INTERACTION-GROUNDED LEARNING FOR CONTEXTUAL MARKOV DECISION PROCESSES WITH PERSONALIZED FEEDBACK

Anonymous authors
Paper under double-blind review

ABSTRACT

In this paper, we study Interaction-Grounded Learning (IGL) (Xie et al., 2021), a paradigm designed for realistic scenarios where the learner receives indirect feedback generated by an unknown mechanism, rather than explicit numerical rewards. While prior work on IGL provides efficient algorithms with provable guarantees, those results are confined to single-step settings, restricting their applicability to modern sequential decision-making systems such as multi-turn Large Language Model (LLM) deployments. To bridge this gap, we propose an oracle-efficient algorithm that achieves a sublinear regret guarantee for contextual episodic Markov Decision Processes (MDPs) with personalized feedback. Technically, we extend the reward-estimator construction of Zhang et al. (2024b) from the single-step to the multi-step setting, addressing the unique challenges of decoding latent rewards under MDPs. Building on this estimator, we design an Inverse-Gap-Weighting (IGW) algorithm for policy optimization. Finally, we demonstrate the effectiveness of our method in learning personalized objectives from multi-turn interactions through experiments on both a synthetic episodic MDP and a real-world user booking dataset.

1 INTRODUCTION

While classic formulations of interactive learning assume access to direct numerical rewards, many modern interactive systems rarely receive such clean supervision. Instead, they must rely on indirect, user-facing feedback, such as thumbs-up signals in social media (Neophytou et al., 2022), short textual comments on review platforms (Hasan et al., 2025), electroencephalogram (EEG) signals in brain-computer interfaces (Akinola et al., 2020), or follow-up questions in conversational search that implicitly reveal satisfaction or dissatisfaction (Kim et al., 2024; Lin et al., 2024). This phenomenon is especially salient for Large Language Models (LLMs) deployed in conversational settings: the system produces a sequence of intermediate outputs over multiple turns, yet users typically evaluate the interaction holistically based on the final response quality and its alignment with their personal preferences.

Interaction-Grounded Learning (IGL), first proposed by Xie et al. (2021), offers a principled framework to model learning from such indirect feedback. Rather than receiving a direct reward as the feedback, IGL posits an underlying latent reward and views the observed feedback as a noisy proxy generated by an unknown mechanism. To make learning tractable, earlier works imposed many structural assumptions on the relationship between the feedback and the latent reward, such as requiring feedback to be conditionally independent of the context given the latent reward (Xie et al., 2021; 2022). While these assumptions enable provable guarantees, they are often too restrictive for practical settings. For instance, different users may express feedback in systematically different ways for the same outcome, or a single user may react differently depending on the conversational context. This highlights a critical need for a context-dependent grounding mechanism, rather than a universal mapping. Motivated by this, recent work has generalized the IGL framework to incorporate context-dependent feedback structures (Maghakian et al., 2023; Zhang et al., 2024b).

Despite this progress, to the best of our knowledge, all existing IGL literature is restricted to the single-step contextual bandit scenario. This abstraction misses the defining feature of many contemporary

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

applications: interactions are inherently sequential and multi-step. In an LLM-based system, for example, an agent may ask clarifying questions, invoke tools, or decide how much evidence to gather—decisions that fundamentally shape the trajectory of the conversation—before receiving any feedback. This combination of multi-step sequential decision-making and indirect feedback creates a fundamental gap between current IGL theory and the needs of modern systems. This motivates our central research question:

Can we design provably efficient algorithms for interaction-grounded learning in multi-step contextual Markov Decision Processes (MDPs) with personalized feedback?

