.g., efficient first-order contextual bandits with offline regression oracles (Foster and Krishnamurthy, 2021)? Are there finer structural characterizations of DOEC, e.g., in terms of the value function star number (Foster et al., 2020b)? Can we extend the OE2D principle to other applications such as partial monitoring, reinforcement learning, and RLHF (Wang et al., 2023; Li et al., 2025)?

Acknowledgments. We thank the anonymous COLT reviewers for their constructive feedback. We thank Advait Khopade for help with experiments in a companion project. CZ would like to thank Kyoungseok Jang and Kwang-Sung Jun for helpful discussions of the proof of Simchi-Levi and Xu (2022), and the coauthors of Krishnamurthy et al. (2020) and Alekh Agarwal for helpful discussions of the optimization and statistical analysis of Mini-Monster (Agarwal et al., 2014). We also thank Chen-Yu Wei for bringing to our attention the per-context FTRL analysis of Neu and Olkhovskaya (2020) for adversarial linear contextual bandits. We would also like to thank Haipeng Luo for making his lecture notes on Mini-Monster available (Luo, 2017). This work is supported by National Science Foundation grant IIS-2440266 (CAREER).

References

Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. *Advances in neural information processing systems*, 24, 2011.

Naoki Abe and Philip M Long. Associative reinforcement learning using linear probabilistic concepts. In *ICML*, pages 3–11, 1999.

Alekh Agarwal, Miroslav Dud\'\ik, Satyen Kale, John Langford, and Robert Schapire. Contextual bandit learning with predictable rewards. In *Artificial Intelligence and Statistics*, pages 19–26, 2012.

Alekh Agarwal, Daniel Hsu, Satyen Kale, John Langford, Lihong Li, and Robert Schapire. Taming the monster: A fast and simple algorithm for contextual bandits. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 1638–1646, 2014.

Alekh Agarwal, Haipeng Luo, Behnam Neyshabur, and Robert E Schapire. Corralling a Band of Bandit Algorithms. In *Proceedings of the Conference on Learning Theory (COLT)*, volume 65, pages 12–38, 2017.

Alekh Agarwal, Jian Qian, Alexander Rakhlin, and Tong Zhang. The non-linear f-design and applications to interactive learning. In *Forty-first International Conference on Machine Learning*, 2024.

Kareem Amin, Michael Kearns, and Umar Syed. Bandits, query learning, and the haystack dimension. In *Proceedings of the 24th Annual Conference on Learning Theory*, pages 87–106. JMLR Workshop and Conference Proceedings, 2011.

Peter Auer, Nicolò Cesa-Bianchi, Yoav Freund, and Robert E Schapire. The Nonstochastic Multiarmed Bandit Problem. *SIAM J. Comput.*, 32(1):48–77, jan 2003. ISSN 0097-5397. doi: 10.1137/S0097539701398375.

Alberto Bietti, Alekh Agarwal, and John Langford. A Contextual Bandit Bake-off. *arXiv preprint arXiv:1802.04064*, 2018.

Avrim Blum, Adam Kalai, and John Langford. Beating the hold-out: Bounds for k-fold and progressive cross-validation. In *Proceedings of the twelfth annual conference on Computational learning theory*, pages 203–208, 1999.

Nataly Brukhim, Aldo Pacchiano, Miroslav Dudik, and Robert Schapire. On the hardness of bandit learning. *arXiv preprint arXiv:2506.14746*, 2025.

Nicolo Cesa-Bianchi and Gabor Lugosi. *Prediction, Learning, and Games*. Cambridge University Press, 2006. ISBN 0521841089.

Wei Chu, Lihong Li, Lev Reyzin, and Robert Schapire. Contextual bandits with linear payoff functions. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 208–214. JMLR Workshop and Conference Proceedings, 2011.

Varsha Dani, Thomas P Hayes, and Sham M Kakade. Stochastic Linear Optimization under Bandit Feedback. In *Proceedings of the Conference on Learning Theory (COLT)*, pages 355–366, 2008.

Mert Demirer, Vasilis Syrgkanis, Greg Lewis, and Victor Chernozhukov. Semi-parametric efficient policy learning with continuous actions. *arXiv preprint arXiv:1905.10116*, 2019.

Miroslav Dudik, Daniel Hsu, Satyen Kale, Nikos Karampatziakis, John Langford, Lev Reyzin, and Tong Zhang. Efficient optimal learning for contextual bandits. *arXiv preprint arXiv:1106.2369*, 2011.

Vitaly Feldman, Parikshit Gopalan, Subhash Khot, and Ashok Kumar Ponnuswami. New results for learning noisy parities and halfspaces. In *2006 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS'06)*, pages 563–574. IEEE, 2006.

Sarah Filippi, Olivier Cappe, Aurélien Garivier, and Csaba Szepesvári. Parametric Bandits: The Generalized Linear Case. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 586–594, 2010.

Alexander Rakhlin Dylan J Foster and A Rakhlin. Statistical reinforcement learning and decision making: Course notes. *MIT Lecture notes for course*, 9:S915, 2022.

Dylan J Foster and Akshay Krishnamurthy. Efficient first-order contextual bandits: Prediction, allocation, and triangular discrimination. *Advances in Neural Information Processing Systems*, 34: 18907–18919, 2021.

Dylan J Foster and Alexander Rakhlin. Beyond UCB: Optimal and efficient contextual bandits with regression oracles. *Proceedings of the International Conference on Machine Learning (ICML)*, 2020.

Dylan J Foster, Claudio Gentile, Mehryar Mohri, and Julian Zimmert. Adapting to misspecification in contextual bandits. *Advances in Neural Information Processing Systems*, 33:11478–11489, 2020a.

Dylan J Foster, Alexander Rakhlin, David Simchi-Levi, and Yunzong Xu. Instance-dependent complexity of contextual bandits and reinforcement learning: A disagreement-based perspective. *arXiv preprint arXiv:2010.03104*, 2020b.

Dylan J Foster, Sham M Kakade, Jian Qian, and Alexander Rakhlin. The statistical complexity of interactive decision making. *arXiv preprint arXiv:2112.13487*, 2021a.

Dylan J Foster, Sham M Kakade, Jian Qian, and Alexander Rakhlin. The Statistical Complexity of Interactive Decision Making. *CoRR*, abs/2112.1, 2021b.

Dylan J Foster, Yanjun Han, Jian Qian, and Alexander Rakhlin. Online estimation via offline estimation: An information-theoretic framework. *Advances in Neural Information Processing Systems*, 37:42840–42898, 2024.

Venkatesan Guruswami and Prasad Raghavendra. Hardness of learning halfspaces with noise. *SIAM Journal on Computing*, 39(2):742–765, 2009.

Chi Jin, Qinghua Liu, and Sobhan Miryoosefi. Bellman eluder dimension: New rich classes of rl problems, and sample-efficient algorithms. *Advances in neural information processing systems*, 34:13406–13418, 2021.

Kwang-Sung Jun and Chicheng Zhang. Crush optimism with pessimism: Structured bandits beyond asymptotic optimality. *Advances in Neural Information Processing Systems*, 33:6366–6376, 2020a.

Kwang-Sung Jun and Chicheng Zhang. Crush Optimism with Pessimism: Structured Bandits Beyond Asymptotic Optimality. *ICML Workshop on Theoretical Foundations of Reinforcement Learning (arXiv:2006.08754)*, 2020b.

Sayash Kapoor, Kumar Kshitij Patel, and Purushottam Kar. Corruption-tolerant bandit learning. *Machine Learning*, 108(4):687–715, 2019.

Jack Kiefer and Jacob Wolfowitz. The equivalence of two extremum problems. *Canadian Journal of Mathematics*, 12:363–366, 1960.

Akshay Krishnamurthy, Alekh Agarwal, and Miro Dudik. Contextual semibandits via supervised learning oracles. *Advances In Neural Information Processing Systems*, 29, 2016.

Akshay Krishnamurthy, John Langford, Aleksandrs Slivkins, and Chicheng Zhang. Contextual bandits with continuous actions: Smoothing, zooming, and adapting. *Journal of Machine Learning Research*, 21(137):1–45, 2020.

Sanath Kumar Krishnamurthy, Vitor Hadad, and Susan Athey. Adapting to misspecification in contextual bandits with offline regression oracles. In *International Conference on Machine Learning*, pages 5805–5814. PMLR, 2021.

John Langford and Tong Zhang. The epoch-greedy algorithm for contextual multi-armed bandits. *Advances in neural information processing systems*, 20(1):96–1, 2007.

Tor Lattimore and Rémi Munos. Bounded regret for finite-armed structured bandits. *Advances in neural information processing systems*, 27, 2014.

Tor Lattimore and Csaba Szepesvári. *Bandit Algorithms*. Cambridge University Press, 2020. URL https://tor-lattimore.com/downloads/book/book.pdf.

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

Tor Lattimore, Csaba Szepesvari, and Gellert Weisz. Learning with good feature representations in bandits and in rl with a generative model. In *International conference on machine learning*, pages 5662–5670. PMLR, 2020.

Orin Levy, Alon Cohen, Asaf Cassel, and Yishay Mansour. Efficient rate optimal regret for adversarial contextual mdps using online function approximation. In *International Conference on Machine Learning*, pages 19287–19314. PMLR, 2023.

Orin Levy, Liad Erez, Alon Cohen, and Yishay Mansour. Regret bounds for adversarial contextual bandits with general function approximation and delayed feedback. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

Gene Li, Pritish Kamath, Dylan J Foster, and Nati Srebro. Understanding the eluder dimension. *Advances in Neural Information Processing Systems*, 35:23737–23750, 2022.

Lihong Li, Wei Chu, John Langford, and Robert E Schapire. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th international conference on World wide web*, pages 661–670, 2010.

Lihong Li, Yu Lu, and Dengyong Zhou. Provably Optimal Algorithms for Generalized Linear Contextual Bandits. In *Proceedings of the International Conference on Machine Learning (ICML)*, volume 70, pages 2071–2080, 2017.

Long-Fei Li, Yu-Yang Qian, Peng Zhao, and Zhi-Hua Zhou. Provably efficient rlhf pipeline: A unified view from contextual bandits. *ArXiv preprint*, 2502, 2025.

Haipeng Luo. Csci 699 fall 2017 lecture 21. 2017. URL https://haipeng-luo.net/courses/CSCI699/lecture21.pdf.

Maryam Majzoubi, Chicheng Zhang, Rajan Chari, Akshay Krishnamurthy, John Langford, and Aleksandrs Slivkins. Efficient contextual bandits with continuous actions. *Advances in Neural Information Processing Systems*, 33:349–360, 2020.

Gergely Neu and Julia Olkhovskaya. Efficient and robust algorithms for adversarial linear contextual bandits. In *Conference on Learning Theory*, pages 3049–3068. PMLR, 2020.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35: 27730–27744, 2022.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

A Rakhlin, K Sridharan, and AB Tsybakov. Empirical entropy, minimax regret and minimax risk. *Bernoulli: a journal of mathematical statistics and probability*, 23(2):789–824, 2017.

Daniel Russo and Benjamin Van Roy. Eluder dimension and the sample complexity of optimistic exploration. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 2256–2264, 2013.

Aadirupa Saha and Robert E Schapire. Efficient and near-optimal algorithm for contextual dueling bandits with offline regression oracles. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

David Simchi-Levi and Yunzong Xu. Bypassing the monster: A faster and simpler optimal algorithm for contextual bandits under realizability. *Mathematics of Operations Research*, 47(3):1904–1931, 2022.

Yuda Song, Yifei Zhou, Ayush Sekhari, J Andrew Bagnell, Akshay Krishnamurthy, and Wen Sun. Hybrid rl: Using both offline and online data can make rl efficient. *arXiv preprint arXiv:2210.06718*, 2022.

Ambuj Tewari and Susan A Murphy. From ads to interventions: Contextual bandits in mobile health. In *Mobile health: sensors, analytic methods, and applications*, pages 495–517. Springer, 2017.

Andrea Tirinzoni, Alessandro Lazaric, and Marcello Restelli. A Novel Confidence-Based Algorithm for Structured Bandits. In *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2020.

Tim van Erven, Jack Mayo, Julia Olkhovskaya, and Chen-Yu Wei. An improved algorithm for adversarial linear contextual bandits via reduction. *Advances in Neural Information Processing Systems*, 38:135838–135863, 2026.

Gaël Varoquaux, Lars Buitinck, Gilles Louppe, Olivier Grisel, Fabian Pedregosa, and Andreas Mueller. Scikit-learn: Machine learning without learning the machinery. *GetMobile: Mobile Computing and Communications*, 19(1):29–33, 2015.

Yuanhao Wang, Qinghua Liu, and Chi Jin. Is rlhf more difficult than standard rl? *arXiv preprint arXiv:2306.14111*, 2023.

Tengyang Xie, Dylan J Foster, Yu Bai, Nan Jiang, and Sham M Kakade. The role of coverage in online reinforcement learning. *arXiv preprint arXiv:2210.04157*, 2022.

Yunbei Xu and Assaf Zeevi. Upper counterfactual confidence bounds: a new optimism principle for contextual bandits. *arXiv preprint arXiv:2007.07876*, 2020.

Yuhong Yang and Andrew Barron. Information-theoretic determination of minimax rates of convergence. *Annals of Statistics*, pages 1564–1599, 1999.

Tong Zhang. Feel-good thompson sampling for contextual bandits and reinforcement learning. *SIAM Journal on Mathematics of Data Science*, 4(2):834–857, 2022.

Yinglun Zhu and Paul Mineiro. Contextual bandits with smooth regret: Efficient learning in continuous action spaces. In *International Conference on Machine Learning*, pages 27574–27590. PMLR, 2022.

Yinglun Zhu and Robert Nowak. Efficient active learning with abstention. *Advances in Neural Information Processing Systems*, 35:35379–35391, 2022.

Yinglun Zhu, Dylan J Foster, John Langford, and Paul Mineiro. Contextual bandits with large action spaces: Made practical. In *International Conference on Machine Learning*, pages 27428–27453. PMLR, 2022.

Contents

Appendix A. Additional Related Work

Contextual Bandits Contextual bandits is one of the most fundamental models for sequential decision making with side information (Langford and Zhang, 2007; Agarwal et al., 2014; Li et al., 2010). Our work focuses on the stochastic setting, where the expected cost of taking action a given context x is governed by some fixed function f^*. The adversarial contextual bandit problem has also been extensively studied (e.g., Auer et al., 2003; Neu and Olkhovskaya, 2020; van Erven et al., 2026); our analysis has some similarities with Neu and Olkhovskaya (2020), in that we also first establish a per-context regret bound and conclude the final regret bound via averaging. For stochastic contextual bandits, most of the early works study a global structure on the reward function class, such as linear (Dani et al., 2008; Abbasi-Yadkori et al., 2011; Chu et al., 2011), generalized linear models (Filippi et al., 2010; Li et al., 2017), and general nonlinear (Russo and Van Roy, 2013).

There are two main lines of works for efficient stochastic contextual bandit algorithms with general function approximation: reduction to cost-sensitive classification (Langford and Zhang, 2007; Dudik et al., 2011; Agarwal et al., 2014; Krishnamurthy et al., 2016, 2020; Majzoubi et al., 2020) and reduction to regression (Abe and Long, 1999; Agarwal et al., 2012; Russo and Van Roy, 2013; Foster and Rakhlin, 2020; Simchi-Levi and Xu, 2022; Xu and Zeevi, 2020). Langford and Zhang (2007) first reduces the contextual bandit problem to cost-sensitive classification among a given policy class and achieves $O\big(T^{2/3}\big)$ regret. Dudik et al. (2011) improves the regret to $O\big(\sqrt{T}\big)$ with $O\big(T^5\big)$ oracle calls per round. Agarwal et al. (2014) further reduces the total oracle calls to $\widetilde{O}\big(\sqrt{KT}\big)$. The regression-based approach directly learns a reward model and uses it to guide exploration, which bypasses the computational hardness of agnostic classification in policy search-based approaches (Guruswami and Raghavendra, 2009; Feldman et al., 2006).

Contextual Bandits with Online Regression Oracles Online regression oracles have been widely used in contextual bandit algorithms since Foster and Rakhlin (2020), who proposes the SQUARECB algorithm to reduce the contextual bandit problem to online regression by using inverse gap weighting (IGW) (Abe and Long, 1999) to guide exploration and achieves $O\big(\sqrt{KT\mathrm{Reg}_{\mathrm{on}}}\big)$ regret, where $\mathrm{Reg}_{\mathrm{on}}$ is the online regression oracle's estimation error. More recent works further improve the SQUARECB algorithm in various aspects: Foster et al. (2020a) analyzes the linear reward setting with misspecification error; Foster and Krishnamurthy (2021) extends to first-order regret guarantees via online log-loss regression; Zhu and Mineiro (2022); Zhu et al. (2022) extend the SQUARECB algorithm to large action space settings with smooth regret and per-context linear reward structure,

Algorithm	Regression oracle	Total # oracle calls	Assumptions
E2D (Foster et al., 2021a)	Online	T	General function classes
Falcon (Simchi-Levi and Xu, 2022)	Offline	$\log(T)$ *	Discrete action space
Linear Falcon (Xu and Zeevi, 2020, Sec. 4)	Offline	$\log(T)$ *	Per-context linear reward
UCCB (Xu and Zeevi, 2020)	Offline	T	General function classes
E2D.Off (Foster et al., 2024)	Offline	T	General function classes
OE2D (this paper)	Offline	$\log(T)$ *	General function classes

Table 1: Performance comparison of $\sqrt{T}$-regret contextual bandit algorithms with regression oracles. *: If T is known, the number of oracle calls can be reduced to $O(\log\log(T))$ with a specific schedule (details can be found in Appendix B.6.2).

respectively; Levy et al. (2025) extends to the delayed feedback setting. Foster et al. (2021a) gives a unified framework, E2D, that reduces online learning to online estimation in various interactive learning problems including contextual bandits.

Contextual Bandits with Offline Regression Oracles Offline regression oracles are more practical in real-world applications (for example, Empirical Risk Minimization and its variants including regularized least squares or logistic regression). Finding effective exploration strategies is a key challenge in designing contextual bandit algorithms with offline regression oracles. IGW-based exploration is sufficient to construct an action distribution that satisfies both the low-regret (LR) and good-coverage (GC) conditions in the discrete action space setting (Simchi-Levi and Xu, 2022) and linear reward setting (Xu and Zeevi, 2020, Section 4). Xu and Zeevi (2020) analyzes the contextual bandit problem under a general reward function class and proposes the upper counterfactual confidence bound (UCCB) algorithm using counterfactual action divergence to unify exploration strategies in various function classes. However, their algorithm requires $O(T)$ calls to the offline regression oracle. Saha and Schapire (2025) solves a linear contextual dueling bandit problem where they merge the LR and GC conditions into one constraint to design an efficient algorithm. This inspires our definition of DOEC, where any exploration distribution certifying an upper bound on DOEC satisfies LR and GC simultaneously. Foster et al. (2024) provides E2D.Off that combines the E2D reduction and a new reduction from online estimation to offline estimation; their number of calls to the offline oracle is $O(T)$, and when instantiated to contextual bandits, it has a suboptimal dependence of $\sqrt{\log|\mathcal{F}|}$ in the regret guarantee. We refer the reader to Table 1 for a comparison of the number of oracle calls required for existing oracle-efficient algorithms.

Experimental Design Experimental design is a classical topic in statistics. The classical result of (Kiefer and Wolfowitz, 1960) shows that for linear regression, G-optimal design, minimizing the maximum predictive variance over the space of covariates, is equivalent to D-optimal design, i.e., designing the data distribution such that the determinant of its covariance matrix is maximized. Lattimore and Szepesvári (2020, Ch. 22) first uses G-optimal design in the fixed action set linear bandit setting. Beyond the linear regime, Agarwal et al. (2024) propose nonlinear F-design, which generalizes G-optimal design to general function classes to guide exploration in many interactive

learning settings. These works inspire our exploitative F-design for exploration with general function approximation.

Appendix B. Proofs from Section 3 Part 1: Regret Analysis for OE2D

In this section, we provide the regret analysis for OE2D (Algorithm 1). Before diving into the proof details, we first summarize the main notations used in this section.

B.1. Basic Notations

- p, q: distributions over the action space $\mathcal{A}$.
- $m(t)$: the epoch index that time step t belongs to.
- M: total number of epochs up to time T, i.e., $M = m(T)$.
- p_t: the distribution constructed at time step t based on the estimated reward function $\hat{f}_{m(t)}$ and context x_t.
- π_m: policy executed in epoch m, which maps from context space $\mathcal{X}$ to distributions Λ.
- τ_m: the terminal time step of epoch m.
- $\mathcal{R}(p \mid x) \coloneqq \mathbb{E}_{a\sim p}[f^*(x,a)]$: ground-truth expected reward of distribution p under context x.
- $\widehat{\mathcal{R}}_m(p \mid x) \coloneqq \mathbb{E}_{a\sim p}[\hat{f}_m(x,a)]$: estimated expected reward of distribution p under context x in epoch m based on the estimated reward function $\hat{f}_m$.
- $\mathrm{Reg}(p \mid x) \coloneqq \max_{\lambda'\in\Lambda} \mathcal{R}(\lambda' \mid x) - \mathcal{R}(p \mid x)$, the regret of distribution p under context x.
- $\widehat{\mathrm{Reg}}_m(p \mid x) \coloneqq \max_{\lambda'\in\Lambda} \widehat{\mathcal{R}}_m(\lambda' \mid x) - \widehat{\mathcal{R}}_m(p \mid x)$, the estimated regret of distribution p under context x in epoch m based on the estimated reward function $\hat{f}_m$. To avoid clutter, with a slight abuse of notation, we will use notation $\widehat{\mathrm{Reg}}_m(\pi \mid x)$ to denote $\max_{\lambda'\in\Lambda} \widehat{\mathcal{R}}_m(\lambda' \mid x) - \widehat{\mathcal{R}}_m(\pi(\cdot \mid x) \mid x)$, the estimated regret of policy π under context x in epoch m based on the estimated reward function $\hat{f}_m$.
- $\mathrm{SqErr}_m(f, x) \coloneqq \mathbb{E}_{a\sim\pi_m(\cdot|x)}\left[(f(x,a) - f^*(x,a))^2\right]$:the expected estimation error of f measured by policies executed in epoch m under context x.

B.2. Virtual Policy Viewpoint of OE2D

Our analysis will adopt a "virtual policy" equivalent view of Algorithm 1, that is, Algorithm 3. Specifically, Algorithm 3 computes a policy π_m at the beginning of epoch m such that for every $x \in \mathcal{X}$, the action distribution $\pi_m(\cdot|x)$ certifies a small relaxed DOEC, which further implies low regret and good coverage properties (Eqs. (LR_x) and(GC_x)). We clarify that Algorithm 3 is not intended to be practical since it needs to compute a separate exploration distribution for each context; we introduce it only for the sake of analysis – specifically, we will prove online performance guarantees of policy sequence $\{\pi_{m(t)}\}_{t=1}^T$ on every context x.

Algorithm 3 OE2D: conceptual version

Input: learning parameter $\{\gamma_m, \varepsilon\}_{m=1}^{M}$, epoch schedule $0 = \tau_0 < \tau_1 < \cdots < \tau_M$, benchmark distributions Λ, reward function class $\mathcal{F}$, offline regression oracle $\mathcal{O}_{\text{off}}$, relaxed coverage $\overline{\mathsf{Coverage}}$.

for *epoch* $m = 1$ **to** M **do**
  if $m = 1$ **then**
    Compute $\pi_1(\cdot \mid x) \coloneqq \operatorname{argmin}_{p \in \mathrm{co}(\Lambda)} \max_{\lambda \in \Lambda} \overline{\mathsf{Coverage}}(p, \lambda; \mathcal{F}_x), \forall x \in \mathcal{X}$.
  else
    Compute $\hat{f}_m \leftarrow \mathcal{O}_{\text{off}}(\mathcal{F})(\{(x_i, a_i, r_i)\}_{i=\tau_{m-2}+1}^{\tau_{m-1}})$.
    Define policy π_m as follows:

$$\pi_m(\cdot|x) \coloneqq \operatorname*{argmin}_{p \in \mathrm{co}(\Lambda)} \max_{\lambda \in \Lambda} \left(\widehat{\mathcal{R}}_m(\lambda \mid x) - \widehat{\mathcal{R}}_m(p \mid x) + \frac{\overline{\mathsf{Coverage}}(p, \lambda; \mathcal{F}_x)}{\gamma_m} \right), \forall x \in \mathcal{X}. \tag{7}$$

  for *round* $t = \tau_m + 1$ **to** τ_{m+1} **do**
    Observe context $x_t \in \mathcal{X}$, sample action $a_t \sim \pi_m(\cdot \mid x_t)$, and observe reward r_t.

Here we prove a generalization of Lemma 3, namely Lemma 10, that shows the virtual policies π_m's satisfies low regret and good coverage properties simultaneously, which will serve as the basis of our subsequent regret analysis.

Lemma 10

For context x, distribution $\pi_m(\cdot|x)$ satisfies the following two properties simultaneously for $m \geq 2$:

- *Low Regret:*

$$\widehat{\mathrm{Reg}}_m(\pi_m \mid x) \leq \overline{\mathsf{doec}}_{\gamma_m, \varepsilon_m}(\mathcal{F}_x, \Lambda) \tag{LR$_x$}$$

- *Good Coverage:*

$$\mathsf{Coverage}_{\varepsilon_m}(\pi_m, \lambda; \mathcal{F}_x) \leq \gamma_m \overline{\mathsf{doec}}_{\gamma_m, \varepsilon_m}(\mathcal{F}_x, \Lambda) + \gamma_m \widehat{\mathrm{Reg}}_m(\lambda \mid x), \quad \forall \lambda \in \Lambda, \tag{GC$_x$}$$

And for $m = 1$, we have

$$\mathsf{Coverage}_{\varepsilon_1}(\pi_1, \lambda; \mathcal{F}_x) \leq \gamma_1 \overline{\mathsf{doec}}_{\gamma_1, \varepsilon_1}(\mathcal{F}_x, \Lambda), \quad \forall \lambda \in \Lambda$$

where we recall the convention that $\gamma_1 = 0$ and $\gamma_1 \overline{\mathsf{doec}}_{\gamma_1, \varepsilon_1}(\mathcal{F}_x, \Lambda)$ is interpreted as $\overline{\mathcal{V}}^*_{\varepsilon_1}(\mathcal{F}_x, \Lambda)$.

Proof For $m \geq 2$, since by Eq. (7), $\pi_m(\cdot|x)$ minimizes

$$\widehat{\mathrm{Reg}}_m(p \mid x) + \max_{\lambda \in \Lambda} \left(\tfrac{1}{\gamma_m} \overline{\mathsf{Coverage}}(p, \lambda; \mathcal{F}_x) - \widehat{\mathrm{Reg}}_m(\lambda \mid x) \right),$$

whose optimal objective is $\overline{\mathsf{doec}}_{\gamma_m,\varepsilon_m}(\mathcal{F}_x,\Lambda)$. Therefore,

$$\widehat{\mathrm{Reg}}_m(\pi_m \mid x) + \max_{\lambda\in\Lambda}\left(\tfrac{1}{\gamma_m}\overline{\mathsf{Coverage}}(\pi_m,\lambda;\mathcal{F}_x) - \widehat{\mathrm{Reg}}_m(\lambda \mid x)\right) \le \overline{\mathsf{doec}}_{\gamma_m,\varepsilon_m}(\mathcal{F}_x,\Lambda),$$

where both terms on the left-hand side are nonnegative. Dropping the second term, we get

$$\widehat{\mathrm{Reg}}_m(\pi_m \mid x) \le \overline{\mathsf{doec}}_{\gamma_m,\varepsilon_m}(\mathcal{F}_x,\Lambda),$$

thus proving Eq. (LR_x). On the other hand, dropping the first term, we get

$$\max_{\lambda\in\Lambda}\left(\tfrac{1}{\gamma_m}\overline{\mathsf{Coverage}}(\pi_m,\lambda;\mathcal{F}_x) - \widehat{\mathrm{Reg}}_m(\lambda \mid x)\right) \le \overline{\mathsf{doec}}_{\gamma_m,\varepsilon_m}(\mathcal{F}_x,\Lambda);$$

since $\mathsf{Coverage}_{\varepsilon_m}(\pi_m,\lambda;\mathcal{F}_x) \le \overline{\mathsf{Coverage}}(\pi_m,\lambda;\mathcal{F}_x)$ pointwise, this proves Eq. (GC_x) with the original coverage on the left-hand side.