Our Contribution. In this work, we answer this question affirmatively. We propose an oracle-efficient algorithm for IGL in contextual episodic MDPs with personalized feedback and establish a sublinear regret guarantee. At a high level, our approach decomposes this problem into *reward estimator learning* and *policy learning*. In Section 3, we develop a novel procedure for constructing a reward estimator from indirect feedback, comprising three main components: (i) *Reachable state identification*, which isolates terminal states that are reliably reachable by specific policies to ensure sufficient feedback data collection; (ii) *Inverse kinematic learning*, which estimates the posterior distribution of actions under a uniform policy conditioned on the reachable states identified in the previous step; and (iii) *A Lipschitz reward estimator*, which provides a smooth approximation of the latent reward function by leveraging the learned inverse kinematic at each reachable state. This estimator non-trivially generalizes the contextual bandit construction of Zhang et al. (2024b) to the multi-step setting. Building upon this estimator, in Section 4, we introduce a policy learning algorithm based on Inverse-Gap Weighting (IGW) that achieves a regret bound with $\widetilde{\mathcal{O}}(T^{3/4})$ dependence on the number of episodes T. Finally, in Section 5, we evaluate our approach on both a synthetic episodic MDP benchmark and a real user booking dataset. The empirical results demonstrate that our method effectively learns personalized objectives from multi-turn interactions. Due to the space limit, we defer the discussion on related works to Appendix B.

2 Preliminaries

Throughout this paper, we use $[N]$ to denote the set $\{1, 2, \ldots, N\}$ for any positive integer N, and $\Delta_{\mathcal{B}}$ to denote the probability simplex over a finite set $\mathcal{B}$. We study IGL with personalized feedback within the framework of contextual episodic MDPs with a finite horizon $H \geq 1$ and an unknown transition kernel. Formally, the MDP is defined by a finite state space $\mathcal{S}$ with cardinality S, a finite action set $\mathcal{A}$ with cardinality $|\mathcal{A}| = K$, and a transition kernel $\mathcal{P}$ on nonterminal state-action pairs, where $\mathcal{P}(s'|s, a)$ denotes the probability of transitioning to state s' after taking action a in state s.[1] Without loss of generality, we assume the state space $\mathcal{S}$ is partitioned into H disjoint layers, denoted as $\mathcal{S} = \bigcup_{h=1}^{H} \mathcal{S}_h$, where $\mathcal{S}_1 = \{s_1\}$ consists of a fixed singleton start state.[2] Transitions occur solely between consecutive layers, meaning that $\mathcal{P}(s'|s, a) > 0$ implies $s \in \mathcal{S}_h$ and $s' \in \mathcal{S}_{h+1}$ for some $h \in [H-1]$.

Learning Protocol. The interaction proceeds in episodes $t \in [T]$. At the beginning of each episode t, the learner observes a context $x_t \in \mathcal{X}$ drawn i.i.d from an unknown distribution $\mathcal{D}$. The learner then selects a policy $\pi_t : \mathcal{X} \times \mathcal{S} \to \Delta_{\mathcal{A}}$ based on the current context and history. Starting from the initial state $s_{t,1} = s_1$, for each step $h \in [H]$, the learner observes the current state $s_{t,h} \in \mathcal{S}_h$ and samples an action $a_{t,h} \sim \pi_t(\cdot|x_t, s_{t,h})$. If $h < H$, the next state is drawn as $s_{t,h+1} \sim \mathcal{P}(\cdot|s_{t,h}, a_{t,h})$. Note that we assume the transition dynamics are independent of the context. This modeling choice is applicable to a wide range of scenarios, including multi-task reinforcement learning, where the context defines the task while the underlying physical laws remain invariant across tasks.

Crucially, unlike classic contextual MDPs, the learner does not observe explicit rewards during the episode. Instead, after the trajectory $\tau_t = (s_{t,1}, a_{t,1}, \ldots, s_{t,H}, a_{t,H})$ concludes, the learner receives a feedback signal $y_t \in \mathcal{Y}$ for some observation space $\mathcal{Y}$. This signal serves as a proxy for an unobserved binary reward $r_t \in \{0, 1\}$, which is realized solely at the final step, i.e., $r_t \sim$

[1] For notational simplicity, we assume a common action set $\mathcal{A}$ across all states; however, our results extend straightforwardly to the case of state-dependent action sets.

[2] We adopt the singleton start state assumption solely for notational simplicity, following the line of works in the episodic MDP (Zimin and Neu, 2013; Neu et al., 2012; Rosenberg and Mansour, 2019; Jin et al., 2020a).

Bernoulli($f^\star(x_t, s_{t,H}, a_{t,H})$) for some fixed and unknown reward function $f^\star$. We focus on this *terminal-reward* MDP setting as it naturally models applications like LLM interactions, where user satisfaction depends primarily on the quality of the final response rather than the intermediate conversation leading up to it. This setup extends the contextual bandit framework ($H = 1$) studied in Zhang et al. (2024b) to the setting with general horizon H.

Feedback Structure. Since the true reward r_t is latent, learning is impossible without structural assumptions linking the observed feedback y_t to r_t. Extending the standard IGL framework with personalized feedback (Maghakian et al., 2023; Zhang et al., 2024b), we assume the feedback mechanism satisfies the following conditional independence property, which says that the feedback is conditionally independent of the trajectory given the context, the final state, and the realized reward.

Assumption 2.1 (Conditional Independence of Feedback)**.** For any tuple of context, trajectory, reward, and feedback (x, τ, r, y) where $\tau = (s_1, a_1, \ldots, s_H, a_H)$, the feedback y is conditionally independent of the trajectory prefix $(s_1, a_1, \ldots, s_{H-1}, a_{H-1})$ and the action a_H, given the context x, the final state s_H, and the realized reward r. Formally, $y \perp (s_1, a_1, \ldots, s_{H-1}, a_{H-1}, a_H) \mid (x, s_H, r)$. The feedback is generated by a fixed kernel $Q(\cdot \mid x, s, r)$ in every episode: conditional on the past history and the current (x_t, τ_t, r_t), y_t is drawn from $Q(\cdot \mid x_t, s_{t,H}, r_t)$.

This assumption captures the intuition that user feedback (e.g., a "thumbs up") is a reaction to the final output quality (the realized reward) relative to the user's intent, rather than the specific means used to achieve it. Consequently, conditioned on the reward, the feedback is independent of both the internal reasoning path and the final action a_H, as any action yielding the same reward is indistinguishable from the perspective of the feedback mechanism.

Realizability. Following the literature of realizable contextual MDP (Levy et al., 2023; Levy and Mansour, 2023; Levy et al., 2024) and context-dependent IGL (Zhang et al., 2024b), we make the following realizability assumption on the expected reward given the context, state, and action, as well as a decoder for the realized reward.

Assumption 2.2 (Realizability)**.** The learner is given a function class $\mathcal{F} \subseteq [0,1]^{\mathcal{X}\times\mathcal{S}_H\times\mathcal{A}}$ and, for each $s \in \mathcal{S}_H$, a state-specific decoder class $\Phi_s \subseteq \{0,1\}^{\mathcal{X}\times\mathcal{Y}}$. Define the product class $\Phi \triangleq \prod_{s\in\mathcal{S}_H} \Phi_s$. Each $\phi \in \Phi$ is a tuple $(\phi_s)_{s\in\mathcal{S}_H}$ and induces a mapping $\phi : \mathcal{X} \times \mathcal{Y} \times \mathcal{S}_H \to \{0,1\}$ via $\phi(x, y, s) \triangleq \phi_s(x, y)$. We assume that there exist unknown $f^\star \in \mathcal{F}$ and $\phi^\star = (\phi^\star_s)_{s\in\mathcal{S}_H} \in \Phi$ such that for all $t \in [T]$, $\mathbb{E}[r_t | x_t, s_{t,H}, a_{t,H}] = f^\star(x_t, s_{t,H}, a_{t,H})$ and $\phi^\star(x_t, y_t, s_{t,H}) = r_t$.

Following (Zhang et al., 2024b), we assume that $\mathcal{F}$ and Φ are both finite function classes and denote $|\mathcal{F}|$ and $|\Phi|$ as the cardinality of $\mathcal{F}$ and Φ. Our results can be directly extended to broader function classes with infinite size, which will be discussed in later sections.