For $m=1$, by the definition of π_1 in Algorithm 3, we have that p_1 minimizes $\max_{\lambda\in\Lambda}\overline{\mathsf{Coverage}}(p,\lambda;\mathcal{F}_x)$, whose optimal objective is $\overline{\mathcal{V}}^*_{\varepsilon_1}(\mathcal{F}_x,\Lambda)$, which is equal to $\lim_{\gamma\to 0}\gamma\overline{\mathsf{doec}}_{\gamma,\varepsilon_1}(\mathcal{F}_x,\Lambda)$ by the convention; the above inequality again follows from $\mathsf{Coverage}_{\varepsilon_1} \le \overline{\mathsf{Coverage}}$ pointwise. ■

B.3. Favorable Event

We define an event that the offline regression oracle returns a good estimated reward function for each epoch m:

$$E_m = \left\{\mathbb{E}_{x\sim\mathcal{D}_X,a\sim\pi_{m-1}(\cdot|x)}\left[\hat{f}_m(x,a) - f^*(x,a)\right]^2 \le \mathrm{Reg}_{\mathrm{off}}(\mathcal{F},\tau_{m-1}-\tau_{m-2},\delta_m)\right\}.$$

By taking a union bound over all epochs $m\in[1,M]$, we have that with probability at least $1-\sum_{m=1}^M\delta_m \ge 1-\delta$, event $E=\cap_{m=1}^M E_m$ holds. Throughout the rest of the proof, we condition on event E happening.

B.4. Results of Per-Context Regret Analysis

Next, we are going to prove the concentration between the true reward and the estimated reward in the per context manner in each epoch m.

Lemma 11 (Regret Concentration in Coverage Per Context) *For any $\delta\in(0,1)$, reward function class $\mathcal{F}$, with probability at least $1-\delta$, for all $x\in\mathcal{X}$, $m\in[2,M]$ and any $\lambda\in\Delta(\mathcal{A})$, we have*

$$\left|\widehat{\mathcal{R}}_m(\lambda \mid x) - \mathcal{R}(\lambda \mid x)\right| \le \sqrt{\mathsf{Coverage}_{\varepsilon_{m-1}}(\pi_{m-1},\lambda;\mathcal{F}_x \mid x)\left(\mathrm{SqErr}_{m-1}(\hat{f}_m,x)+\varepsilon_{m-1}\right)}$$

Proof [Proof of Lemma 11] Starting from the squared LHS of the desired inequality, we have:

$$\begin{aligned}
&\left(\widehat{\mathcal{R}}_m(\lambda \mid x) - \mathcal{R}(\lambda \mid x)\right)^2 = \left(\mathbb{E}_{a\sim\lambda}\left[\hat{f}_m(x,a) - f^*(x,a)\right]\right)^2 \\
=&\frac{\left(\mathbb{E}_{a\sim\lambda}\left[\hat{f}_m(x,a) - f^*(x,a)\right]\right)^2}{\mathbb{E}_{a\sim\pi_{m-1}(\cdot|x)}\left[\hat{f}_m(x,a) - f^*(x,a)\right]^2 + \varepsilon_{m-1}} \cdot \left(\mathrm{SqErr}_{m-1}(\hat{f}_m,x)+\varepsilon_{m-1}\right) \\
\le&\mathsf{Coverage}_{\varepsilon_{m-1}}(\pi_{m-1},\lambda;\mathcal{F}_x \mid x)\cdot\left(\mathrm{SqErr}_{m-1}(\hat{f}_m,x)+\varepsilon_{m-1}\right) \quad \text{(Definition of } \mathsf{Coverage}(\cdot)\text{)}
\end{aligned}$$

By taking a square root on both sides of the above inequality, we get the desired inequality. ■

Based on above lemmas, we are ready to prove a central result that control the regret and estimated regret for every x in each epoch $m \in [2, M]$ with each other by a threshold $G_m(x)$. Before we state the lemma, we first define $G_m(x)$:

$$G_m(x) := \frac{1}{\gamma_m} \sum_{s=2}^{m} \left(2\gamma_{s-1} \overline{\mathsf{doec}}_{\gamma_{s-1},\varepsilon_{s-1}}(\mathcal{F}_x, \Lambda) + 9\gamma_s^2 \left(\mathrm{SqErr}_{s-1}(\hat{f}_s, x) + \varepsilon_{s-1} \right) \right). \tag{8}$$

Here we recall the convention that $\gamma_1 = 0$ and $\gamma_1 \overline{\mathsf{doec}}_{\gamma_1,\varepsilon_1} = \overline{\mathcal{V}}^*_{\varepsilon_1}(\mathcal{F}_x, \Lambda)$.

Lemma 12 (Regret Concentration in All Epochs per Context) *For any $\delta \in (0,1)$, with probability at least $1-\delta$, for all $x \in \mathcal{X}$, $m \in [2, M]$ and $\lambda \in \Lambda$, we have*

$$\mathrm{Reg}(\lambda \mid x) \le 2\widehat{\mathrm{Reg}}_m(\lambda \mid x) + G_m(x) \tag{9}$$

$$\widehat{\mathrm{Reg}}_m(\lambda \mid x) \le 2\mathrm{Reg}(\lambda \mid x) + G_m(x) \tag{10}$$

and consequently, the above two inequalities also hold for all $\lambda \in \mathrm{co}(\Lambda)$.

Proof To avoid clutter, we use

- $\mathrm{SqErr}_{m-1} \leftarrow \mathrm{SqErr}_{m-1}(\hat{f}_m, x)$
- $\mathsf{Cov}_{m-1}(\cdot) \leftarrow \mathsf{Coverage}_{\varepsilon_{m-1}}(\pi_{m-1}(\cdot|x), \cdot; \mathcal{F}_x)$
- $\overline{\mathsf{doec}}_{m-1} \leftarrow \overline{\mathsf{doec}}_{\gamma_{m-1},\varepsilon_{m-1}}(\mathcal{F}_x, \Lambda)$
- $\widehat{\lambda}_m := \mathrm{argmax}_{\lambda \in \Lambda} \widehat{\mathrm{Reg}}_m(\lambda \mid x)$
- $\lambda^* := \mathrm{argmax}_{\lambda \in \Lambda} \mathrm{Reg}(\lambda \mid x)$

for notational simplicity in the following proof.

Denote by

$$H_m(x) := \gamma_m G_m(x) = \sum_{s=2}^{m} \left(2\gamma_{s-1} \overline{\mathsf{doec}}_{s-1} + 9\gamma_s^2 (\mathrm{SqErr}_{s-1} + \varepsilon_{s-1}) \right).$$

We will subsequently use the property that

$$H_m(x) - H_{m-1}(x) \ge 2\gamma_{m-1} \overline{\mathsf{doec}}_{m-1} + 9\gamma_m^2 (\mathrm{SqErr}_{m-1} + \varepsilon_{m-1}) \tag{11}$$

Base case: For $m = 2$, Lemma 11 implies that for any $\lambda \in \Lambda$,

$$|\widehat{\mathcal{R}}_2(\lambda \mid x) - \mathcal{R}(\lambda \mid x)| \le \sqrt{\mathsf{Cov}_1(\lambda)(\mathrm{SqErr}_1 + \varepsilon_1)} \le \sqrt{(\mathrm{SqErr}_1 + \varepsilon_1) \cdot \gamma_1 \overline{\mathsf{doec}}_{\gamma_1,\varepsilon_1}},$$

where the second inequality is because Lemma 10 guarantees that for all $\lambda \in \Lambda$, $\mathsf{Coverage}_{\varepsilon_1}(\pi_1(\cdot|x), \lambda; \mathcal{F}_x \mid x) \le \overline{\mathcal{V}}^*_{\varepsilon_1}(\mathcal{F}_x, \Lambda) = \gamma_1 \overline{\mathsf{doec}}_{\gamma_1,\varepsilon_1}(\mathcal{F}_x, \Lambda)$.

Thus, Lemma 13 implies that

$$\begin{aligned}\left|\mathrm{Reg}(\lambda \mid x) - \widehat{\mathrm{Reg}}_2(\lambda \mid x)\right| \leq & 2\sqrt{(\mathrm{SqErr}_1 + \varepsilon_1) \cdot \gamma_1 \overline{\mathsf{doec}}_{\gamma_1,\varepsilon_1}(\mathcal{F}_x, \Lambda)} \\ \leq & \frac{\gamma_2^2(\mathrm{SqErr}_1 + \varepsilon_1) + \gamma_1 \overline{\mathsf{doec}}_{\gamma_1,\varepsilon_1}(\mathcal{F}_x, \Lambda)}{\gamma_2} \\ \leq & G_2(x),\end{aligned}$$

where the second inequality is from AM-GM inequality, and the third inequality is by the definition of $G_m(x)$. This establishes Eqs. (9) and (10) for $m = 2$.

Inductive step: Assume that Eqs. (9) and (10) hold for epoch $m-1$. We will show that the same inequalities hold for epoch m.

Proof of Eq. (9): We will start by upper bounding $\mathrm{Reg}(\lambda \mid x) - \widehat{\mathrm{Reg}}_m(\lambda \mid x)$:

$$\begin{aligned}&\mathrm{Reg}(\lambda \mid x) - \widehat{\mathrm{Reg}}_m(\lambda \mid x) \\ \leq & \left|\mathcal{R}(\lambda^* \mid x) - \widehat{\mathcal{R}}_m(\lambda^* \mid x)\right| + \left|\mathcal{R}(\lambda \mid x) - \widehat{\mathcal{R}}_m(\lambda \mid x)\right| && \text{(Lemma 13)} \\ \leq & \frac{1}{\gamma_m}\left(\sqrt{\mathsf{Cov}_{m-1}(\lambda^*)\gamma_m^2(\mathrm{SqErr}_{m-1} + \varepsilon_{m-1})} + \sqrt{\mathsf{Cov}_{m-1}(\lambda)\gamma_m^2(\mathrm{SqErr}_{m-1} + \varepsilon_{m-1})}\right) \\ & && \text{(Applying Lemma 11 and algebra)} \\ \leq & \frac{1}{\gamma_m}\left(3\gamma_m^2(\mathrm{SqErr}_{m-1} + \varepsilon_{m-1}) + \frac{1}{6}\mathsf{Cov}_{m-1}(\lambda^*) + \frac{1}{6}\mathsf{Cov}_{m-1}(\lambda)\right) && (12)\end{aligned}$$

where the last inequality is by AM-GM inequality.

Continuing Eq. (12), we now upper bound $\mathsf{Cov}_{m-1}(\lambda^*)$ and $\mathsf{Cov}_{m-1}(\lambda)$ by the good coverage property of π_{m-1} as well as inductive hypothesis:

$$\begin{aligned}(12) \leq & \frac{1}{\gamma_m}\left(\begin{array}{l}3\gamma_m^2(\mathrm{SqErr}_{m-1} + \varepsilon_{m-1}) + \frac{\gamma_{m-1}}{3}\overline{\mathsf{doec}}_{m-1} \\ \quad + \frac{\gamma_{m-1}}{6}\widehat{\mathrm{Reg}}_{m-1}(\lambda^* \mid x) + \frac{\gamma_{m-1}}{6}\widehat{\mathrm{Reg}}_{m-1}(\lambda \mid x)\end{array}\right) \\ & \text{(Good Coverage property for } \pi_{m-1}\text{, Lemma 10)} \\ \leq & \frac{1}{\gamma_m}\left(\begin{array}{l}3\gamma_m^2(\mathrm{SqErr}_{m-1} + \varepsilon_{m-1}) + \frac{\gamma_{m-1}}{3}\overline{\mathsf{doec}}_{m-1} \\ \quad + \frac{\gamma_{m-1}}{3}\mathrm{Reg}(\lambda^* \mid x) + \frac{\gamma_{m-1}}{3}\mathrm{Reg}(\lambda \mid x) + \frac{\gamma_{m-1}G_{m-1}(x)}{3}\end{array}\right) \\ & \text{(Inductive hypothesis and algebra)} \\ \leq & \frac{1}{\gamma_m}\left(3\gamma_m^2(\mathrm{SqErr}_{m-1} + \varepsilon_{m-1}) + \frac{\gamma_{m-1}}{3}\overline{\mathsf{doec}}_{m-1} + \frac{H_{m-1}(x)}{3}\right) \\ & \quad + \frac{1}{3}\mathrm{Reg}(\lambda^* \mid x) + \frac{1}{3}\mathrm{Reg}(\lambda \mid x) \\ & \text{($\gamma_m \geq \gamma_{m-1}$, $H_{m-1}(x) = \gamma_{m-1}G_{m-1}(x)$)} \\ \leq & \frac{H_m(x)}{3\gamma_m} + \frac{1}{3}\mathrm{Reg}(\lambda \mid x) && \text{($\mathrm{Reg}(\lambda^* \mid x) = 0$ and Eq. (11))} \\ = & \frac{1}{3}G_m(x) + \frac{1}{3}\mathrm{Reg}(\lambda \mid x) && \text{(Definition of } H_m(x) = \gamma_m G_m(x)\text{)}\end{aligned}$$

Solving the inequality for $\mathrm{Reg}(\lambda \mid x)$, we get:

$$\frac{2}{3}\mathrm{Reg}(\lambda \mid x) \le \widehat{\mathrm{Reg}}_m(\lambda \mid x) + \frac{1}{3}G_m(x) \implies \mathrm{Reg}(\lambda \mid x) \le \frac{3}{2}\widehat{\mathrm{Reg}}_m(\lambda \mid x) + \frac{1}{2}G_m(x),$$

thus establishing Eq. (9).

Proof of Eq. (10): We will start by upper bounding $\widehat{\mathrm{Reg}}_m(\lambda \mid x) - \mathrm{Reg}(\lambda \mid x)$:

$$\begin{aligned}
&\widehat{\mathrm{Reg}}_m(\lambda \mid x) - \mathrm{Reg}(\lambda \mid x) \\
\le& \left|\mathcal{R}(\hat{\lambda}_m \mid x) - \widehat{\mathcal{R}}_m(\hat{\lambda}_m \mid x)\right| + \left|\mathcal{R}(\lambda \mid x) - \widehat{\mathcal{R}}_m(\lambda \mid x)\right| && \text{(Lemma 13)} \\
\le& \frac{1}{\gamma_m}\left(\sqrt{\mathsf{Cov}_{m-1}(\hat{\lambda}_m)\gamma_m^2(\mathrm{SqErr}_{m-1} + \varepsilon_{m-1})} + \sqrt{\mathsf{Cov}_{m-1}(\lambda)\gamma_m^2(\mathrm{SqErr}_{m-1} + \varepsilon_{m-1})}\right) \\
& && \text{(Applying Lemma 11 and algebra)} \\
\le& \frac{1}{\gamma_m}\left(3\gamma_m^2(\mathrm{SqErr}_{m-1} + \varepsilon_{m-1}) + \frac{1}{6}\mathsf{Cov}_{m-1}(\hat{\lambda}_m) + \frac{1}{6}\mathsf{Cov}_{m-1}(\lambda)\right) && (13)
\end{aligned}$$

where the last inequality is by AM-GM inequality.

Continuing Eq. (13), we now upper bound $\mathsf{Cov}_{m-1}(\hat{\lambda}_m)$ and $\mathsf{Cov}_{m-1}(\lambda)$ by the good coverage property of π_{m-1} as well as inductive hypothesis:

$$\begin{aligned}
(13) \le& \frac{1}{\gamma_m}\left(\begin{array}{l} 3\gamma_m^2(\mathrm{SqErr}_{m-1} + \varepsilon_{m-1}) + \frac{\gamma_{m-1}}{3}\overline{\mathsf{doec}}_{m-1} \\ \quad + \frac{\gamma_{m-1}}{6}\widehat{\mathrm{Reg}}_{m-1}(\hat{\lambda}_m \mid x) + \frac{\gamma_{m-1}}{6}\widehat{\mathrm{Reg}}_{m-1}(\lambda \mid x)\end{array}\right) \\
& \text{(Good Coverage property for } \pi_{m-1}\text{, Lemma 10)} \\
\le& \frac{1}{\gamma_m}\left(\begin{array}{l} 3\gamma_m^2(\mathrm{SqErr}_{m-1} + \varepsilon_{m-1}) + \frac{\gamma_{m-1}}{3}\overline{\mathsf{doec}}_{m-1} \\ \quad + \frac{\gamma_{m-1}}{3}\mathrm{Reg}(\hat{\lambda}_m \mid x) + \frac{\gamma_{m-1}}{3}\mathrm{Reg}(\lambda \mid x) + \frac{\gamma_{m-1}G_{m-1}(x)}{3}\end{array}\right) \\
& \text{(Inductive hypothesis and algebra)} \\
\le& \frac{1}{\gamma_m}\left(\begin{array}{l} 3\gamma_m^2(\mathrm{SqErr}_{m-1} + \varepsilon_{m-1}) + \frac{\gamma_{m-1}}{3}\overline{\mathsf{doec}}_{m-1} \\ \quad + \frac{\gamma_{m-1}G_m(x)}{3} + \frac{\gamma_{m-1}}{3}\mathrm{Reg}(\lambda \mid x) + \frac{\gamma_{m-1}G_{m-1}(x)}{3}\end{array}\right) \\
& \text{(Eq. (9) for epoch } m\text{)} \\
\le& \frac{1}{\gamma_m}\left(3\gamma_m^2(\mathrm{SqErr}_{m-1} + \varepsilon_{m-1}) + \frac{\gamma_{m-1}}{3}\overline{\mathsf{doec}}_{m-1} + \frac{H_{m-1}(x)}{3}\right) \\
& \quad + \frac{1}{3}G_m(x) + \frac{1}{3}\mathrm{Reg}(\lambda \mid x) \\
& \text{($\gamma_m \ge \gamma_{m-1}$, } H_{m-1}(x) = \gamma_{m-1}G_{m-1}(x)\text{)} \\
\le& \frac{H_m(x)}{3\gamma_m} + \frac{1}{3}G_m(x) + \frac{1}{3}\mathrm{Reg}(\lambda \mid x) && \text{(Eq. (11))} \\
=& \frac{2}{3}G_m(x) + \frac{1}{3}\mathrm{Reg}(\lambda \mid x) && \text{(Definition of } H_m(x) = \gamma_m G_m(x) \text{ and algebra)}
\end{aligned}$$

Solving the inequality for $\widehat{\mathrm{Reg}}_m(\pi \mid x)$, we get:

$$\widehat{\mathrm{Reg}}_m(\lambda \mid x) \leq \frac{4}{3}\mathrm{Reg}(\pi \mid x) + \frac{2}{3}G_m(x),$$

thus establishing Eq. (10).

This completes the inductive step. And thus, Eqs. (9) and (10) hold for all $m = 2, \ldots, M$.

Finally, since Eqs. (9) and (10) are concerned with $\mathrm{Reg}(\lambda)$ and $\widehat{\mathrm{Reg}}_m(\lambda)$, both of which are linear in λ, they must also hold for all $\lambda \in \mathrm{co}(\Lambda)$ as well. ■

Lemma 13 *For any $m \geq 2$ and any λ in $\Delta(\mathcal{A})$,*

$$\mathrm{Reg}(\lambda \mid x) - \widehat{\mathrm{Reg}}_m(\lambda \mid x) \leq \left|\mathcal{R}(\lambda^* \mid x) - \widehat{\mathcal{R}}_m(\lambda^* \mid x)\right| + \left|\mathcal{R}(\lambda \mid x) - \widehat{\mathcal{R}}_m(\lambda \mid x)\right| \tag{14}$$

$$\widehat{\mathrm{Reg}}_m(\lambda \mid x) - \mathrm{Reg}(\lambda \mid x) \leq \left|\mathcal{R}(\hat{\lambda}_m \mid x) - \widehat{\mathcal{R}}_m(\hat{\lambda}_m \mid x)\right| + \left|\mathcal{R}(\lambda \mid x) - \widehat{\mathcal{R}}_m(\lambda \mid x)\right| \tag{15}$$

Proof We first show Eq. (14):

$$\begin{aligned}
&\mathrm{Reg}(\lambda \mid x) - \widehat{\mathrm{Reg}}_m(\lambda \mid x) \\
&= (\mathcal{R}(\lambda^* \mid x) - \mathcal{R}(\lambda \mid x)) - \left(\widehat{\mathcal{R}}_m(\widehat{\lambda}_m \mid x) - \widehat{\mathcal{R}}_m(\lambda \mid x)\right) \\
&\leq (\mathcal{R}(\lambda^* \mid x) - \mathcal{R}(\lambda \mid x)) - \left(\widehat{\mathcal{R}}_m(\lambda^* \mid x) - \widehat{\mathcal{R}}_m(\lambda \mid x)\right) \quad (\text{Since } \widehat{\mathcal{R}}_m(\widehat{\lambda}_m \mid x) \geq \widehat{\mathcal{R}}_m(\lambda^* \mid x)) \\
&\leq \left|\mathcal{R}(\lambda^* \mid x) - \widehat{\mathcal{R}}_m(\lambda^* \mid x)\right| + \left|\mathcal{R}(\lambda \mid x) - \widehat{\mathcal{R}}_m(\lambda \mid x)\right| \quad (\text{Regrouping})
\end{aligned}$$

Eq. (15) follows from a similar calculation, as we detail below:

$$\begin{aligned}
&\widehat{\mathrm{Reg}}_m(\lambda \mid x) - \mathrm{Reg}(\lambda \mid x) \\
&= \left(\widehat{\mathcal{R}}_m(\widehat{\lambda}_m \mid x) - \widehat{\mathcal{R}}_m(\lambda \mid x)\right) - (\mathcal{R}(\lambda^* \mid x) - \mathcal{R}(\lambda \mid x)) \\
&\leq \left(\widehat{\mathcal{R}}_m(\widehat{\lambda}_m \mid x) - \widehat{\mathcal{R}}_m(\lambda \mid x)\right) - \left(\mathcal{R}(\widehat{\lambda}_m \mid x) - \mathcal{R}(\lambda \mid x)\right) \quad (\text{Since } \mathcal{R}(\lambda^* \mid x) \geq \mathcal{R}(\widehat{\lambda}_m \mid x)) \\
&\leq \left|\mathcal{R}(\widehat{\lambda}_m \mid x) - \widehat{\mathcal{R}}_m(\widehat{\lambda}_m \mid x)\right| + \left|\mathcal{R}(\lambda \mid x) - \widehat{\mathcal{R}}_m(\lambda \mid x)\right| \quad (\text{Regrouping})
\end{aligned}$$

■

We have obtained concentration between the true regret and the estimated regret in a per context manner. Next, we are ready to present an important intermediate result (Lemma 14) that controls the true regret at each time step by the exploration difficulty (the relaxed DOEC) and the regression estimation loss SqErr in each epoch m for each context x.

Lemma 14 (Λ-Regret Bound per Context for OE2D in Each Epoch) *For any $\delta \in (0, 1)$, with probability at least $1 - \delta$, for all $x \in \mathcal{X}$, $m \in [2, T]$, we have that*

$$\mathrm{Reg}(\pi_m \mid x) \leq \frac{1}{\gamma_m}\left(\sum_{s=1}^{m} 3\gamma_s \overline{\mathsf{doec}}_{\gamma_s, \varepsilon_s}(\mathcal{F}_x, \Lambda) + \sum_{s=2}^{m} 9\gamma_s^2\left(\mathrm{SqErr}_{s-1}(\hat{f}_s, x) + \varepsilon_{s-1}\right)\right). \tag{16}$$

Proof [Proof of Lemma 14] Starting from Lemma 12, we have

$$\begin{aligned}\mathrm{Reg}(\pi_m \mid x) &\le 2\widehat{\mathrm{Reg}}_m(\pi_m \mid x) + G_m(x) && \text{(Applying Equation (9) in Lemma 12)}\\ &\le 2\overline{\mathsf{doec}}_{\gamma_m,\varepsilon_m}(\mathcal{F}_x,\Lambda) + G_m(x) && \text{(Applying LR of } \pi_m \text{ (Equation (LR}_x\text{)))}\end{aligned}$$

The lemma follows from the definition of $G_m(x)$. ■

The upper bound on the per-context regret in Equation (16) is a weighted sum among all epochs $s \le m$. Each epoch s contributes to the upper bound in two ways: the first term $\overline{\mathsf{doec}}_{\gamma_s,\varepsilon_s}(\mathcal{F}_x,\Lambda)$ captures the exploration difficulty for context x in epoch s; and the second term $\gamma_s^2\left(\mathrm{SqErr}_{s-1}(\hat{f}_s,x)+\varepsilon_{s-1}\right)$ quantifies the squared loss of the regression function $\hat{f}_s$ with respect to the actions taken by the policy in epoch $s-1$, weighted by the exploration parameter γ_s squared.

B.5. Concluding the Regret Analysis

We are ready to prove the main theorem for OE2D.

Lemma 15 *For any $\delta \in (0,1)$, $\Lambda \subseteq \Delta(\mathcal{A})$, with probability at least $1-\delta$, given accessibility to any offline regression oracle, the Λ-Regret of Algorithm 1 for every context x is bounded as*

$$\begin{aligned}&\sum_{t=1}^{T}\mathrm{Reg}(\pi_{m(t)} \mid x)\\ &\le O\left(\tau_1 + M\cdot\max_{m\in\{2,\ldots,M\}}\frac{\tau_m}{\gamma_m}\cdot\left(\begin{array}{l}\sum_{s=1}^{M}\gamma_s\overline{\mathsf{doec}}_{\gamma_s,\varepsilon_s}(\mathcal{F}_x,\Lambda)\\ \qquad+\sum_{s=2}^{M}\gamma_s^2\left(\mathrm{SqErr}_{s-1}(\hat{f}_s,x)+\varepsilon_{s-1}\right)\end{array}\right)\right),\end{aligned}$$

where we recall that $\mathrm{SqErr}_s(f,x) \coloneqq \mathbb{E}_{a\sim\pi_s(\cdot|x)}\left[(f(x,a)-f^*(x,a))^2\right]$ is square loss of f under the policy used in epoch s.

Proof [Proof of Lemma 15] In Lemma 14, we have shown that for any context $x \in \mathcal{X}$, the regret of the in-epoch policy π_m is bounded. We then bound the total regret across all time steps as follows:

$$\sum_{t=1}^{T} \mathrm{Reg}(\pi_{m(t)} \mid x) \le \tau_1 + \sum_{m=2}^{M} (\tau_m - \tau_{m-1}) \mathrm{Reg}(\pi_m \mid x)$$

$$\le \tau_1 + \sum_{m=2}^{M} \frac{9\tau_m}{\gamma_m} \left(\sum_{s=1}^{M} \gamma_s \overline{\mathsf{doec}}_{\gamma_s,\varepsilon_s}(\mathcal{F}_x, \Lambda) + \sum_{s=2}^{M} \gamma_s^2 \left(\mathrm{SqErr}_{s-1}(\hat{f}_s, x) + \varepsilon_{s-1} \right) \right)$$

(Applying Lemma 14 and relaxing $s \le m$ to $s \le M$)

$$\le O\left(\tau_1 + M \cdot \max_{m \in \{2,\dots,M\}} \frac{\tau_m}{\gamma_m} \cdot \left(\sum_{s=1}^{M} \gamma_s \overline{\mathsf{doec}}_{\gamma_s,\varepsilon_s}(\mathcal{F}_x, \Lambda) + \sum_{s=2}^{M} \gamma_s^2 \left(\mathrm{SqErr}_{s-1}(\hat{f}_s, x) + \varepsilon_{s-1} \right) \right) \right)$$

(Relaxing τ_m/γ_m to the maximum)

■

If the offline regression oracle provides a guarantee on the estimation error on average over all contexts $x \sim D_X$, we can use Lemma 15 to derive an average regret bound over the context distribution $\mathcal{D}_X$. We restate the theorem here for clarity.

Theorem 4 *Suppose $\tau_m \ge 2\tau_{m-1}$ for all m. Let $\delta_m = \frac{\delta}{m(m+1)}$. For any $\delta \in (0,1)$, $\Lambda \subseteq \Delta(\mathcal{A})$, with probability at least $1-\delta$, the Λ-Regret of OE2D is bounded as:* $\mathbb{E}\left[\mathrm{Regret}_\Lambda(T, \textit{OE2D})\right] \le$

$$\widetilde{O}\left(\tau_1 + \max_{m \in \{2,\dots,M\}} \frac{\tau_m}{\gamma_m} \cdot \left(\max_{n \in [M]} \gamma_n \mathbb{E}\left[\overline{\mathsf{doec}}_{\gamma_n,\varepsilon_n}(\mathcal{F}_x, \Lambda) \right] + \max_{n \in \{2,\dots,M\}} \gamma_n^2 \left(\mathrm{Reg}_{\mathrm{off}}(\mathcal{F}, \tau_{n-1}/2, \delta_n) + \varepsilon_{n-1} \right) \right) \right),$$

where we use the convention that $\gamma_1 = 0$ and $\gamma_1 \overline{\mathsf{doec}}_{\gamma_1,\varepsilon_1}(\mathcal{F}_x, \Lambda)$ is $\lim_{\gamma \to 0} \gamma \overline{\mathsf{doec}}_{\gamma,\varepsilon_1}(\mathcal{F}_x, \Lambda) = \overline{V}^*_{\varepsilon_1}(\mathcal{F}_x, \Lambda)$, the value of the relaxed F-design problem solved in the first epoch.

Proof [Proof of Theorem 4] Since in Lemma 15, we have a per-context total regret bound, we can directly use it here to prove this theorem by taking expectation over $x \sim \mathcal{D}_X$, and applying the

regression guarantee. Therefore, we have

$$\begin{aligned}
&\sum_{t=1}^{T} \mathbb{E}_{x_t \sim \mathcal{D}_X} \left[\mathrm{Reg}(\pi_{m(t)} \mid x_t) \right] \\
&\leq \widetilde{O}\left(\tau_1 + \max_{m \in \{2,\ldots,M\}} \frac{\tau_m}{\gamma_m} \cdot \left(\sum_{s=1}^{M} \gamma_s \mathbb{E}_{x \sim \mathcal{D}_X} \left[\overline{\mathsf{doec}}_{\gamma_s, \varepsilon_s}(\mathcal{F}_x, \Lambda) \right] \right. \right. \\
&\qquad \left. \left. + \sum_{m=2}^{M} \gamma_s^2 \left(\mathbb{E}_{x \sim \mathcal{D}_X} \left[\mathrm{SqErr}_{s-1}(\hat{f}_s, x) \right] + \varepsilon_{s-1} \right) \right) \right) \\
&\leq \widetilde{O}\left(\tau_1 + \max_{m \in \{2,\ldots,M\}} \frac{\tau_m}{\gamma_m} \cdot \left(\sum_{s=1}^{M} \gamma_s \mathbb{E}_{x \sim \mathcal{D}_X} \left[\overline{\mathsf{doec}}_{\gamma_s, \varepsilon_s}(\mathcal{F}_x, \Lambda) \right] \right. \right. \\
&\qquad \left. \left. + \sum_{s=2}^{M} \gamma_s^2 \left(\mathrm{Reg}_{\mathrm{off}}(\mathcal{F}, \tau_{s-1}/2, \delta) + \varepsilon_{s-1} \right) \right) \right) \\
&\leq \widetilde{O}\left(\tau_1 + \max_{m \in \{2,\ldots,M\}} \frac{\tau_m}{\gamma_m} \cdot \left(\max_{s \in [M]} \gamma_s \mathbb{E}_{x \sim \mathcal{D}_X} \left[\overline{\mathsf{doec}}_{\gamma_s, \varepsilon_s}(\mathcal{F}_x, \Lambda) \right] \right. \right. \\
&\qquad \left. \left. + \max_{s \in \{2,\ldots,M\}} \gamma_s^2 \left(\mathrm{Reg}_{\mathrm{off}}(\mathcal{F}, \tau_{s-1}/2, \delta) + \varepsilon_{s-1} \right) \right) \right)
\end{aligned}$$

In the first inequality, we apply Lemma 15 and take expectation over $x \sim \mathcal{D}_X$; in the second inequality, we apply the regression guarantee that $\mathbb{E}_{x \sim \mathcal{D}_X} \left[\mathrm{SqErr}_s(\hat{f}_s, x) \right] \leq \mathrm{Reg}_{\mathrm{off}}(\mathcal{F}, \tau_{s-1} - \tau_{s-2}, \delta_s) \leq \mathrm{Reg}_{\mathrm{off}}(\mathcal{F}, \tau_{s-1}/2, \delta_s)$ because event E happens and $\mathrm{Reg}_{\mathrm{off}}$ is monotonically decreasing in the sample size; in the last inequality, we relax $\sum_{s=1}^{M}$ to $M \cdot \max_{s \in [M]}$ and $\sum_{s=2}^{M}$ to $M \cdot \max_{s \in \{2,\ldots,M\}}$. ■

B.6. Regret Guarantees under Two Epoch Schedules

In this subsection, we instantiate Theorem 4 under two concrete epoch schedules: a doubling schedule and a small epoch schedule, which in particular verify the claimed regret bounds of OE2D in the three running examples. We first present Assumption 1, which is satisfied by all three running examples with $D = A$, d and $1/h$ respectively, as well as when the Eluder dimension of $\mathcal{F}_x$ is polylogarithmic in scaling factor ε for all x (Proposition 34, under the trivial relaxation $\overline{\mathsf{Coverage}} = \mathsf{Coverage}_{\varepsilon}$); and the offline oracle implements ERM with a finite class $\mathcal{F}$ (e.g., Agarwal et al., 2012).

Assumption 1 *The following hold:*

- *There exists a constant $D > 0$, such that for any context $x \in \mathcal{X}$, we have $\max_{x \in \mathcal{X}} \overline{\mathsf{doec}}_{\gamma, \varepsilon}(\mathcal{F}_x, \Lambda) \leq \frac{D}{\gamma} \mathrm{polylog}\left(\frac{1}{\varepsilon}\right)$.*
- *The offline regression oracle ensures that $\mathrm{Reg}_{\mathrm{off}}(\mathcal{F}, T, \delta) \leq \frac{\log(|\mathcal{F}|/\delta)}{T}$.*

B.6.1. Regret Bound with Doubling Schedule

Definition 16 (Doubling Schedule) *A schedule of τ_m, γ_m, ε_m, $m = 1, \ldots, M$ is called a doubling schedule, s.t.*

$$M = \log(T), \tau_m = 2^m, \gamma_1 = 0, \gamma_m = \sqrt{\frac{D}{\mathrm{Reg}_{\mathrm{off}}(\mathcal{F}, \tau_{m-1}/2, \delta)}}, \varepsilon_m = \frac{1}{T}.$$

Corollary 17 *If Assumption 1 holds, by using the doubling schedule (Defined in Theorem 16), OE2D achieves a regret of* $\widetilde{O}\left(\sqrt{DT\log|\mathcal{F}|}\right)$.

Proof For every $m \in [M]$, by the assumption we have

$$\max_{m\in[M]} \frac{\tau_m}{\gamma_m} = \max_{m\in[M]} \tau_m \sqrt{\frac{\mathrm{Reg}_{\mathrm{off}}(\mathcal{F}, \tau_{m-1}/2, \delta)}{D}} = O\left(\sqrt{\frac{T\log(|\mathcal{F}|/\delta)}{D}}\right)$$

Then according to Theorem 4, we have

$$\begin{aligned}
&\sum_{t=1}^{T} \mathbb{E}_{x_t\sim\mathcal{D}_X}\left[\mathrm{Reg}(\pi_{m(t)} \mid x_t)\right] \\
&\leq \widetilde{O}\left(\max_m \frac{\tau_m}{\gamma_m} \cdot \left(\begin{array}{c}\max_{m\in[M]} \gamma_m \mathbb{E}_{x\sim\mathcal{D}_X}\left[\overline{\mathsf{doec}}_{\gamma_m,\varepsilon_m}(\mathcal{F}_x, \Lambda)\right] \\ + \gamma_m^2\left(\mathrm{Reg}_{\mathrm{off}}(\mathcal{F}, \tau_{m-1}/2, \delta) + \varepsilon_{m-1}\right)\end{array}\right)\right) && \text{(Applying Theorem 4)} \\
&\leq \widetilde{O}\left(\max_m \frac{\tau_m}{\gamma_m} \cdot \left(\max_{m\in[M]} D\mathrm{polylog}(\frac{1}{\varepsilon_m}) + D\right)\right) \\
& && \text{(By the assumption on } \overline{\mathsf{doec}} \text{ and the choice of } \gamma_m) \\
&\leq \widetilde{O}\left(\sqrt{\frac{T\log(|\mathcal{F}|/\delta)}{D}} \cdot D\mathrm{polylog}(T)\right) && \text{(By Algebra)} \\
&= \widetilde{O}\left(\sqrt{TD\log(|\mathcal{F}|/\delta)}\right)
\end{aligned}$$

■

B.6.2. REGRET BOUND WITH SMALL EPOCH SCHEDULE

Suppose that the total time horizon T is known in advance, we can further reduce the number of epochs to $O(\log\log(T))$, for a discrete action space setting, by using the following small epoch schedule.

Definition 18 (Small Epoch Schedule) *A schedule of* $\tau_m, \gamma_m, \varepsilon_m$, $m = 1, \ldots, M$ *is called a small epoch schedule, s.t.*

$$M = \log\log(T), \tau_m = \left\lfloor 2T^{1-2^{-m}} \right\rfloor, \gamma_1 = 1, \gamma_m = \sqrt{\frac{D}{\mathrm{Reg}_{\mathrm{off}}(\mathcal{F}, \tau_{m-1} - \tau_{m-2}, \delta)}}, \varepsilon_m = \frac{1}{T}.$$

Corollary 19 *If Assumption 1 holds, by using the small epoch schedule (defined in Theorem 18), OE2D achieves a regret of* $\widetilde{O}\left(\sqrt{DT\log|\mathcal{F}|}\right)$.

Proof For every $m \in [M]$, by assumption we have

$$\max_{m\in[M]} \frac{\tau_m}{\gamma_m} = \max_{m\in[M]} \sqrt{\frac{\tau_m^2 \log(|\mathcal{F}|/\delta)}{D(\tau_{m-1} - \tau_{m-2})}} \leq \sqrt{\frac{T\log(|\mathcal{F}|/\delta)}{D}}$$

Then according to Theorem 4, we have

$$\begin{aligned}
&\sum_{t=1}^{T} \mathbb{E}_{x_t \sim \mathcal{D}_X} \left[\mathrm{Reg}(\pi_{m(t)} \mid x_t)\right] \\
&\leq \widetilde{O}\left(\max_m \frac{\tau_m}{\gamma_m} \cdot \left(\begin{array}{c} \max_{m\in[M]} \gamma_m \mathbb{E}_{x\sim\mathcal{D}_X}\left[\overline{\mathsf{doec}}_{\gamma_m,\varepsilon_m}(\mathcal{F}_x, \Lambda)\right] \\ + \gamma_m^2 \left(\mathrm{Reg}_{\mathrm{off}}(\mathcal{F}, \tau_{m-1} - \tau_{m-2}, \delta) + \varepsilon_{m-1}\right)\end{array}\right)\right) \\
&\qquad\qquad \text{(Applying Theorem 4)} \\
&\leq \widetilde{O}\left(\max_m \frac{\tau_m}{\gamma_m} \cdot \left(\max_{m\in[M]} D\mathrm{polylog}(\frac{1}{\varepsilon}) + D\right)\right) \\
&\qquad\qquad \text{(By the assumption on } \overline{\mathsf{doec}} \text{ and the choice of } \gamma_m) \\
&\leq \widetilde{O}\left(\sqrt{\frac{T\log(|\mathcal{F}|/\delta)}{D}} \cdot D\mathrm{polylog}(T)\right) \qquad \text{(By the maximum of } \tau_m/\gamma_m) \\
&= \widetilde{O}\left(\sqrt{TD\log(|\mathcal{F}|/\delta)}\right)
\end{aligned}$$

■

Appendix C. Proofs from Section 3 Part 2: Relaxations Justification and Regret Guarantees

C.1. Computational Necessity of Relaxation

We show the following lemma on the computational hardness of calculating coverage, which is inspired by a computational hardness result of bandit learning (Brukhim et al., 2025).

Proposition 20 (Evaluating the original coverage is NP-hard) *There is a polynomial-time mapping from 3-CNF formulas ϕ (with m clauses over n variables) to instances $(\mathcal{A}_\phi, \mathcal{G}_\phi, p, \varepsilon)$ with a designated action $a_0 \in \mathcal{A}_\phi$, where $\mathcal{A}_\phi$ has only two actions, p is a distribution over $\mathcal{A}_\phi$, $\varepsilon > 0$, and $\mathcal{G}_\phi \subseteq [0,1]^{\mathcal{A}_\phi}$ is a finite, succinctly represented reward class—each $g \in \mathcal{G}_\phi$ is specified by an index from which $g(a)$ is computable in $\mathrm{poly}(n, m)$ time for every action a. The mapping guarantees*

$$\mathsf{Coverage}_\varepsilon(p, \delta_{a_0}; \mathcal{G}_\phi) \begin{cases} = \frac{1}{\varepsilon}, & \phi \text{ satisfiable}, \\ = \frac{1}{\varepsilon+1}, & \phi \text{ unsatisfiable}. \end{cases}$$

Consequently, unless $\mathsf{P} = \mathsf{NP}$, *no polynomial-time algorithm evaluates the original coverage* $\mathsf{Coverage}_\varepsilon(\cdot,\cdot;\mathcal{G})$ *for general succinctly represented classes* $\mathcal{G}$.

Proof Given ϕ over variables $x_1, \ldots, x_n$, use just two actions $\mathcal{A}_\phi = \{a_0, a_1\}$ and set

$$\mathcal{G}_\phi = \{\mathbf{0}\} \cup \{g_x : x \in \{0,1\}^n\}, \quad \text{where} \quad g_x(a_0) = 1, \qquad g_x(a_1) = \mathbf{1}\left[x \text{ does not satisfy } \phi\right].$$

Each g_x takes values in $\{0,1\} \subseteq [0,1]$, and given the index x both entries are computable in time $\mathrm{poly}(n, m)$ (testing whether x satisfies ϕ); the class is finite. Put $p(a_0) = 0$ and $p(a_1) = 1$, and fix any $\varepsilon > 0$.

Consider a difference $h = g - g'$ with $g, g' \in \mathcal{G}_\phi$. Since $g_x(a_0) = 1$ for every x while $\mathbf{0}(a_0) = 0$, we have $h(a_0) \neq 0$ if and only if exactly one of g, g' is $\mathbf{0}$; then $h = \pm g_x$ for some x, so $h(a_0)^2 = 1$ and

$$\sum_{a \in \mathcal{A}_\phi} p(a)\, h(a)^2 = p(a_1)\, h(a_1)^2 = \mathbf{1}\left[x \text{ does not satisfy } \phi\right].$$

(If $h(a_0) = 0$ the numerator $h(a_0)^2$ vanishes and the pair does not contribute to the supremum.) Therefore

$$\mathsf{Coverage}_\varepsilon(p, \delta_{a_0}; \mathcal{G}_\phi) = \sup_{x \in \{0,1\}^n} \frac{h(a_0)^2}{\varepsilon + \mathbf{1}\left[x \text{ does not satisfy } \phi\right]} = \frac{1}{\varepsilon + \min_x \mathbf{1}\left[x \text{ does not satisfy } \phi\right]}.$$

Finally, $\min_x \mathbf{1}\left[x \text{ does not satisfy } \phi\right] = 0$ iff some x satisfies ϕ, i.e. iff ϕ is satisfiable, in which case the coverage equals $\frac{1}{\varepsilon}$; otherwise the minimum is 1 and the coverage equals $\frac{1}{\varepsilon+1}$. A polynomial-time evaluator of the coverage would thus decide 3-SAT, giving $\mathsf{P} = \mathsf{NP}$. ■

In particular, the unrelaxed exploitative F-design (Eq. (3) with $\overline{\mathsf{Coverage}} = \mathsf{Coverage}_\varepsilon$) cannot be run in polynomial time on all classes unless $\mathsf{P} = \mathsf{NP}$, in contrast to the relaxed F-design, whose closed-form coverage makes it a polynomial-time convex program (Lemma 23).

C.2. Validity of the Relaxed Coverages

We first record an elementary linear-algebra inequality, used in the generalized linear case of the validity argument below.

Lemma 21 (Trace bound for a quadratic form) *For any $A \succ 0$, $C \succeq 0$, and vector u,*

$$u^\top C u \leq \operatorname{tr}\left(A^{-1}C\right) \cdot u^\top A u.$$

Proof Write $M := A^{-1/2} C A^{-1/2} \succeq 0$. Then $u^\top C u = (A^{1/2}u)^\top M (A^{1/2}u) \leq ||M||_{\mathrm{op}}\, u^\top A u$, and $||M||_{\mathrm{op}} = \lambda_{\max}(M) \leq \operatorname{tr}(M) = \operatorname{tr}\left(A^{-1}C\right)$, where the inequality holds because $M \succeq 0$ (its largest eigenvalue is at most the sum of all eigenvalues) and the last equality is the cyclic property of the trace. ■

Recall from Section 3 that OE2D computes its action sampling distribution by solving the relaxed exploitative F-design (Eq. (3)), in which the original coverage $\mathsf{Coverage}_\varepsilon$ (Eq. (2)) is replaced by a relaxed coverage $\overline{\mathsf{Coverage}}$. The only property of $\overline{\mathsf{Coverage}}$ that our regret analysis hinges on is that it is a *pointwise upper bound* of $\mathsf{Coverage}_\varepsilon$: any good coverage guarantee with respect to $\overline{\mathsf{Coverage}}$ then implies the same guarantee with $\mathsf{Coverage}_\varepsilon$. In this subsection, we verify this property for the relaxed coverages defined in our running examples (Lemma 22). Notably, none of these relaxations depends on the "cushion parameter" ε: each of them upper bounds $\mathsf{Coverage}_\varepsilon$ for every $\varepsilon \geq 0$, and we accordingly write them without the subscript ε.

Throughout this subsection, we fix a context x and abbreviate $\mathcal{G} = \mathcal{F}_x$ and $\phi(a) = \phi(x, a)$; in the (generalized) linear settings, we write $\Theta_x := \{\theta(x) : \theta \in \Theta\}$ for the per-context parameter set. Since the relaxed coverages below may have zero denominators, we adopt the conventions $\frac{0}{0} := 0$ and $\frac{c}{0} := +\infty$ for $c > 0$; in the (generalized) linear settings, Σ_p^{-1} denotes the Moore–Penrose pseudo-inverse when Σ_p is singular, and $\operatorname{tr}\left(\Sigma_p^{-1}\Sigma_\lambda\right) := +\infty$ when $\operatorname{range}(\Sigma_\lambda) \not\subseteq \operatorname{range}(\Sigma_p)$.

Lemma 22 (Relaxed coverages upper bound the original coverage) *Let p be a nonnegative measure over $\mathcal{A}$, $\lambda \in \Lambda$, and $\varepsilon \geq 0$. Then $\mathsf{Coverage}_\varepsilon(p, \lambda; \mathcal{G}) \leq \overline{\mathsf{Coverage}}_\varepsilon(p, \lambda; \mathcal{G})$ holds in each of the following settings:*

1. ***(Discrete action space)*** *$|\mathcal{A}| < \infty$, $\mathcal{G} \subseteq [0,1]^{\mathcal{A}}$, $\Lambda = \{\delta_a : a \in \mathcal{A}\}$, and $\overline{\mathsf{Coverage}}_\varepsilon(p, \lambda; \mathcal{G}) = \sum_{a \in \mathcal{A}} \frac{\lambda(a)}{p(a)}$.*

2. ***(Per-context generalized linear reward)*** *$\mathcal{G} = \left\{a \mapsto \sigma(\phi(a)^\top \theta) : \theta \in \Theta_x\right\}$, $\Lambda = \{\delta_a : a \in \mathcal{A}\}$, with link function σ satisfying $0 < \underline{L} \leq \sigma' \leq \overline{L}$, and $\overline{\mathsf{Coverage}}_\varepsilon(p, \lambda; \mathcal{G}) = \kappa^2 \operatorname{tr}\left(\Sigma_p^{-1} \Sigma_\lambda\right)$ with $\kappa := \overline{L}/\underline{L}$.*

3. ***(h-smoothed regret)*** *$\mathcal{G} \subseteq [0,1]^{\mathcal{A}}$, $\lambda \in \Delta_h^\mu(\mathcal{A})$, p admits a density with respect to μ, and $\overline{\mathsf{Coverage}}_\varepsilon(p, \lambda; \mathcal{G}) = \frac{1}{h}\mathbb{E}_{a \sim \mu}\left[\frac{\lambda(a)}{p(a)}\right]$, where $\lambda(a)$ and $p(a)$ denote the densities of λ and p with respect to μ.*

Proof Fix $g, g' \in \mathcal{G}$ and let $h := g - g'$ denote the function difference. By the definition of $\mathsf{Coverage}_\varepsilon$ (Eq. (2)) and since $\varepsilon \geq 0$, it suffices to show $(\mathbb{E}_{a \sim \lambda}[h(a)])^2 \leq \overline{\mathsf{Coverage}}_\varepsilon(p, \lambda; \mathcal{G}) \cdot \mathbb{E}_{a \sim p}[h(a)^2]$ in each setting. We may assume $\overline{\mathsf{Coverage}}_\varepsilon(p, \lambda; \mathcal{G}) < \infty$, as the claim is trivial otherwise.

Setting 1. Finiteness of coverage implies $p(a) > 0$ whenever $\lambda(a) > 0$. By the Cauchy–Schwarz inequality (with sums restricted to the support of λ),

$$\left(\sum_{a \in \mathcal{A}} \lambda(a) h(a)\right)^2 \leq \left(\sum_{a \in \mathcal{A}} \frac{\lambda(a)^2}{p(a)}\right)\left(\sum_{a \in \mathcal{A}} p(a) h(a)^2\right) \leq \left(\sum_{a \in \mathcal{A}} \frac{\lambda(a)}{p(a)}\right)\left(\sum_{a \in \mathcal{A}} p(a) h(a)^2\right),$$

where the last inequality uses $\lambda(a)^2 \leq \lambda(a)$.

Setting 2. By the mean value theorem, $h(a) = \sigma'(\xi_a)\, \phi(a)^\top u$ for some ξ_a, where $u := \theta - \theta'$; since $\underline{L} \leq \sigma' \leq \overline{L}$, Jensen's inequality gives

$$(\mathbb{E}_{a \sim \lambda}[h(a)])^2 \leq \mathbb{E}_{a \sim \lambda}\left[h(a)^2\right] \leq \overline{L}^2\, u^\top \Sigma_\lambda u, \qquad \mathbb{E}_{a \sim p}\left[h(a)^2\right] \geq \underline{L}^2\, u^\top \Sigma_p u.$$

Then by applying Theorem 21 with $C = \Sigma_\lambda$ and $A = \Sigma_p$ we have $u^\top \Sigma_\lambda u \leq \operatorname{tr}\left(\Sigma_p^{-1}\Sigma_\lambda\right)\, u^\top \Sigma_p u$. Combining the three inequalities gives $(\mathbb{E}_{a \sim \lambda}[h(a)])^2 \leq \kappa^2 \operatorname{tr}\left(\Sigma_p^{-1}\Sigma_\lambda\right)\, \mathbb{E}_{a \sim p}\left[h(a)^2\right]$. The per-context linear reward is the special case $\sigma = \mathrm{id}$ ($\kappa = 1$).

Setting 3. Below, $h(a)$ denotes the function difference and h the smoothing parameter. Finiteness implies $p(a) > 0$ for μ-almost every a with $\lambda(a) > 0$. By the Cauchy–Schwarz inequality with respect to μ,

$$(\mathbb{E}_{a \sim \mu}[\lambda(a) h(a)])^2 \leq \mathbb{E}_{a \sim \mu}\left[\frac{\lambda(a)^2}{p(a)}\right] \cdot \mathbb{E}_{a \sim \mu}\left[p(a)\, h(a)^2\right] \leq \frac{1}{h}\mathbb{E}_{a \sim \mu}\left[\frac{\lambda(a)}{p(a)}\right] \cdot \mathbb{E}_{a \sim \mu}\left[p(a) h(a)^2\right],$$

where the last inequality uses $\lambda(a) \leq \frac{1}{h}$ (as $\lambda \in \Delta_h^\mu(\mathcal{A})$). ■

C.3. Relaxed DOEC Bounds for the Running Examples

Having verified that the relaxed coverages are valid (Lemma 22), we now bound the induced relaxed DOEC in each running example, and in doing so justify that OE2D can implement its per-step exploitative F-design (line 9, Eq. (3)) by solving a convex program. The key structural fact is that each relaxed coverage is the directional derivative of a concave *barrier*, which both turns Eq. (3) into a barrier-regularized reward maximization and produces the certified bound $\overline{V}$ of Table 2 through a first-order optimality argument.

Lemma 23 (Relaxed DOEC bounds for the running examples) *Fix a context x, abbreviate $\mathcal{G} = \mathcal{F}_x$, and let $\hat{g} \in \mathcal{G}$, $\gamma > 0$, and $\varepsilon \geq 0$. In each of the three running examples of Lemma 22, the relaxed exploitative F-design (Eq. (3)) is solved by the same p^* as the concave maximization*

$$p^* \in \operatorname*{argmax}_{p \in \mathrm{co}(\Lambda)} \left\{ \mathbb{E}_{a\sim p}\left[\hat{g}(a)\right] + \tfrac{1}{\gamma} B(p) \right\}, \qquad B(p) = \begin{cases} \sum_{a\in\mathcal{A}} \log p(a), & \textit{discrete}, \\ \kappa^2 \log\det \Sigma_p, & \textit{generalized linear}, \\ \frac{1}{h}\mathbb{E}_{a\sim\mu}\left[\log p(a)\right], & \textit{h-smoothed}, \end{cases}$$

where B is the concave barrier whose directional derivative recovers the relaxed coverage, $\overline{\mathsf{Coverage}}(p, \lambda; \mathcal{G}) = \langle \nabla B(p), \lambda \rangle$.

Furthermore, $\overline{\mathsf{doec}}_{\gamma,\varepsilon}(\hat{g}, \mathcal{G}, \Lambda) = \overline{V}$, with $\overline{V} = \frac{|\mathcal{A}|}{\gamma}$, $\frac{\kappa^2 d}{\gamma}$, $\frac{1}{\gamma h}$ in the discrete, generalized linear, and h-smoothed settings, respectively.