Identifiability for States. As pointed out by Xie et al. (2021); Maghakian et al. (2023); Xie et al. (2022); Zhang et al. (2024b), it is impossible to learn the optimal policy without breaking the symmetry between reward values of 0 and 1. To address this, we introduce the following assumption on the structure of the reward function:

Assumption 2.3. We assume that the true expected reward function $f^\star$ defined in Assumption 2.2 satisfies that: there exist known constants $M \in (0, \frac{K}{2})$, $c \in [0,1]$, and $\theta \in (0,1)$ such that given context x, each final-layer state $s \in \mathcal{S}_H$ falls into either one of the following two categories:

- **Heterogeneous state**: $\sum_{a\in[K]} f^\star(x, s, a) \le M < \frac{K}{2}$, $\max_{a\in[K]} f^\star(x, s, a) \ge \theta > 0$, with $\frac{\theta(K-M)}{M} \triangleq \sigma > 1$.
- **Homogeneous state**: for all $a \in [K]$, $f^\star(x, s, a) = c$.

This assumption generalizes the identifiability condition from the contextual bandit setting ($H = 1$) of Zhang et al. (2024b) to the episodic MDP setting. Specifically, the heterogeneous state satisfies the same sparsity and separation conditions as in Zhang et al. (2024b), ensuring sufficient information asymmetry to enable efficient learning. In contrast, the homogeneous state captures MDP-specific scenarios where actions become indistinguishable due to identical feedback. Note that these two definitions are mutually exclusive. If $c = 0$, a homogeneous state has $\max_a f^\star(x, s, a) = 0 < \theta$ and therefore cannot be heterogeneous. If $c > 0$ and a homogeneous state also satisfied the heterogeneous-state conditions, then $Kc \le M$ and $c \ge \theta$, which would imply $\frac{\theta(K-M)}{M} \le \frac{c(K-Kc)}{Kc} = 1 - c < 1$,

contradicting the heterogeneity condition $\sigma > 1$. The existence of homogeneous state is common in scenarios such as learning language model where the system may become stuck in "degenerate" states, generating uniformly poor responses regardless of the chosen actions.

One may also question why the constant c needs to be known for homogeneous states. However, we argue that this knowledge is necessary for learning the optimal policy because, within an homogeneous state, the feedback is identical across all actions, making the reward value indistinguishable from interaction data alone. Without knowing c, the learner cannot determine the relative value of reaching an homogeneous state versus a heterogeneous state, leading to an impossibility to learn the optimal policy. Practically, homogeneous states typically correspond to degenerate scenarios or failure modes, such as a conversational agent producing a generic "I don't know" response, where the designer can explicitly assign a default utility (e.g., $c = 0$) to penalize uninformative outcomes.

Goal. The goal of the learner is to design a sequence of policies $\{\pi_t\}_{t=1}^T$ that maximize the cumulative reward over T episodes. We measure the performance via the notion of regret defined as $\mathbf{Reg} \triangleq T \cdot V^\star - \mathbb{E}\left[\sum_{t=1}^T V(\pi_t)\right]$. Here, $V(\pi) \triangleq \mathbb{E}[f^\star(x, s_H, a_H)]$ denotes the value of policy π, where the expectation is taken over the random context and the trajectory $\{(s_h, a_h)\}_{h=1}^H$ induced by the policy π given the context x and the transition dynamics. $V^\star$ denotes the highest possible value achieved by any stochastic policy in the class $\Pi \triangleq \{\pi : \mathcal{X} \times \mathcal{S} \to \Delta_\mathcal{A}\}$, and we denote the corresponding optimal policy by $\pi^\star \triangleq \operatorname{argmax}_{\pi\in\Pi} V(\pi)$. The expectation in the definition of regret is taken over the internal randomness of the algorithm and the stochasticity of the algorithm's observations.

Other Notations. For all $s \in \mathcal{S}$, define $P^\pi(s)$ as the probability of reaching state s under policy π and define $p_s^\star = \max_{\pi\in\Pi} P^\pi(s)$ to be the maximum probability of reaching state s. Denote $\mathbf{1}$ to be the all-one vector in an appropriate dimension. Denote q_{unif} to be the uniform distribution over an appropriate action space. Throughout, we adopt the convention that $0/0 = 0$.

3 Reward Decoder Learning

In this section, we explain in detail the construction of our reward estimator, which transforms indirect feedback into a usable proxy for latent rewards. Our approach proceeds in three stages: first, we identify reachable terminal states to ensure valid data collection; second, we learn the inverse kinematic (posterior distribution) of actions at these states; and third, we construct a Lipschitz-continuous reward estimator that robustly handles both heterogeneous and homogeneous states.