Proof Write $\Phi(p, \lambda) := \mathbb{E}_{a\sim\lambda}\left[\hat{g}(a)\right] - \mathbb{E}_{a\sim p}\left[\hat{g}(a)\right] + \frac{1}{\gamma}\overline{\mathsf{Coverage}}_{\varepsilon}(p, \lambda; \mathcal{G})$, so that, by Definition 2, $\overline{\mathsf{doec}}_{\gamma,\varepsilon}(\hat{g}, \mathcal{G}, \Lambda) = \min_{p\in\mathrm{co}(\Lambda)} \max_{\lambda\in\Lambda} \Phi(p, \lambda)$. It therefore suffices to find one $p^* \in \mathrm{co}(\Lambda)$ with $\Phi(p^*, \lambda) \leq \overline{V}$ for all $\lambda \in \Lambda$.

A unified barrier. In each setting the relaxed coverage is the directional derivative of a concave barrier B, in the sense that

$$\overline{\mathsf{Coverage}}(p, \lambda; \mathcal{G}) = \langle \nabla B(p), \lambda \rangle, \qquad \text{and } \langle \nabla B(p), p \rangle = \gamma \overline{V} \text{ is constant in } p,$$

where $\langle \cdot, \cdot \rangle$ is the Euclidean inner product over $\mathcal{A}$ (discrete, generalized linear) or the $L^2(\mu)$ inner product (h-smoothed). The convex program in Table 2 is exactly $\max_{p\in\Lambda} G(p)$ with $G(p) := \mathbb{E}_{a\sim p}\left[\hat{g}(a)\right] + \frac{1}{\gamma}B(p)$.

Certification by first-order optimality. Optimality of p^* for the concave program over the convex set Λ is the variational inequality $\langle \nabla G(p^*), \lambda - p^* \rangle \leq 0$ for all $\lambda \in \Lambda$. Substituting $\nabla G(p^*) = \hat{g} + \frac{1}{\gamma}\nabla B(p^*)$ and rearranging,

$$\Phi(p^*, \lambda) = \langle \hat{g}, \lambda - p^* \rangle + \frac{1}{\gamma}\langle \nabla B(p^*), \lambda \rangle \leq \frac{1}{\gamma}\langle \nabla B(p^*), p^* \rangle = \overline{V} \qquad \text{for all } \lambda \in \Lambda,$$

using $\overline{\mathsf{Coverage}}_{\varepsilon}(p^*, \lambda; \mathcal{G}) = \langle \nabla B(p^*), \lambda \rangle$ in the first equality and the constant diagonal value in the last. Maximizing over $\lambda \in \Lambda$ proves the displayed certification, and $\overline{\mathsf{doec}}_{\gamma,\varepsilon}(\hat{g}, \mathcal{G}, \Lambda) = \min_p \max_\lambda \Phi(p, \lambda) \leq \max_\lambda \Phi(p^*, \lambda) \leq \overline{V}$.

Furthermore, for all p, note that $\Phi(p, \lambda)$ is linear in λ, therefore,

$$\max_{\lambda\in\Lambda} \Phi(p, \lambda) = \max_{\lambda\in\mathrm{co}(\Lambda)} \Phi(p, \lambda) \geq \Phi(p, p) = \langle \nabla B(p), p \rangle = \gamma \overline{V}.$$

This implies that $\overline{\mathsf{doec}}_{\gamma,\varepsilon}(\hat{g}, \mathcal{G}, \Lambda) \geq \overline{V}$.

In summary,

$$\overline{\mathsf{doec}}_{\gamma,\varepsilon}(\hat{g}, \mathcal{G}, \Lambda) = \overline{V}.$$

Instantiating the barrier. The three settings differ only in B and the constant $\langle \nabla B(p), p \rangle$:

- **Discrete** ($\Lambda = \Delta(\mathcal{A})$): $B(p) = \sum_a \log p(a)$, so $\langle \nabla B(p), \lambda \rangle = \sum_a \frac{\lambda(a)}{p(a)}$ and $\langle \nabla B(p), p \rangle = \sum_a \frac{p(a)}{p(a)} = |\mathcal{A}|$.
- **Generalized linear** ($\Lambda = \Delta(\mathcal{A})$): $B(p) = \kappa^2 \log\det \Sigma_p$, so $\langle \nabla B(p), \lambda \rangle = \kappa^2 \operatorname{tr}\left(\Sigma_p^{-1}\Sigma_\lambda\right)$ (using $\frac{\partial}{\partial p(a)} \log\det \Sigma_p = \phi(a)^\top \Sigma_p^{-1} \phi(a)$) and $\langle \nabla B(p), p \rangle = \kappa^2 \operatorname{tr}\left(\Sigma_p^{-1}\Sigma_p\right) = \kappa^2 d$.
- **h-smoothed** ($\Lambda = \Delta_h^\mu(\mathcal{A})$): $B(p) = \frac{1}{h}\mathbb{E}_{a\sim\mu}[\log p(a)]$, so $\langle \nabla B(p), \lambda \rangle = \frac{1}{h}\mathbb{E}_{a\sim\mu}\left[\frac{\lambda(a)}{p(a)}\right]$ and $\langle \nabla B(p), p \rangle = \frac{1}{h}\mathbb{E}_{a\sim\mu}\left[\frac{p(a)}{p(a)}\right] = \frac{1}{h}$.

This gives $\overline{V} = \frac{|\mathcal{A}|}{\gamma}$, $\frac{\kappa^2 d}{\gamma}$, $\frac{1}{\gamma h}$, i.e. the stated bounds. ■

Based on the above lemma, we can derive the closed-form solutions of the relaxed exploitative F-design in the discrete and h-smoothed settings, which are the (capped) inverse gap weighting distributions (Abe and Long, 1999; Foster and Rakhlin, 2020). Especially in the h-smoothed setting, the closed-form solution is a novel variant of inverse gap weighting that incorporates the smoothing parameter h into the weights and caps.

Corollary 24 (Closed-form F-design via inverse gap weighting) *In the discrete and h-smoothed settings of Lemma 23, the certifying distribution p^* admits a closed form. Writing $\Delta_a := \max_{a' \in \mathcal{A}} \hat{g}(a') - \hat{g}(a)$ for the estimated suboptimality gap of action a,*

$$p^*(a) = \frac{1}{\nu + \gamma\Delta_a} \quad (\textit{discrete}), \qquad p^*(a) = \min\left\{\frac{1}{h}, \frac{1}{\nu + \gamma h \Delta_a}\right\} \quad (\textit{h-smoothed}),$$

where in each case the multiplier ν is chosen so that p^* is normalized.

Proof [Proof sketch] By Lemma 23, p^* maximizes the concave objective $G(p) = \mathbb{E}_{a\sim p}[\hat{g}(a)] + \frac{1}{\gamma}B(p)$ over Λ, with barrier $B(p) = \sum_a \log p(a)$ in the discrete setting and $B(p) = \frac{1}{h}\mathbb{E}_{a\sim\mu}[\log p(a)]$ in the h-smoothed setting. In the discrete setting, stationarity of G on the simplex gives a multiplier ν for the constraint $\sum_a p(a) = 1$ with $\hat{g}(a) + \frac{1}{\gamma p^*(a)} = \nu$, equivalently $p^*(a) = \frac{1}{\nu + \gamma\Delta_a}$ after absorbing $\max_{a'} \hat{g}(a')$ into ν. In the h-smoothed setting the additional box constraint $p(a) \leq \frac{1}{h}$ of $\Delta_h^\mu(\mathcal{A})$ enters the KKT conditions and caps the solution at $p^*(a) = \min\left\{\frac{1}{h}, \frac{1}{\nu + \gamma h \Delta_a}\right\}$; in both cases ν is set by normalization. The generalized linear program has no closed form but is concave and efficiently solvable. ■

Discrete	Relaxed coverage	$\overline{\mathsf{Coverage}}_\varepsilon(p,\lambda;\mathcal{G}) = \sum_{a\in\mathcal{A}} \frac{\lambda(a)}{p(a)}$
	Equivalent convex program	$\max_{p\in\Delta(\mathcal{A})} \mathbb{E}_{a\sim p}[\hat{g}(a)] + \frac{1}{\gamma}\sum_{a\in\mathcal{A}} \log p(a)$
	Solution	$p^*(a) = \frac{1}{\nu+\gamma\Delta_a}$, ν is such that $\sum_{a\in\mathcal{A}} p^*(a) = 1$
	Certified DOEC bound $\overline{V}$	$\frac{\lvert\mathcal{A}\rvert}{\gamma}$
Linear	Relaxed coverage	$\overline{\mathsf{Coverage}}_\varepsilon(p,\lambda;\mathcal{G}) = \operatorname{tr}\left(\Sigma_p^{-1}\Sigma_\lambda\right)$
	Equivalent convex program	$\max_{p\in\Delta(\mathcal{A})} \mathbb{E}_{a\sim p}[\hat{g}(a)] + \frac{1}{\gamma}\log\det\left(\Sigma_p\right)$
	Solution	no closed form (solution of the convex program above)
	Certified DOEC bound $\overline{V}$	$\frac{d}{\gamma}$
Generalized linear	Relaxed coverage	$\overline{\mathsf{Coverage}}_\varepsilon(p,\lambda;\mathcal{G}) = \kappa^2 \operatorname{tr}\left(\Sigma_p^{-1}\Sigma_\lambda\right)$
	Equivalent convex program	$\max_{p\in\Delta(\mathcal{A})} \mathbb{E}_{a\sim p}[\hat{g}(a)] + \frac{\kappa^2}{\gamma}\log\det\left(\Sigma_p\right)$
	Solution	no closed form (solution of the convex program above)
	Certified DOEC bound $\overline{V}$	$\frac{\kappa^2 d}{\gamma}$
h-smoothed	Relaxed coverage	$\overline{\mathsf{Coverage}}_\varepsilon(p,\lambda;\mathcal{G}) = \frac{1}{h}\mathbb{E}_{a\sim\mu}\left[\frac{\lambda(a)}{p(a)}\right]$
	Equivalent convex program	$\max_{p\in\Delta_h^\mu(\mathcal{A})} \mathbb{E}_{a\sim\mu}[p(a)\hat{g}(a)] + \frac{1}{\gamma h}\mathbb{E}_{a\sim\mu}[\log p(a)]$
	Solution	$p^*(a) = \min\left\{\frac{1}{h}, \frac{1}{\nu+\gamma h\Delta_a}\right\}$, ν is such that $\mathbb{E}_{a\sim\mu}[p^*(a)] = 1$
	Certified DOEC bound $\overline{V}$	$\frac{1}{\gamma h}$

Table 2: Relaxed coverages for the running examples and the induced relaxed exploitative F-design subproblems; none of them depends on the cushion parameter ε. Here, $\hat{g}$ is the reward estimate, $\Delta_a := \max_{a'\in\mathcal{A}} \hat{g}(a') - \hat{g}(a)$ is the estimated suboptimality gap of action a, $\Sigma_p := \sum_{a\in\mathcal{A}} p(a)\phi(a)\phi(a)^\top$, and ν ensures that the solution distributions are properly normalized. In the h-smoothed setting, $p(a)$ and $\lambda(a)$ denote densities with respect to the base measure μ. The settings and constants $\underline{L}, \overline{L}, \kappa$ are as in Lemma 22.

Appendix D. Proofs from Section 3 Part 3: Extensions

In this section, we discuss several extensions of our main regret analysis when using ERM as the offline regression oracle in Algorithm 1. We will demonstrate how to instantiate our regret bound to handle model misspecification, corruption-robustness, and context-distribution shift settings. The key idea is to modify the offline regression oracle guarantees used in our per-context regret analysis to accommodate these different settings. Specifically, in the finite function class with realizable setting, the expected squared error $\mathbb{E}_{x\sim\mathcal{D}_X}[\mathrm{SqErr}_{m-1}(\hat{f}_m, x)]$ is bounded by $\lesssim \frac{\log(|\mathcal{F}|T/\delta)}{\tau_{m-1}-\tau_{m-2}}$ using standard concentration inequalities. However, when the *realizability* assumption does not hold, for example, in the model misspecification setting, the ERM will not give us a logarithmic squared error bound since the true model f^* may not be in the function class $\mathcal{F}$. Thankfully, we use a lemma from Zhu and Nowak (2022, Lemma 3) to bound the expected squared error by empirical excess square error plus an additional logarithmic term $O\big(\log(|\mathcal{F}|)\log(T/\delta)\big)$.

D.1. Model Misspecification

In practice, our function class $\mathcal{F}$ may not perfectly capture the true reward function. To quantify the impact of such model misspecification on our regret analysis, we adopt the *universal* (uniform) misspecification level (Lattimore et al., 2020) , which measures the worst-case deviation between the best in-class approximation and the true reward function over all context-action pairs:

Assumption 2 (Universal misspecification level) *There exist a function $\tilde{f} \in \mathcal{F}$ and a constant $B \geq 0$ such that*

$$\sup_{x \in \mathcal{X},\, a \in \mathcal{A}} \left| \tilde{f}(x,a) - f^*(x,a) \right| \leq \sqrt{B},$$

i.e. the universal misspecification level $\inf_{f \in \mathcal{F}} \sup_{x \in \mathcal{X}, a \in \mathcal{A}} |f(x,a) - f^*(x,a)|$ is at most $\sqrt{B}$.

Assumption 2 states that some in-class function $\tilde{f} \in \mathcal{F}$ uniformly approximates f^* to within $\sqrt{B}$ over all contexts and actions. Based on this assumption, we extend our main regret analysis to account for the misspecification error. The only place realizability enters our analysis is the per-context OPE bound (Theorem 11), whose proof pairs $\hat{f}_m$ with the out-of-class f^* inside the coverage supremum. The following misspecified version restores it by pairing $\hat{f}_m$ with the in-class surrogate $\tilde{f}$ and paying the universal level $\sqrt{B}$.

Lemma 25 (Per-context OPE under misspecification) *Under Assumption 2, on the same probability-$(1-\delta)$ event as Theorem 11, for all $x \in \mathcal{X}$, $m \in [2, M]$, and $\lambda \in \Delta(\mathcal{A})$,*

$$\left| \widehat{\mathcal{R}}_m(\lambda \mid x) - \mathcal{R}(\lambda \mid x) \right| \leq \sqrt{\mathsf{Coverage}_{\varepsilon_{m-1}}(\pi_{m-1}, \lambda; \mathcal{F}_x \mid x) \left(2\mathrm{SqErr}_{m-1}(\hat{f}_m, x) + 2B + \varepsilon_{m-1} \right)} + \sqrt{B}.$$

Proof Let $\tilde{f} \in \mathcal{F}$ be the surrogate of Assumption 2 and write $\mathcal{R}_{\tilde{f}}(\lambda \mid x) := \mathbb{E}_{a \sim \lambda} \left[\tilde{f}(x,a) \right]$. Since $\hat{f}_m, \tilde{f} \in \mathcal{F}_x$, the pair $(\hat{f}_m, \tilde{f})$ is admissible in the supremum defining Coverage, so repeating the proof of Theorem 11 with $\tilde{f}$ in place of f^* gives

$$\left| \widehat{\mathcal{R}}_m(\lambda \mid x) - \mathcal{R}_{\tilde{f}}(\lambda \mid x) \right| \leq \sqrt{\mathsf{Coverage}_{\varepsilon_{m-1}}(\pi_{m-1}, \lambda; \mathcal{F}_x \mid x) \left(\mathbb{E}_{a \sim \pi_{m-1}} \left[(\hat{f}_m - \tilde{f})^2 \right] + \varepsilon_{m-1} \right)}.$$

By $(a+b)^2 \leq 2a^2 + 2b^2$ and $\sup_{x,a} (\tilde{f} - f^*)^2 \leq B$,

$$\mathbb{E}_{a \sim \pi_{m-1}} \left[(\hat{f}_m - \tilde{f})^2 \right] \leq 2\, \mathbb{E}_{a \sim \pi_{m-1}} \left[(\hat{f}_m - f^*)^2 \right] + 2\, \mathbb{E}_{a \sim \pi_{m-1}} \left[(\tilde{f} - f^*)^2 \right] \leq 2\mathrm{SqErr}_{m-1}(\hat{f}_m, x) + 2B.$$

Finally, the bias from replacing $\tilde{f}$ by f^* is controlled by the universal level: $\left| \mathcal{R}_{\tilde{f}}(\lambda \mid x) - \mathcal{R}(\lambda \mid x) \right| = \left| \mathbb{E}_{a \sim \lambda} \left[\tilde{f} - f^* \right] \right| \leq \sqrt{\mathbb{E}_{a \sim \lambda} \left[(\tilde{f} - f^*)^2 \right]} \leq \sqrt{B}$. The claim follows by the triangle inequality $\left| \widehat{\mathcal{R}}_m - \mathcal{R} \right| \leq \left| \widehat{\mathcal{R}}_m - \mathcal{R}_{\tilde{f}} \right| + \left| \mathcal{R}_{\tilde{f}} - \mathcal{R} \right|$. ■

Using Lemma 25 in place of Lemma 11, the per-context regret bound (Theorem 15) goes through verbatim with $\mathrm{SqErr}_{m-1}(\hat{f}_m, x)$ replaced by $2\mathrm{SqErr}_{m-1}(\hat{f}_m, x) + 2B$ and an additional additive $\sqrt{B}$ per step; both only inflate constants and the B-dependent term, which is absorbed into the bound below.

Corollary 26 (Regret Bound under Misspecification) *If Assumptions 1 and 2 holds, with probability at least $1-\delta$, under the doubling schedule (Theorem 16) but changing $\gamma_m = \sqrt{\frac{D}{B+\log(|\mathcal{F}|T/\delta)/\tau_{m-1}}}$, the total regret of Algorithm 1 satisfies*

$$\sum_{t=1}^{T} \mathbb{E}_{x_t \sim \mathcal{D}_X} \left[\mathrm{Reg}(\pi_{m(t)} \mid x_t) \right] = \widetilde{O}\left(\sqrt{TD\left(\log(|\mathcal{F}|/\delta)\right)} + T\sqrt{BD} \right)$$

The above corollary is directly derived from our per-context regret bound in Lemma 15 by incorporating the misspecification error term, which handles the estimation error term SqErr in a similar manner as in the realizable case but has an additional error term B to account for the misspecification error.

Proof [Proof of Corollary 26] Denote $\hat{f}_m$ as the best ERM predictor returned by the offline regression oracle trained on the dataset collected up in epoch $m-1$, and let $\widetilde{f} \in \mathcal{F}$ be the universal surrogate of Assumption 2, i.e. $\sup_{x,a} \left| \widetilde{f}(x,a) - f^*(x,a) \right| \le \sqrt{B}$.

$$\begin{aligned}
&\sum_{t=\tau_{m-2}+1}^{\tau_{m-1}} \mathbb{E}_{x_t \sim \mathcal{D}_X, a_t \sim \pi_{m-1}} [\mathrm{SqErr}_{m-1}(\hat{f}_m, x_t)] \\
&\le 2 \sum_{t=\tau_{m-2}+1}^{\tau_{m-1}} \left(\hat{f}_m(x_t,a_t) - r_t \right)^2 - \left(f^*(x_t,a_t) - r_t \right)^2 + \log(|\mathcal{F}|T/\delta) \\
&\qquad \text{(By the second inequality of Zhu and Nowak (2022, Lemma 3))} \\
&\le 2 \sum_{t=\tau_{m-2}+1}^{\tau_{m-1}} \left(\widetilde{f}(x_t,a_t) - r_t \right)^2 - \left(f^*(x_t,a_t) - r_t \right)^2 + \log(|\mathcal{F}|T/\delta) \quad (\hat{f}_m \text{ is the ERM predictor}) \\
&\le 3 \sum_{t=\tau_{m-2}+1}^{\tau_{m-1}} \mathbb{E}_{x_t \sim \mathcal{D}_X, a_t \sim \pi_{m-1}} \left[\left(\widetilde{f}(x_t,a_t) - f^*(x_t,a_t) \right)^2 \right] + 2\log(|\mathcal{F}|T/\delta) \\
&\qquad \text{(By the first inequality of Zhu and Nowak (2022, Lemma 3))} \\
&\le 3(\tau_{m-1} - \tau_{m-2})B + 2\log(|\mathcal{F}|T/\delta) \qquad \text{(By the Assumption 2)}
\end{aligned}$$

Then, we plug this bound into the per-context regret bound in Lemma 15 to get the total regret bound as follows:

$$\begin{aligned}
&\sum_{t=1}^{T} \mathbb{E}_{x_t \sim \mathcal{D}_X} \left[\mathrm{Reg}(\pi_{m(t)|x_t)} \right] \le \widetilde{O}\left(\max_m \frac{\tau_m}{\gamma_m} \left(D + \sum_{s=2}^{M} \gamma_s^2 \mathbb{E}_{x \sim \mathcal{D}_X} \left[\mathrm{SqErr}_{s-1}(\hat{f}_s, x) \right] \right) \right) \\
&\qquad \text{(From Lemma 15)} \\
&\le \widetilde{O}\left(\frac{\tau_M}{\gamma_M} \left(D\mathrm{polylog}(\frac{1}{\varepsilon}) + D \right) \right) \qquad (\gamma_m \text{ and } \tau_m/\gamma_m \text{ is non-decreasing in } m) \\
&\le \widetilde{O}\left(\sqrt{TD\left(\log(|\mathcal{F}|/\delta)\right)} + T\sqrt{BD} \right) \qquad (\tau_M/\gamma_M = \widetilde{O}\big(T\sqrt{B/D} + \sqrt{T(\log(|\mathcal{F}|T/\delta))/D}\big))
\end{aligned}$$

■

D.2. Corruption-Robustness

In practice, the observed rewards may be corrupted by an adversary, which can significantly affect the performance of contextual bandit algorithms. We extend our per-context regret analysis to the corruption-robust setting, where the observed rewards may be corrupted by an adversary. Specifically, we consider a general corruption model:

Assumption 3 (Corruption Model) *The generative process of (x_t, a_t, r_t)'s are: first, the adversary specifies rounds for corruptions $\mathcal{C} \subset [T]$, where $|\mathcal{C}| \leq C$. Then for the sequence $(\tilde{x}_t, (\tilde{R}_t(a))_{a\in\mathcal{A}}) \sim D$, when $t \in \mathcal{C}$, the $(x_t, (R_t(a))_{a\in\mathcal{A}})$ can be arbitrarily chosen by the adversary; otherwise, it must equal $(\tilde{x}_t, (\tilde{R}_t(a))_{a\in\mathcal{A}})$. Then the learner chooses a_t and observes $r_t = R_t(a_t)$.*

Under the Assumption 3, we extend our per-context regret analysis to account for the adversarial corruption in the observed rewards.

Corollary 27 (Regret Bound under Corruption) *If Assumptions 1 and 3 hold, then with probability at least $1-\delta$, under the doubling schedule (Theorem 16) but changing $\gamma_m = \sqrt{\frac{D\tau_{m-1}}{C+\log(|\mathcal{F}|T/\delta)}}$, the total regret of Algorithm 1 satisfies*

$$\sum_{t=1}^{T} \mathbb{E}_{x_t\sim\mathcal{D}_X}\left[\mathrm{Reg}(\pi_{m(t)} \mid x_t)\right] = \widetilde{O}\left(\sqrt{TD\left(C+\log(|\mathcal{F}|)\right)}\right)$$

Proof [Proof of Corollary 27] We follow the proof of Lemma 15 but modify the application of the offline regression oracle guarantee to account for the corrupted feedback. Denote $\hat{f}_m$ as the best ERM predictor returned by the offline regression oracle at the beginning of epoch m over the corrupted data. Denote $\widetilde{f}_m$ as the best ERM predictor returned over the uncorrupted data. Denote the dataset collected up in epoch $m-1$ as $\mathcal{S}_{m-1} = \{(x_t, a_t, r_t)\}_{t=\tau_{m-2}+1}^{\tau_{m-1}}$, and the uncorrupted subset as $\widetilde{\mathcal{S}}_{m-1} = \{(\tilde{x}_t, \tilde{a}_t, \tilde{r}_t)\}_{t=\tau_{m-2}+1}^{\tau_{m-1}}$. For every tuple $(x_t, a_t, r_t) \in \mathcal{S}_{m-1}$, there exists a corresponding uncorrupted tuple $(\tilde{x}_t, \tilde{a}_t, \tilde{r}_t)$. For every function $f \in \mathcal{F}$, we have

$$\sum_{t=\tau_{m-2}+1}^{\tau_{m-1}} \left(f(x_t, a_t) - r_t\right)^2 \leq \sum_{t=\tau_{m-2}+1}^{\tau_{m-1}} \left(f(\tilde{x}_t, \tilde{a}_t) - \tilde{r}_t\right)^2 + C.$$

Then, we bound the expected squared error under the corrupted data distribution as follows:

$$\sum_{t=\tau_{m-2}+1}^{\tau_{m-1}} \mathbb{E}_{x_t\sim\mathcal{D}_X, a_t\sim\pi_m}[\mathrm{SqErr}_{m-1}(\hat{f}_m, x_t)]$$

$$\leq 2\sum_{t=\tau_{m-2}+1}^{\tau_{m-1}} \left(\hat{f}_m(x_t,a_t)-r_t\right)^2 - (f^*(x_t,a_t)-r_t)^2 + \log(|\mathcal{F}|T/\delta)$$

(By the second inequality of Zhu and Nowak (2022, Lemma 3))

$$\leq 2\sum_{t=\tau_{m-2}+1}^{\tau_{m-1}} \left(\widetilde{f}(x_t,a_t)-r_t\right)^2 - (f^*(x_t,a_t)-r_t)^2 + \log(|\mathcal{F}|T/\delta) \quad (\hat{f}_m \text{ is the ERM predictor})$$

$$\leq 2\sum_{t=\tau_{m-2}+1}^{\tau_{m-1}} \left(\widetilde{f}(\tilde{x}_t,\tilde{a}_t)-\tilde{r}_t\right)^2 - (f^*(\tilde{x}_t,\tilde{a}_t)-\tilde{r}_t)^2 + 4C + \log(|\mathcal{F}|T/\delta)$$

(By corruption definition)

$$= 4C + \log(|\mathcal{F}|T/\delta) \quad (\tilde{f}_m \text{ is the ERM predictor over uncorrupted data in the realizable setting})$$

Then, we plug this bound into the per-context regret bound in Lemma 15 to get the total regret bound as follows:

$$\sum_{t=1}^{T}\mathbb{E}_{x_t\sim\mathcal{D}_X}\left[\mathrm{Reg}(\pi_{m(t)|x_t})\right] \leq \widetilde{O}\left(\max_m \frac{\tau_m}{\gamma_m}\left(D + \sum_{s=2}^{M}\gamma_s^2\mathbb{E}_{x\sim\mathcal{D}_X}\left[\mathrm{SqErr}_{s-1}(\hat{f}_s,x)\right]\right)\right)$$

(From Lemma 15)

$$\leq \widetilde{O}\left(\frac{\tau_M}{\gamma_M}\left(D\mathrm{polylog}(\frac{1}{\varepsilon}) + \tau_{m-1}D\right)\right) \qquad (\gamma_m \text{ and } \tau_m/\gamma_m \text{ is non-decreasing in } m)$$

$$\leq \widetilde{O}\left(\sqrt{TD\left(C+\log(|\mathcal{F}|)\right)}\right) \qquad (\tau_M/\gamma_M = \widetilde{O}\big(\sqrt{T(C+\log(|\mathcal{F}|T/\delta))/D}\big))$$

■

D.3. Context-Distribution Shift

Another practical challenge in contextual bandit problems is the potential shift in the context distribution over time. Specifically, we made the following assumption:

Assumption 4 (Context-Distribution Shift) *Let context x_t be drawn from context distributions $\mathcal{D}_t$ at each time step t. There exists a constant $A \geq 1$ and a context distribution $\mathcal{D}^*$, such that for all $t \in [T]$ and $x \in \mathcal{X}$,*

$$A^{-1}\mathcal{D}^*(x) \leq \mathcal{D}_t(x) \leq A\mathcal{D}^*(x).$$

Under Assumption 4, we extend our per-context regret analysis to accommodate the changing context distributions. The key idea is to adjust the regression guarantees to account for the distribution shift, ensuring that our algorithm remains robust to these changes. We give the following corollary:

Corollary 28 (Regret Bound under Context Distribution Shift) *Under Assumptions 1 and 4, with probability at least $1-\delta$, under the doubling schedule (Theorem 16) but changing $\gamma_m = \sqrt{\frac{\tau_{m-1}AD}{\log(|\mathcal{F}|F/\delta)}}$, the total regret of Algorithm 1 satisfies*

$$\sum_{t=1}^{T} \mathbb{E}_{x_t\sim\mathcal{D}_t}\left[\mathrm{Reg}(\pi_{m(t)} \mid x_t)\right] = \widetilde{O}\left(\sqrt{A^3TD\log\left(\frac{|\mathcal{F}|\,T}{\delta}\right)}\right)$$

Proof [Proof of Corollary 28] We follow the proof of Lemma 15 but modify the application of the offline regression oracle guarantee to account for the context-distribution shift. For each epoch m, denote $\hat{f}_m$ as the ERM predictor returned by the offline regression oracle trained on the dataset collected in epoch $m-1$ over the context distribution $\mathcal{D}^*$. Denote $\widetilde{f}_m$ as the best ERM predictor trained on the dataset collected in epoch $m-1$ but under the original context distribution $\mathcal{D}_t$. Then, by the context-distribution shift definition, we have

$$\sum_{t=\tau_{m-2}+1}^{\tau_{m-1}} \mathbb{E}_{x_t\sim\mathcal{D}^*,a_t\sim\pi_m}[\mathrm{SqErr}_{m-1}(\hat{f}_m,x_t)] \le A\sum_{t=\tau_{m-2}+1}^{\tau_{m-1}} \mathbb{E}_{x_t\sim\mathcal{D}_t,a_t\sim\pi_m}[\mathrm{SqErr}_{m-1}(\widetilde{f}_m,x_t)]$$

$$\le A\log(|\mathcal{F}|T/\delta) \qquad \text{(By the offline regression oracle guarantee in the realizable setting)}$$

Therefore:

$$\sum_{t=1}^{T} \mathbb{E}_{x_t\sim\mathcal{D}^*}\left[\mathrm{Reg}(\pi_{m(t)|x_t})\right] \le \widetilde{O}\left(\max_m \frac{\tau_m}{\gamma_m}\left(D+\sum_{s=2}^{M}\gamma_s^2\mathbb{E}_{x\sim\mathcal{D}^*}\left[\mathrm{SqErr}_{s-1}(\hat{f}_s,x)\right]\right)\right)$$

(From Lemma 15)

$$\le \widetilde{O}\left(\frac{\tau_M D}{\gamma_M}+\tau_M\gamma_M\mathbb{E}_{x\sim\mathcal{D}^*}\left[\sum_{s=2}^{M}\mathrm{SqErr}_{s-1}(\hat{f}_s,x)\right]\right)$$

(γ_m and τ_m/γ_m is non-decreasing in m)

$$\le \widetilde{O}\left(\frac{\tau_M D}{\gamma_M}+A\gamma_M\log\left(\frac{|\mathcal{F}|\,T}{\delta}\right)\right)$$

$$\le \widetilde{O}\left(\sqrt{ATD\left(\frac{|\mathcal{F}|\,T}{\delta}\right)}\right)$$

Then, following the above derivation, we have

$$\sum_{t=1}^{T} \mathbb{E}_{x_t\sim\mathcal{D}_t}\left[\mathrm{Reg}(\pi_{m(t)|x_t})\right] \le A\sum_{t=1}^{T} \mathbb{E}_{x_t\sim\mathcal{D}^*}\left[\mathrm{Reg}(\pi_{m(t)|x_t})\right] \le \widetilde{O}\left(\sqrt{A^3TD\left(\frac{|\mathcal{F}|\,T}{\delta}\right)}\right)$$

■

D.4. Context-dependent Benchmark Distribution Space

For some applications, each context x may be associated with its own benchmark space of distributions. For example, in LLM alignment applications (Ouyang et al., 2022), we have a base policy π_{base} and it

is suitable to define $\Lambda_x^h = \left\{\lambda : \frac{d\lambda(\cdot)}{d\pi_{\text{base}}(\cdot|x)} \leq \frac{1}{h}\right\}$. Our per-context regret analysis can be easily adapted to handle this context-dependent benchmark space setting. The idea is to modify our per-context regret result to account for the context-dependent benchmark space. We only need to replace the per-context lemma 15 with the following lemma:

Lemma 29 *For any $\delta \in (0,1)$, $\Lambda \subseteq \Delta(\mathcal{A})$, with probability at least $1-\delta$, given accessibility to any offline regression oracle, the Λ-Regret of Algorithm 1 for every context x is bounded as*

$$\sum_{t=1}^{T} \mathrm{Reg}(\pi_{m(t)} \mid x)$$
$$\leq O\left(\tau_1 + \max_{m\in\{2,\dots,M\}} \frac{\tau_m}{\gamma_m} \cdot \left(\sum_{s=1}^{M} \gamma_s \mathsf{doec}_{\gamma_s,\varepsilon_s}(\mathcal{F}_x, \Lambda_x) + \sum_{s=2}^{M} \gamma_s^2 \left(\mathrm{SqErr}_{s-1}(\hat{f}_s, x) + \varepsilon_{s-1}\right)\right)\right),$$

The proof will be identical to that of Lemma 15, with the only difference being that we replace the use of the fixed benchmark space Λ with the context-dependent benchmark space Λ_x throughout the proof. Therefore, following similar steps as in Appendix B, we derive the total regret bound under the context-dependent benchmark space setting of OE2D:

$$\mathbb{E}\left[\mathrm{Regret}(T, \mathrm{OE2D})\right]$$
$$\leq \widetilde{O}\left(\tau_1 + \max_{m\in\{2,\dots,M\}} \frac{\tau_m}{\gamma_m} \cdot \left(\begin{array}{l} \max_{m\in[M]} \gamma_m \mathbb{E}\left[\mathsf{doec}_{\gamma_m,\varepsilon_m}(\mathcal{F}_x, \Lambda_x)\right] \\ \qquad + \max_{m\in\{2,\dots,M\}} \gamma_m^2 \left(\mathrm{Reg}_{\mathrm{off}}(\mathcal{F}, \tau_{m-1}/2, \delta_m) + \varepsilon_{m-1}\right) \end{array}\right)\right).$$

Appendix E. Proofs from Section 4

In this section, we prove Theorem 7: for any relaxed coverage satisfying Assumption 5 with step-size threshold $\bar{\Delta}$, Algorithm 2 terminates within $\lfloor 1/\bar{\Delta} \rfloor$ iterations, and its output distribution p^* certifies Eq. (17):

$$\overline{\mathsf{doec}}_{\gamma,\varepsilon}(\hat{g}, \mathcal{G}, \Lambda) \leq \frac{10}{\gamma} \overline{\mathsf{SEC}}_{\varepsilon}(\mathcal{G}, \Lambda) \tag{17}$$

This section is organized as follows. We first state the assumption that an admissible relaxed coverage should satisfy, then prove Theorem 7 via a potential-function argument, relying on two properties of the potential: a constant per-iteration decrease, proved in Appendix E.1.1 using the dilution-stability of Assumption 5, and a uniform lower bound, proved in Appendix E.1.2 via the weighted sequential extrapolation bound Eq. (23), which follows from Lemma 31. Appendices E.2.1 and E.2.2 bound the $\overline{\mathsf{SEC}}_{\varepsilon}$ in the finite Eluder dimension with trivial relaxation (Proposition 34) and under the three running examples (Lemma 36); Appendix E.3 verifies the admissible assumption for the trivial relaxation (Lemma 37) and for the relaxed coverages with cushion parameter ε of the running examples (Lemma 38); Appendix E.4 shows that a more aggressive step size yields faster termination when $\Lambda = \{\delta_a : a \in \mathcal{A}\}$ (Proposition 40); and Appendix E.6 proves Proposition 8, showing that the DOEC can be far smaller than the SEC. Throughout, we write $\mathcal{DG} := \{g - g' : g, g' \in \mathcal{G}\}$.

E.0.1. ADMISSIBLE RELAXED COVERAGE

We state the admissibility assumption required of the relaxed coverage $\overline{\mathsf{Coverage}}_\varepsilon$, distilled from the convergence analysis of Algorithm 2 below.

Assumption 5 (Admissible relaxed coverage) *The relaxed coverage $\overline{\mathsf{Coverage}}_\varepsilon$ (extended to take an unnormalized nonnegative measure as its first argument) satisfies the following four properties for every $\lambda \in \Lambda$ and $\varepsilon > 0$:*

(i) ***(Monotonicity)*** *$\overline{\mathsf{Coverage}}_\varepsilon(p, \lambda; \mathcal{G})$ is non-increasing in ε, and if $p \succeq q$, $\overline{\mathsf{Coverage}}_\varepsilon(p, \lambda; \mathcal{G}) \leq \overline{\mathsf{Coverage}}_\varepsilon(q, \lambda; \mathcal{G})$;*

(ii) ***(Homogeneity)*** *$\overline{\mathsf{Coverage}}_{c\varepsilon}(c\,p, \lambda; \mathcal{G}) = \frac{1}{c}\,\overline{\mathsf{Coverage}}_\varepsilon(p, \lambda; \mathcal{G})$;*

(iii) ***(Continuity)*** *$\overline{\mathsf{Coverage}}_\varepsilon(p, \lambda; \mathcal{G})$ is continuous in the covering measure p with respect to the total variation distance;*

(iv) ***(Dilution stability)*** *there exists a step-size threshold $\bar{\Delta} \in (0, 1]$, for all nonnegative measures p and λ, such that $\overline{\mathsf{Coverage}}_\varepsilon(p + \bar{\Delta}\lambda, \lambda; \mathcal{G}) \geq \frac{1}{2}\,\overline{\mathsf{Coverage}}_\varepsilon(p, \lambda; \mathcal{G})$.*

All four properties describe the behavior of the relaxed coverage: Monotonicity (i), homogeneity (ii), and continuity (iii) require the relaxed coverage to be well-behaved in the covering measure p and the cushion parameter ε. Dilution stability (iv) controls how fast the coverage of a direction λ can decay as mass is added to the covering measure. By monotonicity (i), augmenting p with mass $\bar{\Delta}\lambda$ only *dilutes* (i.e., decreases) the coverage $\overline{\mathsf{Coverage}}_\varepsilon(p, \lambda; \mathcal{G})$ of that same direction; the property requires this dilution to be *stable*, in the sense that one step of size $\bar{\Delta}$ shrinks the coverage by at most a factor of 2. This is exactly the per-step guarantee that the coordinate descent step of Algorithm 2 consumes when it adds mass $\Delta_t\lambda_t$ ($\Delta_t \leq \bar{\Delta}$) along a newly selected direction λ_t.

E.1. Proof of Theorem 7

Theorem 7 *For any reward function class $\mathcal{G} : \mathcal{A} \to [0, 1]$, benchmark distribution class Λ, relaxed coverage $\overline{\mathsf{Coverage}}$ satisfying Assumption 5 with step-size threshold $\bar{\Delta}$, $\gamma > 0$, and $\varepsilon \in (0, 1)$, Algorithm 2 with step size $\Delta_t = \bar{\Delta}$ terminates within $\lfloor 1/\bar{\Delta} \rfloor$ iterations and outputs a distribution $p^* \in \mathrm{co}(\Lambda)$ such that*

$$\max_{\lambda \in \Lambda} \left(\mathbb{E}_{a \sim \lambda}\left[\hat{g}(a)\right] - \mathbb{E}_{a \sim p^*}\left[\hat{g}(a)\right] + \tfrac{1}{\gamma}\overline{\mathsf{Coverage}}_\varepsilon(p^*, \lambda; \mathcal{G}) \right) \leq \tfrac{10}{\gamma}\overline{\mathsf{SEC}}_\varepsilon(\mathcal{G}, \Lambda). \tag{6}$$

Proof [Proof of Theorem 7] The theorem statement consists of two parts: the termination of Algorithm 2 and p^* certifies Eq. (17).

Termination of Algorithm 2: We will use the following potential function in our analysis:

$$\Phi_t = \underbrace{-\sum_{s=1}^{t} \Delta_s \overline{\mathsf{Coverage}}_\varepsilon(p_s, \lambda_s; \mathcal{G})}_{F_t} + \underbrace{\gamma \sum_{s=1}^{t} \frac{1}{2}\Delta_s \widehat{R}(\lambda_s)}_{G_t \geq 0}$$

We give three essential properties of the potential function Φ_t that help us analyze the termination. In the proof below, we also extend the definition of $\widehat{R}$ to any nonnegative measures that are not necessarily normalized. Specifically,

$$\widehat{R}(p) = \|p\|_1 \max_{\lambda' \in \Lambda} \mathbb{E}_{a \sim \lambda'}[\hat{g}(a)] - \sum_{a \in \mathcal{A}} p(a)\hat{g}(a). \tag{18}$$

With this notation, $G_t = \frac{1}{2}\widehat{R}(p_t)$, since $p_t = \sum_{s=1}^{t} \Delta_s \lambda_s$. We also recall that Algorithm 2 uses an extended definition of relaxed coverage that does not require the covering measure p_s to be normalized.

1. Zero-Initialization property: $\Phi_0 = 0$. This holds since $F_0 = 0$ and $G_0 = 0$ by definition.
2. Constant-Decreasing (Proposition 30): for every coordinate descent step from p_{t-1} to p_t in Algorithm 2, since $\Delta_t = \bar{\Delta}$, we have
$$\Phi_{t-1} - \Phi_t \geq 4\bar{\Delta} \cdot \overline{\mathsf{SEC}}_\varepsilon(\mathcal{G}, \Lambda).$$
3. Lower-Boundedness (Proposition 32): for every $t > 0$,
$$\Phi_t \geq -F_t \geq -\overline{\mathsf{SEC}}_{\varepsilon/||p_t||_1}(\mathcal{G}, \Lambda).$$

We now take these properties as given and defer the proofs of Propositions 30 and 32 to Sections E.1.1 and E.1.2, respectively.

We now claim that the end of iteration $t = \lfloor 1/\bar{\Delta} \rfloor$ is never reached. Otherwise, at that iteration, $||p_t||_1 = \bar{\Delta} t = \bar{\Delta} \cdot \lfloor 1/\bar{\Delta} \rfloor \leq 1$, and thus $\overline{\mathsf{SEC}}_{\varepsilon/||p_t||_1}(\mathcal{G}, \Lambda) \leq \overline{\mathsf{SEC}}_\varepsilon(\mathcal{G}, \Lambda)$ since $\varepsilon \mapsto \overline{\mathsf{SEC}}_\varepsilon(\mathcal{G}, \Lambda)$ is monotonically decreasing by the monotonicity property (i) of Assumption 5. Therefore, according to the Lower-Boundedness property, we have

$$\Phi_t \geq -\overline{\mathsf{SEC}}_{\varepsilon/||p_t||_1}(\mathcal{G}, \Lambda) \geq -\overline{\mathsf{SEC}}_\varepsilon(\mathcal{G}, \Lambda).$$

On the other hand, since $\bar{\Delta} \leq 1$, we have $\lfloor 1/\bar{\Delta} \rfloor \geq \frac{1}{2\bar{\Delta}}$. According to the Constant-Decreasing property, we have

$$\Phi_t \leq -4\overline{\mathsf{SEC}}_\varepsilon(\mathcal{G}, \Lambda) \cdot \bar{\Delta} t < -2\overline{\mathsf{SEC}}_\varepsilon(\mathcal{G}, \Lambda).$$

Thus, we reach a contradiction. Therefore, Algorithm 2 terminates in $t_0 \leq \lfloor 1/\bar{\Delta} \rfloor$ iterations, and $\|p_{t_0}\|_1 \leq t_0 \bar{\Delta} \leq 1$. In addition, p_{t_0} is a nonnegative combination of λ_t, which are elements of Λ. Therefore,

$$p^* = p_{t_0} + (1 - \|p_{t_0}\|)\widehat{\lambda}$$

is also a nonnegative combination of elements of Λ, and since $\|p^*\|_1 = 1$, and all elements in Λ are valid probability distributions, the combination is also a convex combination, and thus $p^* \in \mathrm{co}(\Lambda)$.

We additionally show that the output distribution p^* satisfies the Low Regret (LR) property. Indeed, since p^* is a mixture of p_{t_0} and $\widehat{\lambda}$, we have

$$\widehat{R}(p^*) = \widehat{R}(p_{t_0}) = \frac{2}{\gamma}(F_{t_0} + \Phi_{t_0}) \leq \frac{2}{\gamma} F_{t_0} \leq \frac{2}{\gamma}\overline{\mathsf{SEC}}_\varepsilon(\mathcal{G}, \Lambda)$$

(By the lower Boundedness Property)

where the first equality is because $\widehat{R}(\widehat{\lambda}) = 0$ and we use the extended definition of $\widehat{R}$ (Eq. (18)), the second equality is by the definition of Φ_{t_0}, and the first inequality is by the fact that $\Phi_{t_0} \leq 0$ by the Constant-Decreasing property. The second inequality is by the Lower-Boundedness property.

Next, we show that the output distribution p^* satisfies Eq. (6). Since the algorithm terminates at iteration t_0, p_{t_0} satisfies that $\max_{\lambda\in\Lambda}\left(\frac{1}{\gamma}\overline{\mathsf{Coverage}}_\varepsilon(p_{t_0},\lambda) - \widehat{R}(\lambda)\right) \leq \frac{8}{\gamma}\overline{\mathsf{SEC}}_\varepsilon(\mathcal{G},\Lambda)$; since $p^* \succeq p_{t_0}$, we also have $\overline{\mathsf{Coverage}}_\varepsilon(p^*,\lambda) \leq \overline{\mathsf{Coverage}}_\varepsilon(p_{t_0},\lambda)$ by the monotonicity property of Assumption 5, and thus,

$$\max_{\lambda\in\Lambda}\left(\frac{1}{\gamma}\overline{\mathsf{Coverage}}_\varepsilon(p^*,\lambda) - \widehat{R}(\lambda)\right) \leq \frac{8}{\gamma}\overline{\mathsf{SEC}}_\varepsilon(\mathcal{G},\Lambda). \tag{19}$$

Combining this with the fact that $\widehat{R}(p^*) \leq \frac{2}{\gamma}\overline{\mathsf{SEC}}_\varepsilon(\mathcal{G},\Lambda)$, we have

$$\widehat{R}(p^*) + \max_{\lambda\in\Lambda}\left(\frac{1}{\gamma}\overline{\mathsf{Coverage}}_\varepsilon(p^*,\lambda) - \widehat{R}(\lambda)\right) \leq \frac{10}{\gamma}\overline{\mathsf{SEC}}_\varepsilon(\mathcal{G},\Lambda),$$

i.e., p^* certifies that $\overline{\mathsf{doec}}_{\gamma,\varepsilon}(\hat{g},\mathcal{G},\Lambda) \leq \frac{10}{\gamma}\overline{\mathsf{SEC}}_\varepsilon(\mathcal{G},\Lambda)$. ■

E.1.1. PROPERTY OF POTENTIAL FUNCTION: CONSTANT DECREASING

Proposition 30 (Constant-Decreasing Proposition of Potential Function) *Suppose the relaxed coverage $\overline{\mathsf{Coverage}}$ satisfies Assumption 5 with step-size threshold $\bar{\Delta}$. For each iteration t in Algorithm 2 such that the condition in line 5 is satisfied, if $\Delta_t \leq \bar{\Delta}$, we have*

$$\Phi_{t-1} - \Phi_t \geq 4\Delta_t \cdot \overline{\mathsf{SEC}}_\varepsilon(\mathcal{G},\Lambda)$$

Proof [Proof of Proposition 30] We first examine $\Phi_{t-1} - \Phi_t$, which by its definition, is equal to

$$\Phi_{t-1} - \Phi_t = \Delta_t\overline{\mathsf{Coverage}}_\varepsilon(p_t,\lambda_t) - \frac{\gamma}{2}\Delta_t\widehat{R}(\lambda_t) \tag{20}$$

Since at every iteration t, the condition in line 5 is satisfied, we also have

$$\overline{\mathsf{Coverage}}_\varepsilon(p_{t-1},\lambda_t) \geq \gamma\widehat{R}(\lambda_t) + 8\overline{\mathsf{SEC}}_\varepsilon(\mathcal{G},\Lambda) \tag{21}$$

Although this does not directly help with lower bounding Eq. (20), we now use that with $\Delta_t \geq \bar{\Delta}$, by the dilution stability of Assumption 5, $\overline{\mathsf{Coverage}}_\varepsilon(p_t,\lambda_t) \geq \frac{1}{2}\overline{\mathsf{Coverage}}_\varepsilon(p_{t-1},\lambda_t)$.

Therefore, Eq. (21) implies that

$$\overline{\mathsf{Coverage}}_\varepsilon(p_t,\lambda_t) \geq 4\overline{\mathsf{SEC}}_\varepsilon(\mathcal{G},\Lambda) + \frac{\gamma}{2}\widehat{R}(\lambda_t) \tag{22}$$

Plugging this back to Eq. (20), we have

$$\Phi_{t-1} - \Phi_t \geq \Delta_t\left(\frac{\gamma}{2}\widehat{R}(\lambda_t) + 4\overline{\mathsf{SEC}}_\varepsilon(\mathcal{G},\Lambda) - \frac{\gamma}{2}\widehat{R}(\lambda_t)\right) = 4\Delta_t\overline{\mathsf{SEC}}_\varepsilon(\mathcal{G},\Lambda).$$

■

E.1.2. PROPERTY OF POTENTIAL FUNCTION: LOWER BOUNDEDNESS

We establish a *weighted sequential extrapolation bound* (Lemma 31 below) for any relaxed coverage satisfying the monotonicity, homogeneity, and continuity properties (i)–(iii) of Assumption 5: monotonicity gives the bound for integer weights via a "repetition" argument, and homogeneity and continuity extend it to real weights via a limiting argument. Proposition 32 then follows from the real-weighted sequential extrapolation lemma (Lemma 31) by the definition of the potential function Φ_t.

Lemma 31 (Weighted relaxed SEC bound) *Suppose the relaxed coverage* $\overline{\mathsf{Coverage}}$ *satisfies admissibility (Assumption 5). Then the following* weighted sequential extrapolation bound *holds: for every* $N \in \mathbb{N}$, $\lambda_1, \dots, \lambda_N \in \Lambda$, *and weights* $m_1, \dots, m_N \geq 0$ *with* $M := \sum_{i=1}^N m_i > 0$, *and every cushion parameter* $\delta > 0$,

$$\sum_{i=1}^{N} m_i \, \overline{\mathsf{Coverage}}_{M\delta}\left(\textstyle\sum_{j=1}^{i} m_j\lambda_j,\ \lambda_i;\ \mathcal{G}\right) \leq \overline{\mathsf{SEC}}_{\delta}(\mathcal{G}, \Lambda). \tag{23}$$

Proof We first prove the claim for integer weights $m_1, \dots, m_N \in \mathbb{N}$. Consider the length-$M$ sequence of elements of Λ that repeats λ_1 for m_1 times, then λ_2 for m_2 times, and so on. By monotonicity of admissible relaxed coverage (property (i)), for every $i \in [N]$ and $k \in [m_i]$,

$$\overline{\mathsf{Coverage}}_{M\delta}\left(\textstyle\sum_{j=1}^{i} m_j\lambda_j,\ \lambda_i;\ \mathcal{G}\right) \leq \overline{\mathsf{Coverage}}_{M\delta}\left(\textstyle\sum_{j=1}^{i-1} m_j\lambda_j + k\lambda_i,\ \lambda_i;\ \mathcal{G}\right),$$

since $\sum_{j=1}^{i-1} m_j\lambda_j + k\lambda_i \preceq \sum_{j=1}^{i} m_j\lambda_j$. Summing the inequality over $k \in [m_i]$ and $i \in [N]$ gives

$$\sum_{i=1}^{N} m_i \, \overline{\mathsf{Coverage}}_{M\delta}\Big(\sum_{j=1}^{i} m_j\lambda_j, \lambda_i; \mathcal{G}\Big) \leq \overline{\mathsf{SEC}}_{\delta}(\mathcal{G}, \Lambda)$$

evaluated on the repeated sequence; the integer-weight claim follows.

For real weights $m_1, \dots, m_N \geq 0$, fix $Z > 0$ and apply the integer-weight claim to the weights $w_i := \lfloor Z m_i \rfloor$ with $W := \sum_{i=1}^N w_i$:

$$\sum_{i=1}^{N} w_i \, \overline{\mathsf{Coverage}}_{W\delta}\left(\textstyle\sum_{j=1}^{i} w_j\lambda_j,\ \lambda_i;\ \mathcal{G}\right) \leq \overline{\mathsf{SEC}}_{\delta}(\mathcal{G}, \Lambda).$$

Since $W \leq ZM$ and $\overline{\mathsf{Coverage}}$ is non-increasing in the cushion parameter (Monotonicity), the inequality continues to hold with $W\delta$ replaced by $ZM\delta$. By homogeneity with $c = Z$,

$$\begin{aligned}\sum_{i=1}^{N} w_i \, \overline{\mathsf{Coverage}}_{ZM\delta}\left(\textstyle\sum_{j=1}^{i} w_j\lambda_j,\ \lambda_i;\ \mathcal{G}\right) &= \sum_{i=1}^{N} \frac{w_i}{Z} \, \overline{\mathsf{Coverage}}_{M\delta}\left(\textstyle\sum_{j=1}^{i} \frac{w_j}{Z}\lambda_j,\ \lambda_i;\ \mathcal{G}\right) \\ &\leq \overline{\mathsf{SEC}}_{\delta}(\mathcal{G}, \Lambda).\end{aligned}$$

Letting $Z \to \infty$, so that $\frac{w_i}{Z} \to m_i$ for every i and hence $\sum_{j=1}^{i} \frac{w_j}{Z}\lambda_j \to \sum_{j=1}^{i} m_j\lambda_j$ in total variation, and using continuity of the coverage in its first argument (property (iii) of Assumption 5), the left-hand side converges to $\sum_{i=1}^{N} m_i \, \overline{\mathsf{Coverage}}_{M\delta}\left(\sum_{j=1}^{i} m_j\lambda_j, \lambda_i; \mathcal{G}\right)$, which proves the claim. ■

Proposition 32 (Lower Boundedness Property of Potential Function) *For every iteration $t > 0$ of Algorithm 2, the following inequality holds:*

$$\Phi_t \geq -F_t \geq -\overline{\mathsf{SEC}}_{\varepsilon/||p_t||_1}(\mathcal{G}, \Lambda). \tag{24}$$

Proof [Proof of Proposition 32] Starting from the definition of Φ_t, we have $\Phi_t = -F_t + \gamma G_t \geq -F_t$ since $G_t \geq 0$ by definition. Therefore, it suffices to show that $F_t \leq \overline{\mathsf{SEC}}_{\varepsilon/||p_t||_1}(\mathcal{G}, \Lambda)$. To this end, we use the properties of the relaxed coverage. Since $p_s = \sum_{j=1}^{s} \Delta_j \lambda_j$ for every $s \leq t$, the sum $F_t = \sum_{s=1}^{t} \Delta_s \, \overline{\mathsf{Coverage}}_\varepsilon(p_s, \lambda_s; \mathcal{G})$ is a real-weighted sequential extrapolation sum with weights $m_s = \Delta_s$ and total weight $M = \sum_{s=1}^{t} \Delta_s = ||p_t||_1$; by the admissibility of $\overline{\mathsf{Coverage}}$ (Assumption 5), Lemma 31 above asserts that it obeys the weighted sequential extrapolation bound Eq. (23). Applying Eq. (23) by setting the cushion parameter δ therein to be $\varepsilon/||p_t||_1$ (so that its cushion parameter $M\delta = M \cdot \frac{\varepsilon}{||p_t||_1}$ equals the cushion ε of the coverage terms in F_t), we have $F_t \leq \overline{\mathsf{SEC}}_{\varepsilon/||p_t||_1}(\mathcal{G}, \Lambda)$. ■

E.2. Relating $\overline{\mathsf{SEC}}_\varepsilon$ to Other Complexity Measures

In this section, we prove upper bounds on the relaxed ε-SEC for our settings of interest: the trivial relaxation under finite Eluder dimension (Proposition 34), and the three running examples: discrete action space, per-context (generalized) linear reward, and h-smoothed regret, under their cushioned relaxed coverages (Lemma 36). These bounds, together with the guarantee of Algorithm 2 in Theorem 7, lead to regret guarantees of OE2D under these settings.