3.1 Step 1: Reachable State Identification

The first challenge in learning the reward decoder is ensuring we can collect sufficient feedback data. If a state has intrinsically low maximum reaching probability $p_s^\star$, it will never be visited often enough to allow reliable estimation; at the same time, its contribution to the total regret is also inherently negligible. Therefore, our goal in this step is to identify the set of states that are reachable with sufficiently high probability.

We begin by learning a homing policy for each state $s \in \mathcal{S}_H$ to maximize visitation. Specifically, for each target state s, we define a dummy reward function that is 1 if the learner reaches s and 0 otherwise. While any RL algorithm that provides provable PAC or regret guarantees suffices, we employ Euler (Zanette and Brunskill, 2019) on this auxiliary MDP to learn a policy $\widehat{\pi}_s$. With the number of episodes specified in Lemma 3.1, we obtain a policy capable of visiting s near-optimally (up to error ε). The full procedure is outlined in Algorithm 2 and deferred to Appendix C.

Lemma 3.1. *Applying Algorithm 2 to each $s \in \mathcal{S}_H$ with failure probability $\delta/|\mathcal{S}_H|$ and $N = \widetilde{\mathcal{O}}\left(\frac{SKH}{\varepsilon^2} + \frac{S^2KH^4}{\varepsilon}\right)$ episodes guarantees that with probability at least $1-\delta$, the output policy set $\{\widehat{\pi}_s\}_{s\in\mathcal{S}_H}$ satisfies $P^{\widehat{\pi}_s}(s) \geq p_s^\star - \varepsilon$ for all $s \in \mathcal{S}_H$. We take $N \geq \log(2S/\delta)/(2\varepsilon^2)$ as well.*

Next, we quantify the reachability of each state using these learned policies. Since both the optimal reachability $p_s^\star$ and $P^{\widehat{\pi}_s}(s)$ are unknown, we must estimate them based on empirical data. Let $\widehat{p}_s$ denote the empirical visitation frequency of state s observed by running $\widehat{\pi}_s$ for N fresh episodes.

A standard Hoeffding inequality and a union bound guarantee that with probability at least $1-\delta$, $|\widehat{p}_s - P^{\widehat{\pi}_s}(s)| \le \sqrt{\frac{\log(2S/\delta)}{2N}}$ for all $s \in \mathcal{S}_H$. Combining this concentration bound with Lemma 3.1 allows us to reliably distinguish between states with high and low maximum reaching probability based solely on the observable empirical visitation frequency $\widehat{p}_s$.

Lemma 3.2. *Let $\beta = \sqrt{\frac{\log(2S/\delta)}{2N}} \le \varepsilon$. Then, with probability at least $1-2\delta$, for every $s \in \mathcal{S}_H$ and any $\tau \in (0,1)$: (i) if $\widehat{p}_s \ge \tau + \beta$, then $p^\star_s \ge P^{\widehat{\pi}_s}(s) \ge \tau$; (ii) if $\widehat{p}_s \le \tau - (\beta+\varepsilon)$, then $p^\star_s \le \tau$.*

Finally, based on these estimates, we filter the state space to restrict our attention to states that can be reached often enough for reliable learning. We define the set of reliably reachable states as:

$$\widehat{\mathcal{S}}_{H,\varepsilon} \triangleq \{s \in \mathcal{S}_H : \widehat{p}_s \ge 4\varepsilon\}. \tag{1}$$

On the event used in Lemma 3.2, any excluded state satisfies $p^\star_s \le P^{\widehat{\pi}_s}(s) + \varepsilon \le \widehat{p}_s + \beta + \varepsilon < 6\varepsilon$, since $\widehat{p}_s < 4\varepsilon$ and $\beta \le \varepsilon$. Consequently, omitting these states incurs negligible regret, allowing us to focus the subsequent reward decoding steps exclusively on $\widehat{\mathcal{S}}_{H,\varepsilon}$.