E.2.1. CASE 1: SEC_ε IN FINITE ELUDER DIMENSION CASE

Recall that, for a function class that has a finite Eluder dimension and Λ is a set of Dirac measures over the action space, we show that the original ε-SEC is bounded. We first recall the definition of the Eluder dimension:

Definition 33 (Eluder Dimension (Russo and Van Roy, 2013)) *Given a function class $\mathcal{G}$ mapping from a domain $\mathcal{Z}$ to $\mathbb{R}$, a point $z \in \mathcal{Z}$ is said to be ε-independent of a set of points $\{z_1, z_2, \ldots, z_n\} \subseteq \mathcal{Z}$ with respect to $\mathcal{G}$ if there exist two functions $f, f' \in \mathcal{G}$ such that $\sqrt{\sum_{i=1}^{n}(f(z_i) - f'(z_i))^2} \leq \varepsilon$ and $|f(z) - f'(z)| > \varepsilon$. The eluder dimension $\mathrm{Edim}(\mathcal{G}, \varepsilon)$ is defined as the length of the longest sequence of points $z_1, z_2, \ldots, z_m \in \mathcal{Z}$ such that each point z_i is ε'-independent of its predecessors $\{z_1, z_2, \ldots, z_{i-1}\}$ with respect to $\mathcal{G}$ for some $\varepsilon' \geq \varepsilon$.*

Next, we are going to prove Proposition 34, which states that $\mathsf{SEC}_\varepsilon(\mathcal{G}, \Lambda)$ is upper bounded by the eluder dimension upto some logarithmic factors. We note that this property was already given by Agarwal et al. (2024, Lemma E.3). However, their proof contains a small typo that relies on an incorrect lemma (Agarwal et al., 2024, Lemma E.2) in their peeling argument. Specifically, we need to improve their $\log T$ factors therein to $\log(1/\varepsilon)$. Hence, we provide a corrected proof with a peeling argument and highlight the difference between the proof of Lemma 35 and Agarwal et al. (2024, Lemma E.2). The readers are welcome to refer to Agarwal et al. (2024, Lemma E.2 and E.3) for more context.

Proposition 34 *For* $\Lambda = \{\delta_a : a \in \mathcal{A}\}$, $\mathsf{SEC}_\varepsilon(\mathcal{G}, \Lambda) \lesssim \mathrm{Edim}(\mathcal{G}, \sqrt{\varepsilon}) \log^2(\frac{1}{\varepsilon})$, *and thus*

$$\mathsf{doec}_{\gamma,\varepsilon}(\mathcal{G}, \Lambda) \lesssim \frac{1}{\gamma} \mathrm{Edim}(\mathcal{G}, \sqrt{\varepsilon}) \log^2(\frac{1}{\varepsilon}),$$

where Edim *denotes the Eluder dimension.*

Aside from our running Example 2, Proposition 34 enables provable regret guarantee of OE2D under the per-context generalized linear reward model (Xu and Zeevi, 2020, Example 2). In this setting, for each x, $\mathrm{Edim}(\mathcal{F}_x, \sqrt{\varepsilon}) \le \widetilde{O}\big(d \log \frac{1}{\varepsilon}\big)$ (Russo and Van Roy, 2013), and $\mathsf{doec}_{\gamma,\varepsilon}(\mathcal{F}_x, \Lambda) \le \frac{d}{\gamma} \log^3(\frac{1}{\varepsilon})$, and thus OE2D has a regret bound of $\widetilde{O}\big(\sqrt{dT \log |\mathcal{F}|}\big)$, which matches Xu and Zeevi (2020) and improves the number of offline regression oracle calls significantly, from $O(T)$ to $O(\log(T))$. We also note that small Eluder dimension can capture reward classes beyond generalized linear models (Li et al., 2022).

Proof [Proof of Proposition 34] For any sequence $\{\lambda_i = \delta_{a_i}, h_i\}_{i=1}^N$ such that $h_i \in \mathcal{DG}$, we will split the interval $(0, 1]$ into multiple small intervals with length increasing exponentially. Specifically, we split $(0, 1]$ into disjoint intervals $(2^{k-1}\sqrt{\varepsilon}, 2^k\sqrt{\varepsilon}]$ for $k = 1, 2, \ldots, K$, where $K = \lceil \log_2(1/\sqrt{\varepsilon}) \rceil$. Then, we can bound the weighted smooth Eluder summation as

$$\begin{aligned}
\sum_{i=1}^N \frac{h_i(a_i)^2}{N\varepsilon + \sum_{j=1}^i h_i(a_j)^2} &= \sum_{k=1}^K \sum_{\substack{i \in [1,N]: \\ h_i(a_i) \in (2^{k-1}\sqrt{\varepsilon}, 2^k\sqrt{\varepsilon}]}} \frac{h_i(a_i)^2}{N\varepsilon + \sum_{j=1}^i h_i(a_j)^2} \\
&\le \sum_{k=1}^K 4\mathrm{Edim}(\mathcal{G}, 2^{k-1}\sqrt{\varepsilon}) \log\left(1 + \frac{(2^{k-1}\sqrt{\varepsilon})^2}{\varepsilon}\right) \\
&\qquad \text{(apply Corrected Lemma 35 to each small interval)} \\
&\le 4\mathrm{Edim}(\mathcal{G}, \sqrt{\varepsilon}) \sum_{k=1}^K \log\left(1 + 4^{k-1}\right) \\
&\qquad (\mathrm{Edim}(\mathcal{G}, 2^{k-1}\sqrt{\varepsilon}) \le \mathrm{Edim}(\mathcal{G}, \sqrt{\varepsilon}) \text{ for all } k) \\
&\le 4\mathrm{Edim}(\mathcal{G}, \sqrt{\varepsilon}) K \log\left(\frac{1}{K} \sum_{k=1}^K \left(1 + 4^{k-1}\right)\right) \quad \text{(Jensen's inequality)} \\
&\le 16\mathrm{Edim}(\mathcal{G}, \sqrt{\varepsilon}) K^2
\end{aligned}$$

Since the above inequality holds for every $N \in \mathbb{N}$, $h_1, \ldots, h_N \in \mathcal{DG}$ and $\lambda_1, \ldots, \lambda_N \in \Lambda$ and every decomposition of q into weighted distributions in Λ, we take supremum over them to have Proposition 34 holds. ■

Lemma 35 (Bounding ε-SEC in Finite Eluder Dimension Case with Peeling Argument) *For any function class* $\mathcal{G} : \mathcal{A} \to \mathbb{R}$, $\varepsilon > 0$, *and any sequence* $\{a_i, h_i\}_{i=1}^N$ *such that* $h_i \in \mathcal{DG}$ *and* $h_i(a) \in (\theta, 2\theta]$ *for some* $\theta > 0$. *Then, we have the following inequality holds:*

$$\sum_{i=1}^N \frac{h_i(a_i)^2}{N\varepsilon + \sum_{j=1}^i h_i(a_j)^2} \le \mathrm{Edim}(\mathcal{G}, \sqrt{\varepsilon}) \log\left(1 + \frac{\theta^2}{\varepsilon}\right) \tag{25}$$

Proof [Proof of Lemma 35] The proof is similar to the proof of Agarwal et al. (2024, Lemma E.2).

$$\begin{aligned}
&\sum_{i=1}^{N} \frac{h_i(a_i)^2}{N\varepsilon + \sum_{j=1}^{i} h_i(a_j)^2} = \sum_{i=1}^{N} \frac{h_i(a_i)^2}{N\varepsilon + \sum_{j=1}^{i-1} h_i(a_j)^2 + h_i(a_i)^2} \\
&\le 4\mathrm{Edim}(\mathcal{G}, \sqrt{\varepsilon}) \sum_{n=1}^{N/\mathrm{Edim}(\mathcal{G},\sqrt{\varepsilon})} \frac{\theta^2}{N\varepsilon + n\theta^2} \\
&\le 4\mathrm{Edim}(\mathcal{G}, \sqrt{\varepsilon}) \int_0^{N/\mathrm{Edim}(\mathcal{G},\sqrt{\varepsilon})} \frac{1}{N\varepsilon/\theta^2 + x}\,\mathrm{d}x \\
&= 4\mathrm{Edim}(\mathcal{G}, \sqrt{\varepsilon}) \left(\log\left(\frac{N\varepsilon}{\theta^2} + \frac{N}{\mathrm{Edim}(\mathcal{G}, \sqrt{\varepsilon})} \right) - \log\left(\frac{N\varepsilon}{\theta^2} \right) \right) \\
&\le 4\mathrm{Edim}(\mathcal{G}, \sqrt{\varepsilon}) \log\left(1 + \frac{\theta^2}{\varepsilon} \right) \qquad (\mathrm{Edim}(\mathcal{G}, \varepsilon) \text{ is at least } 1)
\end{aligned}$$

The first inequality holds by filling the summation with the maximum possible allocations in the constructed buckets. The detailed argument is the same as the proof of Agarwal et al. (2024, Lemma E.2). For each bucket, the maximum number of elements is the eluder dimension $\mathrm{Edim}(\mathcal{G}, \varepsilon)$ due to the construction of buckets. The second inequality holds because the summation can be approximated by an integral. The third inequality holds by relaxing Edim in the denominator to 1. ■

E.2.2. CASE 2: $\overline{\mathsf{SEC}}_\varepsilon$ IN THE RUNNING EXAMPLES

In this section, we bound the relaxed SEC $\overline{\mathsf{SEC}}_\varepsilon(\mathcal{G}, \Lambda)$ in the three running examples, using the cushioned relaxed coverages of Section 3; these are the values quoted in Section 4. Throughout, $\Sigma_p := \sum_{a\in\mathcal{A}} p(a)\phi(a)\phi(a)^\top$, and in the h-smoothed setting $\lambda(a), p(a)$ denote densities with respect to a base measure $\mu \in \Delta(\mathcal{A})$, where $\Delta_h^\mu(\mathcal{A}) = \left\{ \lambda \in \Delta(\mathcal{A}) : \frac{\mathrm{d}\lambda}{\mathrm{d}\mu}(a) \le \frac{1}{h}\ \forall a \in \mathcal{A} \right\}$.

Lemma 36 (Relaxed SEC bounds for the running examples) *With the cushioned relaxed coverages below, $\overline{\mathsf{SEC}}_\varepsilon(\mathcal{G}, \Lambda)$ is bounded as follows:*

1. ***Discrete action space*** *($\mathcal{G} \subseteq [0,1]^{\mathcal{A}}$, $\Lambda = \{\delta_a : a \in \mathcal{A}\}$), with $\overline{\mathsf{Coverage}}_\varepsilon(p, \lambda; \mathcal{G}) = \sum_{a\in\mathcal{A}} \frac{\lambda(a)}{p(a)+\varepsilon/|\mathcal{A}|}$:*

$$\overline{\mathsf{SEC}}_\varepsilon(\mathcal{G}, \Lambda) \le |\mathcal{A}| \log\left(1 + \frac{|\mathcal{A}|}{\varepsilon} \right);$$

2. ***Per-context (generalized) linear reward*** *($\mathcal{G} = \left\{ a \mapsto \sigma(\phi(a)^\top\theta) \right\}$, link σ with $\kappa := \overline{L}/\underline{L}$ and parameter diameter B), with $\varepsilon' := \frac{\varepsilon}{\underline{L}^2 B^2}$ and $\overline{\mathsf{Coverage}}_\varepsilon(p, \lambda; \mathcal{G}) = \kappa^2 \operatorname{tr}\left((\Sigma_p + \varepsilon' I)^{-1} \Sigma_\lambda \right)$:*

$$\overline{\mathsf{SEC}}_\varepsilon(\mathcal{G}, \Lambda) \le \kappa^2 d \log\left(1 + \frac{1}{\varepsilon'} \right)$$

(taking $\sigma = \mathrm{id}$, so $\kappa = 1$, covers the per-context linear reward);

3. ***h*-smoothed regret** ($\Lambda = \Delta_h^\mu(\mathcal{A})$), *with* $\overline{\mathsf{Coverage}}_\varepsilon(p, \lambda; \mathcal{G}) = \frac{1}{h}\mathbb{E}_{a\sim\mu}\left[\frac{\lambda(a)}{p(a)+\varepsilon}\right]$:

$$\overline{\mathsf{SEC}}_\varepsilon(\mathcal{G}, \Lambda) \le \frac{1}{h}\log\left(1 + \frac{1}{h\varepsilon}\right).$$

Proof Fix $N \in \mathbb{N}$ and $\lambda_1, \dots, \lambda_N \in \Lambda$, and write $\lambda_{1:i} := \sum_{j=1}^{i} \lambda_j$. Recall that, in the relaxed ε-SEC (Definition 6), the cushion parameter of a length-N sequence is $N\varepsilon$. We repeatedly use the scalar elliptical potential inequality (the one-dimensional case of the elliptical potential lemma (see, e.g., Abbasi-Yadkori et al., 2011; Lattimore and Szepesvári, 2020))

$$\sum_{i=1}^{N} \frac{a_i}{\varepsilon_0 + a_{1:i}} \le \log\frac{\varepsilon_0 + a_{1:N}}{\varepsilon_0} \qquad \text{for } a_1, \dots, a_N \ge 0,\ \varepsilon_0 > 0, \tag{26}$$

which follows by summing $\frac{a_i}{\varepsilon_0+a_{1:i}} = \frac{x_i}{1+x_i} \le \log(1+x_i) = \log\frac{\varepsilon_0+a_{1:i}}{\varepsilon_0+a_{1:i-1}}$ with $x_i := \frac{a_i}{\varepsilon_0+a_{1:i-1}}$.

Discrete: applying Eq. (26) per action with $a_i = \lambda_i(a)$ and $\varepsilon_0 = \frac{N\varepsilon}{|\mathcal{A}|}$, and using $\lambda_{1:N}(a) \le N$,

$$\sum_{i=1}^{N}\sum_{a\in\mathcal{A}} \frac{\lambda_i(a)}{\lambda_{1:i}(a) + \frac{N\varepsilon}{|\mathcal{A}|}} \le \sum_{a\in\mathcal{A}} \log\left(1 + \frac{|\mathcal{A}|\,\lambda_{1:N}(a)}{N\varepsilon}\right) \le |\mathcal{A}|\log\left(1 + \frac{|\mathcal{A}|}{\varepsilon}\right);$$

taking the supremum over N and the sequence gives the claimed bound.

Per-context generalized linear: by the elliptical potential lemma, $\operatorname{tr}\left((A+B)^{-1}B\right) \le \log\frac{\det(A+B)}{\det A}$ for $A \succ 0$ and $B \succeq 0$ (apply $\frac{x}{1+x} \le \log(1+x)$ to the eigenvalues of $A^{-1/2}BA^{-1/2}$); with $A = \Sigma_{\lambda_{1:i-1}} + N\varepsilon' I$ and $B = \Sigma_{\lambda_i}$, so that $A + B = \Sigma_{\lambda_{1:i}} + N\varepsilon' I$, the sum telescopes:

$$\kappa^2 \sum_{i=1}^{N} \operatorname{tr}\left(\left(\Sigma_{\lambda_{1:i}} + N\varepsilon' I\right)^{-1}\Sigma_{\lambda_i}\right) \le \kappa^2 \log\frac{\det\left(N\varepsilon' I + \Sigma_{\lambda_{1:N}}\right)}{\det\left(N\varepsilon' I\right)} \le \kappa^2 d\log\left(1 + \frac{1}{d\varepsilon'}\right),$$

where the last step uses $\log\frac{\det(N\varepsilon' I+\Sigma_{\lambda_{1:N}})}{\det(N\varepsilon' I)} = \log\det\left(I + \frac{1}{N\varepsilon'}\Sigma_{\lambda_{1:N}}\right) \le d\log\left(1 + \frac{\operatorname{tr}(\Sigma_{\lambda_{1:N}})}{dN\varepsilon'}\right) \le d\log\left(1 + \frac{1}{d\varepsilon'}\right)$, and $\lambda_{\max}\left(\Sigma_{\lambda_{1:N}}\right) \le \operatorname{tr}\left(\Sigma_{\lambda_{1:N}}\right) \le N$ (as $||\phi(a)||_2 \le 1$).

h-smoothed: applying Eq. (26) per action under μ with $a_i = \lambda_i(a)$ and $\varepsilon_0 = N\varepsilon$, and using $\lambda_{1:N}(a) \le \frac{N}{h}$,

$$\frac{1}{h}\sum_{i=1}^{N} \mathbb{E}_{a\sim\mu}\left[\frac{\lambda_i(a)}{\lambda_{1:i}(a) + N\varepsilon}\right] \le \frac{1}{h}\,\mathbb{E}_{a\sim\mu}\left[\log\frac{N\varepsilon + \lambda_{1:N}(a)}{N\varepsilon}\right] \le \frac{1}{h}\log\left(1 + \frac{1}{h\varepsilon}\right).$$

In each case, taking the supremum over N and $\lambda_1, \dots, \lambda_N \in \Lambda$ gives the stated bound on $\overline{\mathsf{SEC}}_\varepsilon(\mathcal{G}, \Lambda)$; the relaxed DOEC bounds then follow from Theorem 7. ■

E.3. Admissibility of the Relaxed Coverages in the Running Examples

In this section, we verify Assumption 5 in our settings of interest: first for the trivial relaxation $\overline{\mathsf{Coverage}}_\varepsilon = \mathsf{Coverage}_\varepsilon$ (Lemma 37), and then for the relaxed coverages of three running examples (Lemma 38, whose step-size thresholds are quoted in Section 4; their relaxed SEC bounds are established separately in Appendix E.2.2, Lemma 36). The cushioned relaxations are needed only by Algorithm 2 and its analysis; Section 3 and its supporting Appendix C.2 work with the cushion-free relaxed coverages throughout and do not depend on this subsection.

Lemma 37 (Admissibility of the original coverage) *The original coverage* $\mathsf{Coverage}_\varepsilon$ *(Eq. (2)), extended to unnormalized covering measures as in line 3 of Algorithm 2 is admissible with step-size threshold* $\bar{\Delta} = \varepsilon$.

Proof Recall that $\mathsf{Coverage}_\varepsilon(p, \lambda; \mathcal{G}) = \sup_{h \in \mathcal{DG}} \frac{(\mathbb{E}_{a\sim\lambda}[h(a)])^2}{\varepsilon + \sum_{a\in\mathcal{A}} p(a)h(a)^2}$, where every $h \in \mathcal{DG}$ has range $[-1, 1]$.

Properties (i) and (ii). The covering measure p and the cushion parameter ε enter each ratio only through the denominator $\varepsilon + \sum_{a\in\mathcal{A}} p(a)h(a)^2$, which is nondecreasing in (p, ε) and scales by c under $(p, \varepsilon) \to (cp, c\varepsilon)$; hence the supremum over h is non-increasing in (p, ε) and jointly homogeneous of degree -1 in (p, ε).

Property (iii). The numerator $(\mathbb{E}_{a\sim\lambda}[h(a)])^2$ does not depend on the covering measure, while the denominator $\varepsilon + \sum_{a\in\mathcal{A}} p(a)h(a)^2$ is bounded below by ε and is 1-Lipschitz in p with respect to the total variation distance (as $h(a)^2 \le 1$). Hence each ratio is Lipschitz in p, uniformly over $h \in \mathcal{DG}$, and so is the supremum $\mathsf{Coverage}_\varepsilon(\cdot, \lambda; \mathcal{G})$; in particular, the coverage is continuous in its first argument, as required.

Property (iv). For every $h \in \mathcal{DG}$, every $\lambda \in \Lambda$, and every $0 \le \Delta \le \varepsilon$, since $\sum_{a\in\mathcal{A}} \Delta\lambda(a)h(a)^2 = \Delta\, \mathbb{E}_{a\sim\lambda}\left[h(a)^2\right]$,

$$\frac{\varepsilon + \sum_{a\in\mathcal{A}} p(a)h(a)^2}{\varepsilon + \sum_{a\in\mathcal{A}} p(a)h(a)^2 + \Delta\, \mathbb{E}_{a\sim\lambda}\left[h(a)^2\right]} \ge \frac{\varepsilon}{\varepsilon + \Delta\, \mathbb{E}_{a\sim\lambda}\left[h(a)^2\right]} \ge \frac{\varepsilon}{\varepsilon + \Delta} \ge \frac{1}{2},$$

where the first inequality uses the elementary fact that $\frac{A+B}{A+C} \ge \frac{B}{C}$ for $C \ge B \ge 0$ and $A \ge 0$, the second uses $h(a)^2 \le 1$, and the third uses $\Delta \le \varepsilon$. Taking the supremum over h gives $\mathsf{Coverage}_\varepsilon(p + \Delta\lambda, \lambda; \mathcal{G}) \ge \frac{1}{2}\,\mathsf{Coverage}_\varepsilon(p, \lambda; \mathcal{G})$, i.e., property (iv) holds with $\bar{\Delta} = \varepsilon$. ■

We now state the main result of this section, referenced in Section 4; the relaxed coverage with cushion parameter ε below are obtained from the cushion-free relaxed coverages of Section 3 by keeping the cushion parameter when performing the Cauchy–Schwarz relaxation and distributing it over the relevant directions (actions, eigendirections, or the base measure, respectively).

Lemma 38 (Validity and Admissibility of the relaxed coverages for the running examples) *In each of the following settings,* $\overline{\mathsf{Coverage}}_\varepsilon$ *is a valid relaxed coverage, i.e.,* $\overline{\mathsf{Coverage}}_\varepsilon(p, \lambda; \mathcal{G}) \ge \mathsf{Coverage}_\varepsilon(p, \lambda; \mathcal{G})$ *for every nonnegative measure* p *and* $\lambda \in \Lambda$*, and satisfies Assumption 5 with the step-size threshold* $\bar{\Delta}$ *given below:*

1. ***(Discrete action space)*** $|\mathcal{A}| < \infty$, $\mathcal{G} \subseteq [0, 1]^{\mathcal{A}}$, $\Lambda = \{\delta_a : a \in \mathcal{A}\}$*:*

$$\overline{\mathsf{Coverage}}_\varepsilon(p, \lambda; \mathcal{G}) = \sum_{a\in\mathcal{A}} \frac{\lambda(a)}{p(a) + \varepsilon/|\mathcal{A}|}, \quad \bar{\Delta} = \frac{\varepsilon}{|\mathcal{A}|};$$

2. ***(Per-context generalized linear reward)*** $\Lambda = \{\delta_a : a \in \mathcal{A}\}$, $\mathcal{G} = \left\{a \mapsto \sigma(\phi(a)^\top\theta) : \theta \in \Theta_x\right\} \subseteq [0, 1]^{\mathcal{A}}$ *with* $\Theta_x := \{\theta(x) : \theta \in \Theta\}$*, link function* σ *satisfying* $0 < \underline{L} \le \sigma' \le \overline{L}$ *and* $\kappa := \overline{L}/\underline{L}$*,* $||\phi(a)||_2 \le 1$*, and* $\sup_{\theta,\theta'\in\Theta_x} ||\theta - \theta'||_2 \le B$*: with* $\varepsilon' := \frac{\varepsilon}{\underline{L}^2 B^2}$*,*

$$\overline{\mathsf{Coverage}}_\varepsilon(p, \lambda; \mathcal{G}) = \kappa^2 \operatorname{tr}\left(\left(\Sigma_p + \varepsilon' I\right)^{-1} \Sigma_\lambda\right), \quad \bar{\Delta} = \min\left\{\varepsilon', 1\right\},$$

where $\Sigma_p := \sum_{a\in\mathcal{A}} p(a)\phi(a)\phi(a)^\top$*; taking* σ *to be the identity (*$\kappa = 1$*,* $\varepsilon' = \frac{\varepsilon}{B^2}$*) covers the per-context linear reward;*

3. ***(h-smoothed regret)*** $\mathcal{G} \subseteq [0,1]^{\mathcal{A}}$, $\Lambda = \Delta_h^{\mu}(\mathcal{A})$:

$$\overline{\mathsf{Coverage}}_{\varepsilon}(p, \lambda; \mathcal{G}) = \frac{1}{h}\mathbb{E}_{a\sim\mu}\left[\frac{\lambda(a)}{p(a)+\varepsilon}\right], \quad \bar{\Delta} = h\varepsilon,$$

where $\lambda(a)$ and $p(a)$ denote densities with respect to the base measure μ.

Two remarks are in order. First, the cushion parameter in these refinements is essential for admissibility: the cushion-free relaxed coverages of Section 3, albeit valid pointwise upper bounds of $\mathsf{Coverage}_{\varepsilon}$, have *infinite* relaxed SEC – already in the discrete case, taking $\lambda_i = \mathrm{Unif}(\mathcal{A})$ for all i makes each term in Eq. (5) equal to $\frac{1}{i}$, so the sum grows as $\ln N$ – and they also violate dilution stability (property (iv) of Assumption 5) at $p = 0$, where the coverage is infinite. Second, since the relaxed coverages with cushion are pointwise no larger than their cushion-free counterparts, they remain valid (indeed tighter) inputs to OE2D, and the certified relaxed DOEC bounds of Table 2 in Appendix C.2 continue to hold for them.

Proof [Proof of Lemma 38] Throughout, write $h := g - g'$ for $g, g' \in \mathcal{G}$, so that $\mathsf{Coverage}_{\varepsilon}(p, \lambda; \mathcal{G}) = \sup_{h\in\mathcal{DG}} \frac{(\mathbb{E}_{a\sim\lambda}[h(a)])^2}{\varepsilon+\sum_{a\in\mathcal{A}} p(a)h(a)^2}$. For each setting, we verify in order: the validity of the relaxation $\mathsf{Coverage}_{\varepsilon} \le \overline{\mathsf{Coverage}}_{\varepsilon}$; properties (i)–(iii) of Assumption 5 (monotonicity, homogeneity, and continuity); and the dilution-stability property (iv) with the stated step-size threshold $\bar{\Delta}$.

Validity of the relaxation. Each claim follows by re-running the Cauchy–Schwarz argument of Lemma 22 (Appendix C.2) while keeping the cushion parameter and distributing it over the relevant directions.

- *Discrete:* since $h(a)^2 \le 1$ and $\lambda(a)^2 \le \lambda(a)$,

$$\begin{aligned}\left(\sum_{a\in\mathcal{A}} \lambda(a)h(a)\right)^2 &\le \left(\sum_{a\in\mathcal{A}} \frac{\lambda(a)^2}{p(a)+\frac{\varepsilon}{|\mathcal{A}|}}\right)\left(\sum_{a\in\mathcal{A}} \left(p(a)+\tfrac{\varepsilon}{|\mathcal{A}|}\right) h(a)^2\right) \\ &\le \left(\sum_{a\in\mathcal{A}} \frac{\lambda(a)}{p(a)+\frac{\varepsilon}{|\mathcal{A}|}}\right)\left(\sum_{a\in\mathcal{A}} p(a)h(a)^2 + \varepsilon\right),\end{aligned}$$

  and dividing both sides by $\varepsilon + \sum_a p(a)h(a)^2$ and taking the supremum over h gives the claim.