3.2 Step 2: Inverse Kinematic Learning

Having identified the set of reliably reachable states $\widehat{\mathcal{S}}_{H,\varepsilon}$, we proceed to the second component: inverse kinematic learning. Specifically, inspired by Zhang et al. (2024b), we adopt an *inverse kinematic* strategy, which infers the latent reward by analyzing the posterior distribution of actions conditioned on the context, state, and observed feedback.

We first characterize the theoretical properties of this posterior when actions are selected uniformly.

Lemma 3.3. *For any context $x \in \mathcal{X}$ and state $s \in \mathcal{S}_H$, suppose the action a is chosen uniformly from $[K]$. Let r denote the realized (binary) reward and y denote the feedback. If $0 < \sum_{a'=1}^K f^\star(x,s,a') < K$, then the posterior distribution of a conditioned on (x,y,s) is given almost surely by*

$$\mathsf{Prob}[a|x,y,s] = \frac{f^\star(x,s,a)\cdot\phi^\star(x,y,s)}{\sum_{a'=1}^K f^\star(x,s,a')} + \frac{(1-f^\star(x,s,a))\cdot(1-\phi^\star(x,y,s))}{K-\sum_{a'=1}^K f^\star(x,s,a')}.$$

If $\sum_{a'=1}^K f^\star(x,s,a') \in \{0,K\}$, this posterior is uniform almost surely.

The following lemma establishes how the true reward can be recovered from this posterior. Crucially, for heterogeneous states, the posterior deviates from uniformity in the ℓ_∞-norm, enabling reward inference. In contrast, for homogeneous states, the posterior collapses to the uniform distribution, signalling that the expected reward is the known constant c.

Lemma 3.4. *For any context $x \in \mathcal{X}$ and state $s \in \mathcal{S}_H$, define $h^\star(x,y,s) = \frac{1}{K}\mathbf{1}$ if $\sum_{a'=1}^K f^\star(x,s,a') \in \{0,K\}$. Otherwise, define its components by*

$$h^\star_a(x,y,s) \triangleq \frac{f^\star(x,s,a)\cdot\phi^\star(x,y,s)}{\sum_{a'=1}^K f^\star(x,s,a')} + \frac{(1-f^\star(x,s,a))(1-\phi^\star(x,y,s))}{K-\sum_{a'=1}^K f^\star(x,s,a')}. \tag{2}$$

For any action $a \in [K]$ and feedback y generated by the realized reward $r(x,s,a)$, if s is a heterogeneous state, then:

- $\left\|h^\star(x,y,s) - \frac{1}{K}\mathbf{1}\right\|_\infty \ge \kappa$, *where* $\kappa \triangleq \frac{K\theta - M}{K(K-M)}$;
- $r(x,s,a) = \phi^\star(x,y,s) \ge \mathbb{1}\{h^\star_a(x,y,s) \ge \frac{\theta}{M}\}$ *for all* $a \in [K]$;
- $r(x,s,a) = \phi^\star(x,y,s) = \mathbb{1}\{h^\star_a(x,y,s) \ge \frac{\theta}{M}\}$ *when* $a \in \operatorname{argmax}_{a' \in [K]} f^\star(x,s,a')$;

otherwise, $h^\star(x,y,s) = \frac{1}{K}\mathbf{1}$.

The proof is provided in Appendix C. This result generalizes the inverse-kinematics characterization of Lemma 2 in Zhang et al. (2024b) to the MDP setting. Specifically, when s is a heterogeneous state, $\mathbb{1}\{h^\star_a(x,y,s) \ge \frac{\theta}{M}\}$ serves as an *underestimator* of the true reward $r(x,s,a)$ and matches the reward of the optimal policy. This matches the main idea in the estimator construction in the

single-state case shown in Zhang et al. (2024b). However, unlike the single-state setting, here the posterior serves an additional purpose, which acts as a discriminator between state types. If the posterior is bounded away from uniformity, the state is identified as heterogeneous; conversely, if it is uniform, the state is then homogeneous.

With the theoretical characterization of $h^\star$ established, we now turn to estimating it from data. We focus exclusively on the set of reachable states $\widehat{\mathcal{S}}_{H,\varepsilon}$ identified in Step 1. For each $s \in \widehat{\mathcal{S}}_{H,\varepsilon}$, we employ the approximate homing policy $\widehat{\pi}_s$ learned in Step 1 to collect a dataset of context–action–signal tuples. Whenever $\widehat{\pi}_s$ reaches state s, we record the tuple (x, s, a, y). Note that according to the definition of the policies collected in Algorithm 2, action a is uniformly sampled from $[K]$. Since every $s \in \widehat{\mathcal{S}}_{H,\varepsilon}$ satisfies $\widehat{p}_s \geq 4\varepsilon$, Lemma 3.2 ensures the true reaching probability is at least 3ε. This guarantees that we can collect a dataset D_s of size N_0 efficiently by using $\widetilde{\mathcal{O}}(\frac{N_0}{\varepsilon})$ number of episodes. The procedure is shown in Algorithm 3 deferred to Appendix C.

Lemma 3.5. *With probability at least $1 - 4\delta$, Algorithm 3 stops after $\mathcal{O}\Big(\frac{S\,(N_0+\log(S/\delta))}{\varepsilon}\Big)$ episodes.*

Finally, we estimate the posterior $h^\star$ for each $s \in \widehat{\mathcal{S}}_{H,\varepsilon}$ using Empirical Risk Minimization (ERM) on the collected dataset D_s. For $f \in \mathcal{F}$ and $\phi_s \in \Phi_s$, let $Q_f(x, s) = \sum_{i=1}^{K} f(x, s, i)$ and define

$$h_{f,\phi_s,a}(x, y) = \begin{cases} \frac{1}{K}, & Q_f(x, s) \in \{0, K\}, \\ \frac{f(x,s,a)\phi_s(x,y)}{Q_f(x,s)} + \frac{(1-f(x,s,a))(1-\phi_s(x,y))}{K-Q_f(x,s)}, & 0 < Q_f(x, s) < K. \end{cases}$$

This defines a probability vector for every (x, y). We consider the hypothesis class $\mathcal{H}_s \triangleq \{h_{f,\phi_s} : f \in \mathcal{F}, \phi_s \in \Phi_s\}$. We write $|\mathcal{H}| \triangleq \max_{s\in\mathcal{S}_H} |\mathcal{H}_s| \leq |\mathcal{F}||\Phi|$. Since the true posterior $h^\star_s \in \mathcal{H}_s$ for all $s \in \mathcal{S}_H$ based on Eq. (2), we can learn the posterior distribution at each state using an empirical risk minimization (ERM) oracle with squared loss. Specifically, for each state $s \in \widehat{\mathcal{S}}_{H,\varepsilon}$, we estimate $h^\star_s$ using D_s, and the following lemma, adapted from Lemma 3 of Zhang et al. (2024b), shows that the ERM estimator achieves an excess risk of $\mathcal{O}(\log(|\mathcal{H}|)/|D_s|)$ with high probability.

Lemma 3.6 (Lemma 3 of Zhang et al. (2024b))**.** *For each $s \in \widehat{\mathcal{S}}_{H,\varepsilon}$, let $\widehat{h}_s$ be the ERM estimator defined as:*

$$\widehat{h}_s \triangleq \operatorname*{argmin}_{h_s\in\mathcal{H}_s} \sum_{(x_i,s,a_i,y_i)\in D_s} \big\|h_s(x_i, y_i) - e_{a_i}\big\|_2^2, \tag{3}$$

where $\{D_s\}_{s\in\widehat{\mathcal{S}}_{H,\varepsilon}}$ is collected by running Algorithm 3 with N_0. Then, with probability at least $1 - \delta$, for all $s \in \widehat{\mathcal{S}}_{H,\varepsilon}$, $\mathbb{E}\Big[\big\|\widehat{h}_s(x, y) - h^\star(x, y, s)\big\|_2^2\Big] \leq \mathcal{O}\Big(\frac{\log(S|\mathcal{H}|/\delta)}{N_0}\Big)$, where the expectation is taken over $x \sim \mathcal{D}$, $a \sim q_{\text{unif}}$, and the resulting feedback y conditioned on (x, s, a).