- *Per-context generalized linear:* by the mean value theorem, $h(a) = \sigma'(\xi_a)\,\phi(a)^\top u$ for some ξ_a, where $u := \theta - \theta'$; hence $\underline{L}^2(\phi(a)^\top u)^2 \le h(a)^2 \le \overline{L}^2(\phi(a)^\top u)^2$. Applying Lemma 21 with $A = \Sigma_p + \varepsilon' I$ and $C = \Sigma_\lambda$, together with Jensen's inequality $\left(\mathbb{E}_{a\sim\lambda}[\phi(a)^\top u]\right)^2 \le u^\top \Sigma_\lambda u$,

$$\begin{aligned}(\mathbb{E}_{a\sim\lambda}[h(a)])^2 &\le \overline{L}^2\, u^\top \Sigma_\lambda u \le \overline{L}^2 \operatorname{tr}\left((\Sigma_p + \varepsilon' I)^{-1}\Sigma_\lambda\right) \cdot u^\top(\Sigma_p + \varepsilon' I)u \\ &\le \kappa^2 \operatorname{tr}\left((\Sigma_p + \varepsilon' I)^{-1}\Sigma_\lambda\right)\left(\sum_{a\in\mathcal{A}} p(a)h(a)^2 + \varepsilon\right),\end{aligned}$$

  where the last step uses $\underline{L}^2\, u^\top \Sigma_p u \le \sum_a p(a)h(a)^2$ and $\underline{L}^2\varepsilon'||u||_2^2 \le \underline{L}^2\varepsilon' B^2 = \varepsilon$.

- *h-smoothed:* by the Cauchy–Schwarz inequality with respect to μ, using $h(a)^2 \le 1$, $\lambda(a) \le \frac{1}{h}$, and that μ is a probability measure,

$$\begin{aligned}\left(\mathbb{E}_{a\sim\mu}\left[\lambda(a)h(a)\right]\right)^2 &\le \mathbb{E}_{a\sim\mu}\left[\frac{\lambda(a)^2}{p(a)+\varepsilon}\right]\cdot\mathbb{E}_{a\sim\mu}\left[(p(a)+\varepsilon)h(a)^2\right]\\ &\le \frac{1}{h}\mathbb{E}_{a\sim\mu}\left[\frac{\lambda(a)}{p(a)+\varepsilon}\right]\left(\mathbb{E}_{a\sim\mu}\left[p(a)h(a)^2\right]+\varepsilon\right).\end{aligned}$$

Properties (i) and (ii). In all three settings, the covering measure p and the cushion parameter ε enter only through the denominators $p(a)+\frac{\varepsilon}{|\mathcal{A}|}$, $\Sigma_p+\varepsilon' I$, and $p(a)+\varepsilon$, each of which is nondecreasing in (p,ε); monotonicity follows, in the generalized linear setting via the fact that $A \succeq B \succ 0$ implies $\operatorname{tr}\left(A^{-1}C\right) \le \operatorname{tr}\left(B^{-1}C\right)$ for $C \succeq 0$. Moreover, each denominator scales by c under $(p,\varepsilon)\to(cp,c\varepsilon)$, so $\overline{\mathsf{Coverage}}_\varepsilon(p,\lambda;\mathcal{G})$ scales by $\frac{1}{c}$; i.e. it is jointly homogeneous of degree -1 in (p,ε), which verifies property (ii).

Property (iii). In each setting $\overline{\mathsf{Coverage}}_\varepsilon(p,\lambda;\mathcal{G})$ depends on the covering measure p only through the reciprocal of a denominator. The cushion parameter $\varepsilon>0$ floors each denominator away from zero, keeping p in the region where these maps are continuous. Hence $\overline{\mathsf{Coverage}}_\varepsilon(\cdot,\lambda;\mathcal{G})$ is continuous in p with respect to the total variation distance, as property (iii) requires.

Property (iv). *Discrete* ($\bar\Delta=\frac{\varepsilon}{|\mathcal{A}|}$): for every $a\in\mathcal{A}$ and $0\le\Delta\le\bar\Delta$, since $\lambda(a)\le 1$,

$$\frac{p(a)+\frac{\varepsilon}{|\mathcal{A}|}}{p(a)+\Delta\lambda(a)+\frac{\varepsilon}{|\mathcal{A}|}} \ge \frac{\frac{\varepsilon}{|\mathcal{A}|}}{\frac{\varepsilon}{|\mathcal{A}|}+\Delta} \ge \frac{1}{2},$$

so each term of $\overline{\mathsf{Coverage}}_\varepsilon(p+\Delta\lambda,\lambda;\mathcal{G})$ is at least half the corresponding term of $\overline{\mathsf{Coverage}}_\varepsilon(p,\lambda;\mathcal{G})$. *Per-context generalized linear*: since $||\phi(a)||_2\le 1$, we have $\Sigma_\lambda \preceq I$, so for $\Delta\le\varepsilon'$, $\Sigma_p+\Delta\Sigma_\lambda+\varepsilon' I \preceq 2(\Sigma_p+\varepsilon' I)$ and thus $\operatorname{tr}\left((\Sigma_p+\Delta\Sigma_\lambda+\varepsilon' I)^{-1}\Sigma_\lambda\right) \ge \frac{1}{2}\operatorname{tr}\left((\Sigma_p+\varepsilon' I)^{-1}\Sigma_\lambda\right)$. *h-smoothed* ($\bar\Delta = h\varepsilon$): the density of $\Delta\lambda$ with respect to μ is at most $\frac{\Delta}{h}\le\varepsilon$, so for every a, $\frac{p(a)+\varepsilon}{p(a)+\Delta\lambda(a)+\varepsilon} \ge \frac{\varepsilon}{\varepsilon+\frac{\Delta}{h}} \ge \frac{1}{2}$. ■

E.4. Fast Termination With Large Step Size in Finite Eluder Dimension

We found that when the $\overline{\mathsf{SEC}}_\varepsilon(\mathcal{G},\Lambda)$ is small, such as in the finite Eluder dimension case, the step size Δ_t in Algorithm 2 can be chosen as large as $\Delta_t = \frac{1}{4\overline{\mathsf{Coverage}}_\varepsilon(p_{t-1},\lambda_t;\mathcal{G})}$ and the algorithm will terminate in a small number of iterations. Underpinning the proposition is a single-action reweighting bound for the coverage, which we record first.

Lemma 39 (Single-action reweighting of the coverage) *Let $\overline{\mathsf{Coverage}}_\varepsilon$ be the original coverage $\mathsf{Coverage}_\varepsilon$ or one of the discrete and per-context generalized linear relaxed coverages of Lemma 38. For every measure p, action $a\in\mathcal{A}$, and scalar $\Delta\ge 0$,*

$$\overline{\mathsf{Coverage}}_\varepsilon(p+\Delta\,\delta_a,\delta_a;\mathcal{G}) \ge \frac{\overline{\mathsf{Coverage}}_\varepsilon(p,\delta_a;\mathcal{G})}{1+\Delta\,\overline{\mathsf{Coverage}}_\varepsilon(p,\delta_a;\mathcal{G})}, \tag{27}$$

with equality except for the generalized linear coverage when $\kappa>1$.

Proof

- For the original coverage, by the definition of coverage and the monotonicity of $x \mapsto \frac{x}{1+x}$,

$$\mathsf{Coverage}_\varepsilon(p+\Delta\delta_a, \delta_a) = \sup_{h \in \mathcal{DG}} \frac{h(a)^2}{\varepsilon + \sum_{a' \in \mathcal{A}} p(a')\, h(a')^2 + \Delta\, h(a)^2} = \frac{\mathsf{Coverage}_\varepsilon(p, \delta_a)}{1 + \Delta\, \mathsf{Coverage}_\varepsilon(p, \delta_a)}.$$

- For the Discrete and per-context generalized linear. Both relaxed coverages take the form $\overline{\mathsf{Coverage}}_\varepsilon(p, \delta_a; \mathcal{G}) = \kappa^2\, \phi(a)^\top \bar{\Sigma}_p^{-1} \phi(a)$ with $\bar{\Sigma}_p = \Sigma_p + \varepsilon' I$ – the discrete coverage $\frac{1}{p(a)+\varepsilon/|\mathcal{A}|}$ being the case $\phi(a) = e_a$, $\kappa = 1$, $\varepsilon' = \varepsilon/|\mathcal{A}|$. The reweighting $p \mapsto p + \Delta\delta_a$ is the rank-one update $\bar{\Sigma}_{p+\Delta\delta_a} = \bar{\Sigma}_p + \Delta\, \phi(a)\phi(a)^\top$, so the Sherman–Morrison formula gives $\phi(a)^\top \bar{\Sigma}_{p+\Delta\delta_a}^{-1} \phi(a) = \frac{\phi(a)^\top \bar{\Sigma}_p^{-1} \phi(a)}{1+\Delta\, \phi(a)^\top \bar{\Sigma}_p^{-1} \phi(a)}$; multiplying by κ^2,

$$\overline{\mathsf{Coverage}}_\varepsilon(p + \Delta\delta_a, \delta_a) = \frac{\overline{\mathsf{Coverage}}_\varepsilon(p, \delta_a)}{1 + \Delta\, \overline{\mathsf{Coverage}}_\varepsilon(p, \delta_a)/\kappa^2} \geq \frac{\overline{\mathsf{Coverage}}_\varepsilon(p, \delta_a)}{1 + \Delta\, \overline{\mathsf{Coverage}}_\varepsilon(p, \delta_a)},$$

  where the inequality uses $\kappa \geq 1$, with equality when $\kappa = 1$.

■

Proposition 40 (Fast termination under the aggressive step) *Suppose $\Lambda = \{\delta_a : a \in \mathcal{A}\}$ and $\overline{\mathsf{Coverage}}_\varepsilon$ is the original coverage $\mathsf{Coverage}_\varepsilon$ or one of the discrete and per-context generalized linear relaxed coverages defined in Lemma 38. Then Algorithm 2 with the aggressive step size $\Delta_t = \frac{1}{4\, \overline{\mathsf{Coverage}}_\varepsilon(p_{t-1}, \lambda_t; \mathcal{G})}$ terminates in at most $\lfloor 32\, \overline{\mathsf{SEC}}_\varepsilon(\mathcal{G}, \Lambda) \rfloor$ iterations and outputs $p^* \in \mathrm{co}(\Lambda)$ satisfying Eq. (6), certifying $\mathsf{doec}_{\gamma,\varepsilon}(\mathcal{G}, \Lambda) \leq \frac{10}{\gamma}\, \overline{\mathsf{SEC}}_\varepsilon(\mathcal{G}, \Lambda)$. In particular, the iteration count is $\lfloor 32\, \mathsf{SEC}_\varepsilon(\mathcal{G}, \Lambda) \rfloor$ for the trivial relaxation $\overline{\mathsf{Coverage}}_\varepsilon = \mathsf{Coverage}_\varepsilon$, and $\widetilde{O}(|\mathcal{A}|)$ and $\widetilde{O}(\kappa^2 d)$ for the discrete and per-context generalized linear relaxed coverages, respectively.*

Proof *Aggressive one-step stability.* We first record the consequence of Lemma 39 that drives the analysis: for every measure p and action $a \in \mathcal{A}$, taking $\Delta = \frac{1}{4\, \overline{\mathsf{Coverage}}_\varepsilon(p, \delta_a)}$ in the reweighting bound (27) (so $\Delta\, \overline{\mathsf{Coverage}}_\varepsilon(p, \delta_a) = \frac{1}{4}$),

$$\begin{aligned} \overline{\mathsf{Coverage}}_\varepsilon(p + \Delta\, \delta_a, \delta_a; \mathcal{G}) &\geq \frac{\overline{\mathsf{Coverage}}_\varepsilon(p, \delta_a; \mathcal{G})}{1 + \Delta\, \overline{\mathsf{Coverage}}_\varepsilon(p, \delta_a; \mathcal{G})} = \tfrac{4}{5}\, \overline{\mathsf{Coverage}}_\varepsilon(p, \delta_a; \mathcal{G}) \\ &\geq \tfrac{3}{4}\, \overline{\mathsf{Coverage}}_\varepsilon(p, \delta_a; \mathcal{G}). \end{aligned} \tag{28}$$

Potential function. We track

$$\Phi_t = \underbrace{-\sum_{s=1}^{t} \Delta_s\, \overline{\mathsf{Coverage}}_\varepsilon(p_s, \lambda_s; \mathcal{G})}_{F_t} + \gamma \underbrace{\sum_{s=1}^{t} \tfrac{1}{2} \Delta_s\, \widehat{R}(\lambda_s)}_{G_t \geq 0},$$

which satisfies $\Phi_0 = 0$ and $\Phi_t \geq -\overline{\mathsf{SEC}}_{\varepsilon/||p_t||}(\mathcal{G}, \Lambda)$; both properties hold for any relaxed coverage, independently of the step size (as in Theorem 7).

Per-step decrease. Consider an iteration t at which the violation check (line 5) fires, and write $\lambda_t = \delta_{a_t}$, so $\Delta_t = \frac{1}{4\,\overline{\mathsf{Coverage}}_\varepsilon(p_{t-1},\delta_{a_t})}$ and $p_t = p_{t-1} + \Delta_t \delta_{a_t}$. Applying (28) at $(p,a) = (p_{t-1}, a_t)$ gives $\overline{\mathsf{Coverage}}_\varepsilon(p_t,\lambda_t) \geq \frac{3}{4}\,\overline{\mathsf{Coverage}}_\varepsilon(p_{t-1},\lambda_t)$, so

$$\begin{aligned}\Phi_{t-1} - \Phi_t &= \Delta_t\,\overline{\mathsf{Coverage}}_\varepsilon(p_t,\lambda_t;\mathcal{G}) - \tfrac{\gamma}{2}\Delta_t\,\widehat{R}(\lambda_t) \\ &\geq \Delta_t\left(\tfrac{3}{4}\,\overline{\mathsf{Coverage}}_\varepsilon(p_{t-1},\lambda_t;\mathcal{G}) - \tfrac{\gamma}{2}\,\widehat{R}(\lambda_t)\right) \geq \tfrac{\Delta_t}{4}\,\overline{\mathsf{Coverage}}_\varepsilon(p_{t-1},\lambda_t;\mathcal{G}) = \tfrac{1}{16},\end{aligned}$$

where the second inequality uses $\overline{\mathsf{Coverage}}_\varepsilon(p_{t-1},\lambda_t;\mathcal{G}) \geq \gamma\,\widehat{R}(\lambda_t) + 8\,\overline{\mathsf{SEC}}_\varepsilon(\mathcal{G},\Lambda) \geq \gamma\,\widehat{R}(\lambda_t)$ at a violating iteration, and the final equality is the choice of Δ_t.

Termination. Were iteration $t = \lfloor 32\,\overline{\mathsf{SEC}}_\varepsilon(\mathcal{G},\Lambda)\rfloor$ reached, the per-step decrease would give $\Phi_t \leq \Phi_0 - \frac{t}{16} < -\overline{\mathsf{SEC}}_\varepsilon(\mathcal{G},\Lambda)$. But $\Delta_s \leq \frac{1}{32\,\overline{\mathsf{SEC}}_\varepsilon(\mathcal{G},\Lambda)}$ for every s, so $||p_t||_1 = \sum_{s=1}^t \Delta_s \leq 1$ and hence $\Phi_t \geq -\overline{\mathsf{SEC}}_{\varepsilon/||p_t||}(\mathcal{G},\Lambda) \geq -\overline{\mathsf{SEC}}_\varepsilon(\mathcal{G},\Lambda)$ – a contradiction. The algorithm therefore halts within $\lfloor 32\,\overline{\mathsf{SEC}}_\varepsilon(\mathcal{G},\Lambda)\rfloor$ iterations with $||p^*||_1 \leq 1$, and p^* satisfies the (LR) and (GC) properties, certifying $\mathsf{doec}_{\gamma,\varepsilon}(\mathcal{G},\Lambda) \leq \frac{10}{\gamma}\,\overline{\mathsf{SEC}}_\varepsilon(\mathcal{G},\Lambda)$.

Iteration counts. The general bound $\lfloor 32\,\overline{\mathsf{SEC}}_\varepsilon(\mathcal{G},\Lambda)\rfloor$ specializes, via the relaxed SEC bounds of Lemma 36, to $\lfloor 32\,\mathsf{SEC}_\varepsilon(\mathcal{G},\Lambda)\rfloor$ for the trivial relaxation and to $\widetilde{O}\big(|\mathcal{A}|\big)$ and $\widetilde{O}\big(\kappa^2 d\big)$ for the discrete and per-context generalized linear relaxed coverages, respectively. ■

Remark 41 *The above proof utilizes the specific structure that $\Lambda = \{\delta_a : a \in \mathcal{A}\}$ to simplify $\Delta_t\mathsf{Coverage}_\varepsilon(p_t,\lambda_t) = \frac{\Delta_t\mathsf{Coverage}_\varepsilon(p_{t-1},\lambda_t)}{1+\Delta_t\mathsf{Coverage}_\varepsilon(p_{t-1},\lambda_t)}$. This is consistent with the step size used in Xu and Zeevi (2020, Section 4). In general, we don't have such inequality and in the proof of Proposition 30, we instead invoke the dilution-stability property (iv) of Assumption 5; for the original coverage, this is verified in Lemma 37 by utilizing the "$\varepsilon + \sum_{a\in\mathcal{A}} p_t(a)h(a)^2$" in the denominator in the definition of coverage, as well as the small step size $\Delta_t \leq \varepsilon$ to ensure that $\Delta_t\mathsf{Coverage}_\varepsilon(p_t,\lambda_t) \geq \frac{1}{2}\Delta_t\mathsf{Coverage}_\varepsilon(p_{t-1},\lambda_t)$. This is similar in spirit to the potential decreasing argument in Agarwal et al. (2014) with each member policy defined by mixing in with some uniform exploration.*

In light of Proposition 34, an important consequence of Proposition 40 is that, when the Eluder dimension of $\mathcal{G}$ is finite, Alg. 2 terminates in $\widetilde{O}\big(\mathrm{Edim}(\mathcal{G},\sqrt{\varepsilon})\big)$ iterations.

E.5. Computation Costs of Algorithm 2

The discrete and h-smoothed running examples do not require Algorithm 2: their relaxed exploitative F-designs are the closed-form (capped) inverse-gap-weighting distributions of Lemma 23, computable in $O(|\mathcal{A}|)$ and $O(|\mathcal{A}|\log|\mathcal{A}|)$ time, respectively (Abe and Long, 1999; Foster and Rakhlin, 2020; Zhu and Mineiro, 2022). We focus on the per-context generalized linear example, where the algorithm is genuinely run, and bound the cost of a single call with its cushioned relaxed coverage (Lemma 38). We count arithmetic operations, with reading $\hat{g}(a)$ and a feature $\phi(a) \in \mathbb{R}^d$ costing $O(1)$ and $O(d)$, and assume line 3 is solved by scanning the $|\mathcal{A}|$ actions.

With $\bar{\Sigma}_p := \Sigma_p + \varepsilon' I$, the relaxed coverage of a point mass is the quadratic form $\overline{\mathsf{Coverage}}_\varepsilon(p,\delta_a;\mathcal{G}) = \kappa^2\,\phi(a)^\top\bar{\Sigma}_p^{-1}\phi(a)$. Using the Sherman–Morrison formula to maintain $\bar{\Sigma}_p^{-1}$, each iteration takes $O(|\mathcal{A}|\,d^2)$ time, matching the per-step cost of the linear F-design of Xu and Zeevi (2020, Section 4).

Since $\Lambda = \{\delta_a : a \in \mathcal{A}\}$ here, the aggressive step (Proposition 40) applies, with iteration count equal to the relaxed SEC bound $\widetilde{O}(\kappa^2 d)$ of the linear example (Lemma 36, in the feature dimension d); a single call of Algorithm 2 therefore costs $\widetilde{O}(|\mathcal{A}|\,\kappa^2 d^3)$.

End-to-end over the full run, and comparison to UCCB. Within OE2D, the relaxed exploitative F-design is solved once per round, so over a horizon T the action-distribution cost is T times the per-call cost above, while the offline regression oracle is called only $O(\log T)$ times. For the per-context generalized linear example under the aggressive step (one call in $\widetilde{O}(|\mathcal{A}|\,\kappa^2 d^3)$), writing $M(n)$ for one offline-regression solve on n samples, relaxed OE2D spends $\widetilde{O}(|\mathcal{A}|\,\kappa^2 d^3 T)$ on action computation and $O(M(T)\log T)$ on regression. The UCCB algorithm of Xu and Zeevi (2020) instead spends $O(|\mathcal{A}|\,d^2 T^2)$ on action computation since it recomputes counterfactual actions against all past contexts, and $O(M(T)\,T)$ on regression as calling the oracle every round. Since $d \ll T$, $\kappa = O(1)$ for well-conditioned links, and $M(n) = \Omega(n)$, relaxed OE2D is substantially cheaper end-to-end, and the design subproblem does not dominate its runtime.

E.6. Proof of Proposition 8 and Discussions

Proof We use the "cheating code" example in Agarwal et al. (2024); Amin et al. (2011); Jun and Zhang (2020b).

We define the action space $\mathcal{A}$ to have two disjoint parts: $\mathcal{A}_1 = \{a_0, \ldots, a_{2^k-1}\}$ and $\mathcal{A}_2 = \{b_0, \ldots, b_{k-1}\}$. The reward function class $\mathcal{G} = \left\{g^1, \ldots g^{2^k}\right\}$, such that:

$$\begin{cases} g^i(a_j) = I(i=j), & j \in \{0, \ldots, 2^k - 1\} \\ g^i(b_l) = \frac{1}{2} \cdot \text{the } l\text{-th bit of number } i, & j \in \{0, \ldots, k-1\} \end{cases}$$

We first show the upper bound on $\mathsf{doec}_{\gamma,\varepsilon}(\mathcal{G}, \Lambda)$. For any function $\hat{g} \in \mathcal{G}$, let $\hat{a} = \operatorname{argmax}_{a \in \mathcal{A}} \hat{g}(a)$ be its greedy action. We choose distribution $p = (1-\beta)\delta_{\hat{a}} + \beta \mathrm{Uniform}(\mathcal{A}_2)$ to certify an upper bound on $\mathsf{doec}_{\gamma,\varepsilon}(\mathcal{G}, \Lambda)$.

Specifically, for any distribution $\lambda \in \Lambda$,

$$\begin{aligned} \mathsf{Coverage}_{\varepsilon}(p, \lambda; \mathcal{G}) \leq & \sup_{g, g' \in \mathcal{G}} \frac{(\mathbb{E}_{a \sim \lambda}[g(a) - g'(a)])^2}{\varepsilon + \beta \mathbb{E}_{a \sim \mathrm{Uniform}(\mathcal{A}_2)} (g(a) - g'(a))^2} \\ \leq & \frac{4k}{\beta} \end{aligned}$$

where the second inequality uses that for any $g, g' \in \mathcal{G}$ such that $g \neq g'$, $\mathbb{E}_{a \sim \mathrm{Uniform}(\mathcal{A}_2)}(g(a) - g'(a))^2 \geq \frac{1}{4k}$.

Thus, for any $\lambda \in \Lambda$,

$$\mathbb{E}_{a \sim \lambda} \hat{g}(a) - \mathbb{E}_{a \sim \lambda} \hat{g}(a) + \frac{\mathsf{Coverage}_{\varepsilon}(p, \lambda; \mathcal{G})}{\gamma} \leq \beta + \frac{4k}{\beta\gamma}$$

and choosing $\beta = \min\left(2\sqrt{\frac{k}{\gamma}}, 1\right)$ implies that p_β certifies $\mathsf{doec}_{\gamma,\varepsilon}(\mathcal{G}, \Lambda) \leq 4\left(\sqrt{\frac{k}{\gamma}} + \frac{k}{\gamma}\right)$.

We next show the lower bound on $\mathsf{SEC}_\varepsilon(\mathcal{G}, \Lambda)$. Consider the sequence of measures $\delta_{a_0}, \ldots, \delta_{a_{2^k-1}}$. Then

$$\begin{aligned}\mathsf{SEC}_\varepsilon(\mathcal{G}, \Lambda) \geq & \sum_{i=0}^{2^k-1} \mathsf{Coverage}_{2^k\varepsilon}\left(\sum_{j=0}^{i} \delta_{a_j}, \delta_{a_i}\right) \\ \geq & \sum_{i=0}^{2^k-1} \frac{1}{2^k\varepsilon + 2} \\ = & \frac{2^k}{2^k\varepsilon + 2} \\ \geq & \min\left(2^{k-2}, \frac{1}{2\varepsilon}\right)\end{aligned}$$

Here, the first inequality is by the definition of SEC; the second inequality is by the observation that $\mathsf{Coverage}_{2^k\varepsilon}\left(\sum_{j=0}^{i} \delta_{a_j}, \delta_{a_i}\right) \geq \frac{1}{2^k\varepsilon+2}$ – this is certified by taking $g = g^1$ and $g' = g^0$ for $i = 0$ and $g = g^i$ and $g' = g^0$ for $i \geq 1$; the rest of the calculations are by algebra. ■

Implications to the regret bounds of OE2D. Given that $\mathsf{doec}_{\gamma,\epsilon}(\mathcal{G}, \Lambda) \lesssim \sqrt{\frac{k}{\gamma}} + \frac{k}{\gamma}$, for OE2D, we can set $\tau_m = 2^m$, $\gamma_m = (\frac{\tau_m}{\ln|\mathcal{F}|})^{\frac{2}{3}} k^{\frac{1}{3}}$, so that Theorem 4 gives a nontrivial regret bound of $\widetilde{O}\big((k \ln|\mathcal{F}|)^{1/3} T^{2/3} + k \ln|\mathcal{F}|\big)$.

In contrast, suppose we only know that $\mathsf{doec}_{\gamma,\epsilon}(\mathcal{G}, \Lambda) \leq \min(\frac{2^k}{\gamma}, \frac{1}{\gamma\varepsilon})$, we cannot hope for Theorem 4 to give a regret bound better than $\min(\sqrt{2^k}, T)$. The reason is as follows:

- If there exists some $m \geq 1$ with $\varepsilon_m < \frac{1}{2^k}$, then the regret bound is at least

$$\frac{\tau_2}{\gamma_2} \cdot \left(2^k + \gamma_2^2 \frac{\ln|\mathcal{F}|}{\tau_1}\right) \geq \frac{\tau_m}{\gamma_m} \cdot \gamma_m \sqrt{\frac{2^k \ln|\mathcal{F}|}{\tau_{m-1}}} \geq \sqrt{2^k}.$$

- Otherwise, for all $m \geq 1$, $\varepsilon_m \geq \frac{1}{2^k}$. Then, for every $m \geq 2$, the second term of the regret bound is at least

$$\frac{1}{\varepsilon_{m-1}} + \gamma_m^2 \varepsilon_{m-1} \geq \sqrt{\frac{1}{\varepsilon_{m-1}} \cdot \gamma_m^2 \varepsilon_{m-1}} = \gamma_m,$$

  where in the first term, we lower bound the maximum by the $(m-1)$-th term in $\max_{n\in[M]}$, and in the second term, we lower bound the maximum by the m-th term in the $\max_{n\in\{2,\ldots,M\}}$ operation. Thus, the regret bound is at least of order

$$\max\left(\tau_1, \max_{m\geq 2} \frac{\tau_m}{\gamma_m} \cdot \max_{m\geq 2} \gamma_m\right) \geq \max_{m\geq 1} \tau_m \geq \frac{T}{M},$$

  which is vacuous.

Appendix F. Proofs from Section 5

F.1. Proof of Theorem 9

Theorem 9 *For any function class $\mathcal{G} : \mathcal{X} \times \mathcal{A} \to [0, 1]$, any set of action distributions $\Lambda \subseteq \Delta(\mathcal{A})$, any $\gamma > 0$ and any $\varepsilon \geq 0$, we have that any distribution p that certifies $\mathsf{doec}_\gamma(\mathcal{G}, \Lambda) \leq V$ also certifies that $\mathsf{dec}_\gamma(\mathcal{G}, \Lambda) \leq V + \frac{1}{\gamma} + \gamma\varepsilon$. As a consequence, $\mathsf{dec}_\gamma(\mathcal{G}, \Lambda) \leq \mathsf{doec}_{\gamma,\varepsilon}(\mathcal{G}, \Lambda) + \frac{1}{\gamma} + \gamma\varepsilon$.*

Proof [Proof of Theorem 9] For notational simplicity, we define $\mathbb{E}_\lambda[g(\cdot)] := \mathbb{E}_{a\sim\lambda}[g(\cdot)]$. Define λ^* as the optimal distribution from Λ that maximizes the expected reward by the true reward function g^*, i.e., $\lambda^* = \operatorname{argmax}_{\lambda\in\Lambda} \mathbb{E}_{a\sim\lambda}[g^*(a)]$. For any p^* that certifies $\mathsf{doec}_\gamma(\mathcal{G}, \Lambda) \leq V$, we decompose the difference into decision-error and estimation error as

$$\begin{aligned}
&\mathbb{E}_{\lambda^*}[g^*(a)] - \mathbb{E}_{p^*}[g^*(a)] \\
&= (\mathbb{E}_{\lambda^*}[\hat{g}(a)] - \mathbb{E}_{p^*}[\hat{g}(a)]) + (\mathbb{E}_{p^*}[\hat{g}(a)] - \mathbb{E}_{p^*}[g^*(a)]) + (\mathbb{E}_{\lambda^*}[g^*(a)] - \mathbb{E}_{\lambda^*}[\hat{g}(a)])
\end{aligned}$$

For the second difference, by Cauchy-Schwarz followed by AM-GM, we have

$$\mathbb{E}_{p^*}[\hat{g}(a)] - \mathbb{E}_{p^*}[g^*(a)] \leq \sqrt{\mathbb{E}_{p^*}\left[(\hat{g}(a) - g^*(a))^2\right]} \leq \frac{1}{\gamma} + \frac{\gamma}{4}\mathbb{E}_{p^*}\left[(\hat{g}(a) - g^*(a))^2\right]$$

For the last difference, we use the definition of coverage:

$$\begin{aligned}
\mathbb{E}_{\lambda^*}[g^*(a)] - \mathbb{E}_{\lambda^*}[\hat{g}(a)] &\leq \sqrt{\mathsf{Coverage}_\varepsilon(p^*, \lambda^*; \mathcal{G}) \cdot \left(\varepsilon + \mathbb{E}_{p^*}\left[(\hat{g}(a) - g^*(a))^2\right]\right)} \\
&\leq \frac{1}{\gamma}\mathsf{Coverage}_\varepsilon(p^*, \lambda^*; \mathcal{G}) + \frac{\gamma}{4}\mathbb{E}_{p^*}\left[(\hat{g}(a) - g^*(a))^2\right] + \frac{\gamma\varepsilon}{4}
\end{aligned}$$

Combining the bounds for the last two differences, we have

$$\begin{aligned}
&\mathbb{E}_{\lambda^*}[g^*(a)] - \mathbb{E}_{p^*}[g^*(a)] \\
&\leq (\mathbb{E}_{\lambda^*}[\hat{g}(a)] - \mathbb{E}_{p^*}[\hat{g}(a)]) + \frac{\gamma}{2}\mathbb{E}_{p^*}\left[(\hat{g}(a) - g^*(a))^2\right] + \frac{1}{\gamma}\mathsf{Coverage}_\varepsilon(p^*, \lambda^*; \mathcal{G}) + \frac{1}{\gamma} + \gamma\varepsilon \\
&\leq V + \gamma\mathbb{E}_{p^*}\left[(\hat{g}(a) - g^*(a))^2\right] + \frac{1}{\gamma} + \gamma\varepsilon \qquad (p^* \text{ certifies } \mathsf{doec}_\gamma(\mathcal{G}, \Lambda) \leq V) \\
&\Rightarrow \mathbb{E}_{\lambda^*}[g^*(a)] - \mathbb{E}_{p^*}[g^*(a)] - \gamma\mathbb{E}_{p^*}\left[(\hat{g}(a) - g^*(a))^2\right] \leq V + \frac{1}{\gamma} + \gamma\varepsilon \\
&\Rightarrow \mathsf{dec}_\gamma(\mathcal{G}, \Lambda) \leq V + \frac{1}{\gamma} + \gamma\varepsilon
\end{aligned}$$

which implies that p^* certifies that $\mathsf{dec}_\gamma(\mathcal{G}, \Lambda) \leq V + \frac{1}{\gamma} + \gamma\varepsilon$. The second part of the lemma follows by taking $V = \mathsf{doec}_{\gamma,\varepsilon}(\mathcal{G}, \Lambda)$. ■

F.2. SQUARECB.F and its Analysis

Since Algorithm 2 finds a distribution p that certifies that $\mathsf{doec}_\gamma(\mathcal{G}, \Lambda) \leq \frac{1}{\gamma}\overline{\mathsf{SEC}}_\varepsilon(\mathcal{G}, \Lambda)$, it also certifies that $\mathsf{dec}_\gamma(\mathcal{G}, \Lambda) \leq \frac{1}{\gamma}\overline{\mathsf{SEC}}_\varepsilon(\mathcal{G}, \Lambda) + \frac{1}{\gamma} + \gamma\varepsilon$. This motivates an online oracle-efficient algorithm, SQUARECB.F (Algorithm 4), with the following guarantees:

Algorithm 4 SQUARECB.F

Input: total time length T, learning parameter $\gamma, \varepsilon = 1/T$, function class $\mathcal{F}$, action distributions Λ, online regression oracle $\mathcal{O}_{\text{on}}$.

for $t = 1$ **to** T **do**

Compute $\hat{f}_t = \mathcal{O}_{\text{on}}(\mathcal{F})\left((x_i, a_i, r_i)_{i=1}^{t-1}\right)$ if $t > 1$, and an arbitrary element in $\mathcal{F}$ otherwise.

Observe context $x_t \in \mathcal{X}$.

Call Algorithm 2 to compute a sampling distribution p_t such that:

$$\max_{\lambda \in \Lambda} \left(\mathbb{E}_{a \sim \lambda} \left[\hat{f}_t(x_t, a) \right] - \mathbb{E}_{a \sim p_t} \left[\hat{f}_t(x_t, a) \right] + \frac{1}{\gamma} \mathsf{Coverage}_{\varepsilon}(p_t, \lambda; \mathcal{F}_{x_t}) \right) \leq \frac{10 \overline{\mathsf{SEC}}_{\varepsilon}(\mathcal{F}_{x_t}, \Lambda)}{\gamma}$$

Sample action $a_t \sim p_t$ and observe reward r_t.

Corollary 42 *For any function class $\mathcal{F} : \mathcal{X} \times \mathcal{A} \to [0, 1]$, any set of action distributions Λ, SQUARECB.F is computationally efficient if step 3 of Algorithm 2 can be implemented efficiently. In addition, with probability at least $1 - \delta$, its regret is bounded by:* $\mathrm{Reg}(T, \textit{SQUARECB.F}) \leq O\left(T \left(\frac{1}{\gamma} \max_{x \in \mathcal{X}} \overline{\mathsf{SEC}}_{\varepsilon}(\mathcal{F}_x, \Lambda) + \frac{1}{\gamma} + \gamma\varepsilon \right) + \gamma \mathrm{Reg}_{\text{off}}(\mathcal{F}, T, \delta) \right)$

When $\overline{\mathsf{SEC}}_{\varepsilon}(\mathcal{F}, \Lambda) = D\mathrm{polylog}(\frac{1}{\varepsilon})$ (as in Proposition 34 and Lemma 36), SQUARECB.F achieves a regret of $\widetilde{O}\left(\sqrt{DT \ln |\mathcal{F}|}\right)$. If step 3 of Algorithm 2 can be solved efficiently , SQUARECB.F may be a computationally attractive variant of E2D, albeit it may suffer from worse regret.

Proof [Proof of Corollary 42] The proof follows directly from Foster et al. (2021a, Theorem 8.1 and Remark 4.1), which gives the regret bound for E2D algorithm with an inexact minimizer in the contextual bandit setting with general function approximation. According to that theorem, since for each t, SQUARECB.F's choice p_t certifies that $\mathsf{doec}_{\gamma,\varepsilon}(\mathcal{F}_x, \Lambda) \leq \frac{10}{\gamma} \overline{\mathsf{SEC}}_{\varepsilon}(\mathcal{F}_x, \Lambda)$, by Theorem 9, it also certifies that $\mathsf{dec}_{\gamma}(\mathcal{F}_x, \Lambda) \leq \frac{10}{\gamma} \overline{\mathsf{SEC}}_{\varepsilon}(\mathcal{F}_x, \Lambda) + \frac{1}{\gamma} + \gamma\varepsilon$. Thus it achieves the following regret bound:

$$\mathrm{Reg}(T, \textsc{SquareCB.F}) \leq \sup_{x \in \mathcal{X}} \left(\frac{10}{\gamma} \overline{\mathsf{SEC}}_{\varepsilon}(\mathcal{F}_x, \Lambda) + \frac{1}{\gamma} + \gamma\varepsilon \right) \cdot T + \gamma \mathrm{Reg}_{\text{on}}(\mathcal{F}, T, \delta)$$

■

Appendix G. Experiments

We test whether replacing the per-round online regression oracle of SMOOTHIGW with a single offline oracle, refit once per epoch and explored through the closed-form exploitative F-design, costs anything in statistical performance. We compare our SMOOTHED-OE2D (Section 3) against SMOOTHIGW (Zhu and Mineiro, 2022) on continuous-action contextual bandits built from large regression datasets.

G.1. Setup

Datasets. We use five OpenML regression datasets under the standard reduction of Majzoubi et al. (2020): at round t the learner observes features x_t, plays an action $a_t \in [0, 1]$, and earns

$r_t = 1 - |a_t - y_t|$ for the normalized target $y_t \in [0, 1]$. Reward is maximized at $a_t = y_t$, so the target must be recovered from bandit feedback alone. Each dataset is processed in a single online pass of length T: `auto-price`, `cpu-act`, and `wisconsin` ($T \approx 10^6$), `black-friday` (1.7×10^5), and `zurich` (5.5×10^6).

Algorithms. We compare SMOOTHED-OE2D against SMOOTHIGW (Zhu and Mineiro, 2022), which is the state-of-the-art online regression oracle-efficient algorithm for continuous-action contextual bandits. Both methods smooth the action space with a smooth-h kernel and share the same oracle class; they differ only in how the oracle is queried (offline, once per epoch, versus online, every round) and in the exploration rule.[6] We try two regression oracles, a linear regression oracle and a nonlinear regression oracle that lifts the representation with random Fourier features approximating a Laplace kernel before fitting a linear model. Each smoothing parameter $h \in \{0.01, 0.02, 0.04, 0.08, 0.16, 0.32, 0.64\}$ yields a distinct smoothed problem instance, and we evaluate both methods on each.

Hyperparameter Tuning. We run a hyperparameter search over 10 runs with different permutations on the dataset, selecting for each configuration with the best progressive-validation reward (Blum et al., 1999), and then evaluate the selected configurations on another 10 runs with different permutations. The hyperparameter is the inverse-gap-weighting learning-rate multiplier γ, which both methods share and which trades off exploration against exploitation: we search it over a grid of 11 values and select a separate γ for each $(\text{dataset}, \text{oracle}, h)$ combination. For SMOOTHIGW we follow the pseudocode of Zhu and Mineiro (2022) rather than their released implementation. This is because their released codebase runs a Corral-type (Agarwal et al., 2017) method over multiple values of γ to sample an action and optimistically scales the suboptimality gap inside the IGW step as in Foster and Krishnamurthy (2021); tuning a single γ keeps the comparison on equal footing with SMOOTHED-OE2D.

G.2. Results

The final reported metric is the realized average reward $R_T = \frac{1}{T}\sum_{t=1}^{T} r_t$ over all time steps in each run (mean$\pm$std over the 10 runs) for each dataset and h cell. Tables 3 and 4 report the final realized reward for the linear and Laplace oracles over all datasets and h values, while Figures 1 and 2 show the learning curves for each cell.

Overall comparison. Both methods are best at small h and degrade as h grows, but SMOOTHED-OE2D is more sensitive to over-smoothing: at $h = 0.64$ its reward collapses (to ≈ 0.80 on `auto-price` and `cpu-act`) while SMOOTHIGW stays near its small-h value. This is because SMOOTHED-OE2D outputs a policy from an h-smoothed policy class while SMOOTHIGW's behavior policy is a mixture of policies from h-smoothed policy classes with the greedy policy (which is not smoothed). The Laplace oracle consistently helps SMOOTHED-OE2D and shrinks the gap with lower variance as well.

Figures 1 and 2 show the average-reward curves for every table cell. SMOOTHED-OE2D learns fastest in the low-data regime, while SMOOTHIGW, whose online oracle is updated one round at a time, frequently drops during its initial exploration before recovering. SMOOTHIGW then closes the gap as t grows.

6. Our implementation builds on the SMOOTHCB codebase.

Table 3: Final realized average reward R_T (mean±std over the held-out seeds) with the **linear** oracle, per dataset (rows) and h (columns). S-OE2D = SMOOTHED-OE2D (ours), S-IGW = SMOOTHIGW; **bold** marks the larger mean in each h cell. Leading zeros are omitted.

Dataset	$h = 0.01$		$h = 0.02$		$h = 0.04$		$h = 0.08$	
	S-OE2D	S-IGW	S-OE2D	S-IGW	S-OE2D	S-IGW	S-OE2D	S-IGW
auto-price	$.871_{\pm.032}$	$\mathbf{.928_{\pm.000}}$	$.899_{\pm.023}$	$\mathbf{.929_{\pm.000}}$	$.906_{\pm.024}$	$\mathbf{.928_{\pm.000}}$	$.875_{\pm.032}$	$\mathbf{.929_{\pm.000}}$
cpu-act	$.951_{\pm.006}$	$\mathbf{.959_{\pm.000}}$	$.943_{\pm.014}$	$\mathbf{.959_{\pm.000}}$	$.949_{\pm.004}$	$\mathbf{.959_{\pm.000}}$	$.921_{\pm.022}$	$\mathbf{.959_{\pm.000}}$
wisconsin	$.843_{\pm.024}$	$\mathbf{.857_{\pm.000}}$	$.836_{\pm.027}$	$\mathbf{.856_{\pm.001}}$	$.839_{\pm.025}$	$\mathbf{.857_{\pm.000}}$	$.817_{\pm.026}$	$\mathbf{.856_{\pm.001}}$
black-friday	$\mathbf{.832_{\pm.007}}$	$.831_{\pm.000}$	$.830_{\pm.007}$	$\mathbf{.833_{\pm.000}}$	$\mathbf{.836_{\pm.007}}$	$.831_{\pm.000}$	$\mathbf{.835_{\pm.007}}$	$.833_{\pm.000}$
zurich	$.951_{\pm.020}$	$\mathbf{.986_{\pm.000}}$	$.953_{\pm.013}$	$\mathbf{.986_{\pm.000}}$	$.954_{\pm.019}$	$\mathbf{.986_{\pm.000}}$	$.938_{\pm.011}$	$\mathbf{.986_{\pm.000}}$

Dataset	$h = 0.16$		$h = 0.32$		$h = 0.64$	
	S-OE2D	S-IGW	S-OE2D	S-IGW	S-OE2D	S-IGW
auto-price	$.891_{\pm.011}$	$\mathbf{.928_{\pm.000}}$	$.893_{\pm.000}$	$\mathbf{.929_{\pm.000}}$	$.803_{\pm.000}$	$\mathbf{.928_{\pm.000}}$
cpu-act	$.939_{\pm.002}$	$\mathbf{.959_{\pm.000}}$	$.906_{\pm.000}$	$\mathbf{.959_{\pm.000}}$	$.818_{\pm.000}$	$\mathbf{.959_{\pm.000}}$
wisconsin	$.821_{\pm.023}$	$\mathbf{.857_{\pm.000}}$	$.845_{\pm.000}$	$\mathbf{.856_{\pm.001}}$	$.780_{\pm.000}$	$\mathbf{.857_{\pm.000}}$
black-friday	$\mathbf{.832_{\pm.005}}$	$.831_{\pm.000}$	$.823_{\pm.001}$	$\mathbf{.833_{\pm.000}}$	$.772_{\pm.000}$	$\mathbf{.831_{\pm.000}}$
zurich	$.930_{\pm.009}$	$\mathbf{.986_{\pm.000}}$	$.886_{\pm.006}$	$\mathbf{.986_{\pm.000}}$	$.801_{\pm.001}$	$\mathbf{.986_{\pm.000}}$

Table 4: Final realized average reward R_T with the **Laplace (RFF)** oracle. Conventions as in Table 3.

Dataset	$h = 0.01$		$h = 0.02$		$h = 0.04$		$h = 0.08$	
	S-OE2D	S-IGW	S-OE2D	S-IGW	S-OE2D	S-IGW	S-OE2D	S-IGW
auto-price	$.878_{\pm.013}$	$\mathbf{.930_{\pm.001}}$	$.858_{\pm.010}$	$\mathbf{.929_{\pm.001}}$	$.866_{\pm.007}$	$\mathbf{.930_{\pm.001}}$	$.874_{\pm.005}$	$\mathbf{.929_{\pm.001}}$
cpu-act	$.916_{\pm.005}$	$\mathbf{.960_{\pm.000}}$	$.924_{\pm.003}$	$\mathbf{.960_{\pm.000}}$	$.910_{\pm.004}$	$\mathbf{.960_{\pm.000}}$	$.903_{\pm.006}$	$\mathbf{.960_{\pm.000}}$
wisconsin	$\mathbf{.864_{\pm.002}}$	$.863_{\pm.001}$	$.859_{\pm.004}$	$\mathbf{.862_{\pm.001}}$	$.858_{\pm.008}$	$\mathbf{.863_{\pm.001}}$	$.852_{\pm.010}$	$\mathbf{.862_{\pm.001}}$
black-friday	$\mathbf{.850_{\pm.003}}$	$.838_{\pm.002}$	$\mathbf{.849_{\pm.003}}$	$.839_{\pm.002}$	$\mathbf{.848_{\pm.003}}$	$.838_{\pm.002}$	$\mathbf{.850_{\pm.003}}$	$.839_{\pm.002}$
zurich	$.946_{\pm.002}$	$\mathbf{.948_{\pm.002}}$	$.945_{\pm.003}$	$\mathbf{.949_{\pm.003}}$	$.945_{\pm.003}$	$\mathbf{.948_{\pm.002}}$	$.942_{\pm.002}$	$\mathbf{.949_{\pm.003}}$

Dataset	$h = 0.16$		$h = 0.32$		$h = 0.64$	
	S-OE2D	S-IGW	S-OE2D	S-IGW	S-OE2D	S-IGW
auto-price	$.871_{\pm.004}$	$\mathbf{.930_{\pm.001}}$	$.895_{\pm.000}$	$\mathbf{.929_{\pm.001}}$	$.805_{\pm.000}$	$\mathbf{.930_{\pm.001}}$
cpu-act	$.901_{\pm.006}$	$\mathbf{.960_{\pm.000}}$	$.908_{\pm.000}$	$\mathbf{.960_{\pm.000}}$	$.820_{\pm.000}$	$\mathbf{.960_{\pm.000}}$
wisconsin	$.837_{\pm.015}$	$\mathbf{.863_{\pm.001}}$	$.849_{\pm.000}$	$\mathbf{.862_{\pm.001}}$	$.784_{\pm.000}$	$\mathbf{.863_{\pm.001}}$
black-friday	$\mathbf{.848_{\pm.003}}$	$.838_{\pm.002}$	$.837_{\pm.003}$	$\mathbf{.839_{\pm.002}}$	$.779_{\pm.002}$	$\mathbf{.838_{\pm.002}}$
zurich	$.934_{\pm.002}$	$\mathbf{.948_{\pm.002}}$	$.898_{\pm.001}$	$\mathbf{.949_{\pm.003}}$	$.802_{\pm.000}$	$\mathbf{.948_{\pm.002}}$

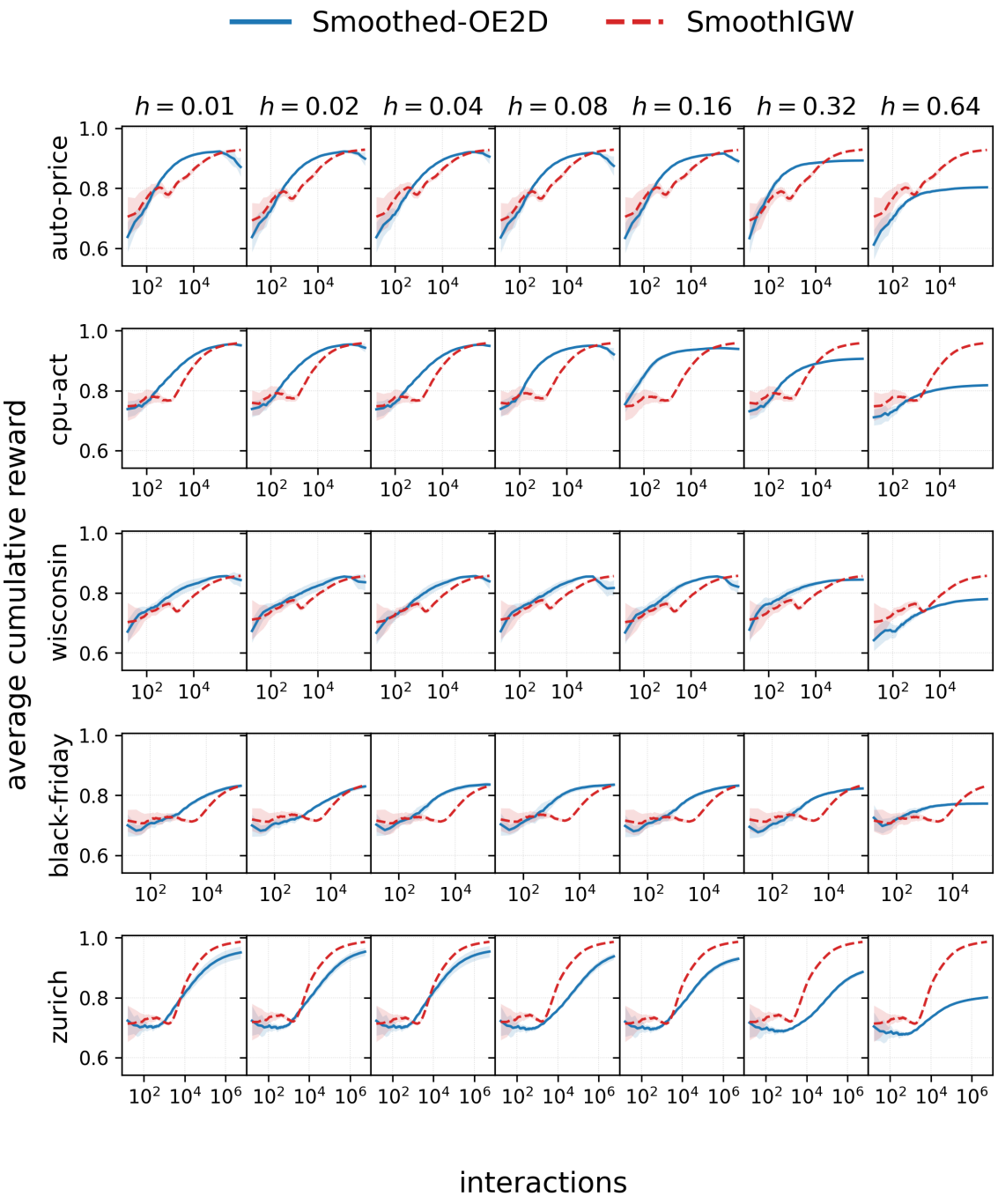

Figure 1: Average-reward learning curves with the **linear** oracle.

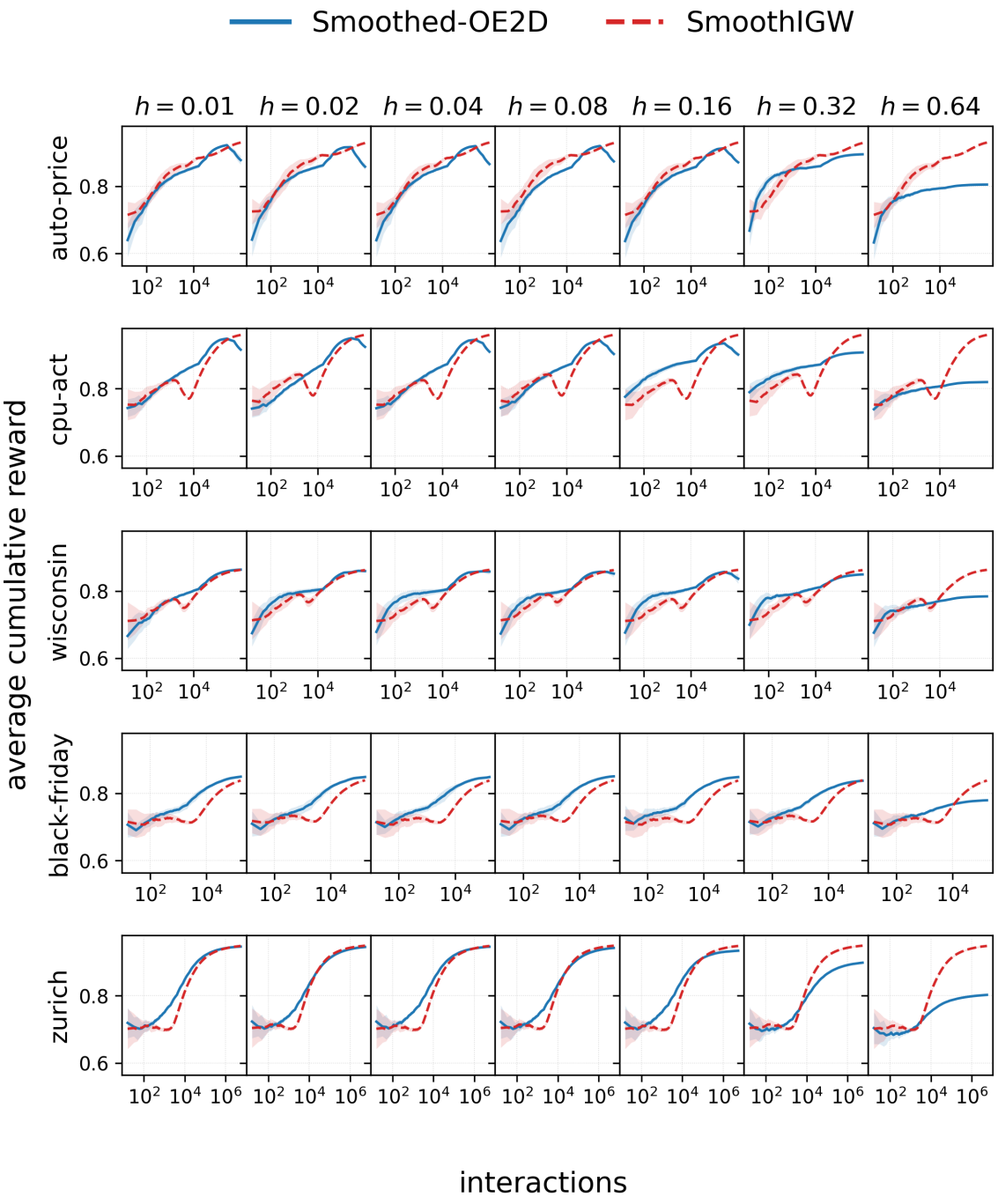

Figure 2: Average-reward learning curves with the **Laplace (RFF)** oracle.