For notation convenience, we define the vector of estimators as $\widehat{h} \triangleq (\widehat{h}_s)_{s\in\widehat{\mathcal{S}}_{H,\varepsilon}}$, and denote $\widehat{h}(x, y, s) = \widehat{h}_s(x, y)$ for any $s \in \widehat{\mathcal{S}}_{H,\varepsilon}$.

3.3 Step 3: Lipschitz Reward Estimator Construction

In the final step, we construct a robust reward estimator using the learned posterior $\widehat{h}$ from Step 2. Based on the characterization in Lemma 3.4, a natural choice for the reward estimator would be the plug-in indicator function:

$$\widehat{r}(x, y, s, a) = c \cdot \mathbb{1}\Big\{\Big\|\widehat{h}(x, y, s) - \tfrac{1}{K}\mathbf{1}\Big\|_\infty \leq \nu\Big\} + \mathbb{1}\Big\{\Big\|\widehat{h}(x, y, s) - \tfrac{1}{K}\mathbf{1}\Big\|_\infty > \nu\Big\}\ \mathbb{1}\Big\{\widehat{h}_a(x, y, s) \geq \tfrac{\theta}{M}\Big\},$$

for certain small positive value ν. Intuitively, this estimator assigns the constant reward c if the state is identified as homogeneous when the posterior distribution is close to uniform, and otherwise applies the thresholding rule from Lemma 3.4 to recover the reward.

However, as observed in Zhang et al. (2024b), the indicator function is discontinuous and non-Lipschitz. Consequently, small estimation errors in $\widehat{h}$, which are inevitable even with the generalization guarantees of Lemma 3.6, can lead to huge errors in the predicted reward. To address this,

we generalize the Lipschitz surrogate construction of Zhang et al. (2024b) to our multi-step setting, ensuring a smooth transition between the homogeneous and heterogeneous state regimes.

Given any posterior distribution $v \in \Delta_{[K]}$, let $\Delta(v) \triangleq \|v - \frac{1}{K}\mathbf{1}\|_\infty$ denote its ℓ_∞-distance from uniformity. We define the Lipschitz surrogate estimator $J(v, a)$ as follows:

$$J(v,a) = \begin{cases} c, & \Delta(v) \le \kappa/2, \\ \dfrac{2c(\kappa - \Delta(v)) + (2\Delta(v) - \kappa) \cdot G\left(v_a, \frac{\theta}{M} - \xi, \xi\right)}{\kappa}, & \Delta(v) \in (\kappa/2, \kappa), \\ G\left(v_a, \frac{\theta}{M} - \xi, \xi\right), & \text{otherwise}, \end{cases} \tag{4}$$

where the smoothing parameter ξ is defined as $\xi = \frac{1}{2}(\frac{\theta}{M} - \frac{1}{K-M})$, and $G(\alpha, \beta, \lambda)$ is the ramp function defined in Zhang et al. (2024b): $G(\alpha, \beta, \lambda) \triangleq \frac{\alpha - \beta}{\lambda}\mathbb{1}\{\beta \le \alpha < \beta + \lambda\} + \mathbb{1}\{\alpha \ge \beta + \lambda\}$.

To give intuitions on the estimator defined in Eq. (4), we visualize its geometry in Figure 1 and explain each part below:

- **Homogeneous Region (Pink):** When the posterior is near-uniform ($\Delta(v) \le \kappa/2$), the state is identified as homogeneous. Here, the estimator is clamped to the constant c, represented by the flat section in the center.
- **Heterogeneous Region (Green):** When the posterior is distinct ($\Delta(v) \ge \kappa$), the state is identified as heterogeneous. In this region, the estimator follows the inverse-kinematics ramp function G (shown as the red dashed line), which itself is a linear interpolation connecting values 0 and 1.
- **Interpolation Region (Yellow):** The middle case in Eq. (4) (for $\Delta(v) \in (\kappa/2, \kappa)$) serves as a *Lipschitz bridge*. As seen in the figure, this region linearly interpolates between the constant c (from the pink region) and the value of G (from the green region). This ensures that $J(v, a)$ smoothly transitions between the two logic types, preserving Lipschitz continuity essential for the regret analysis.

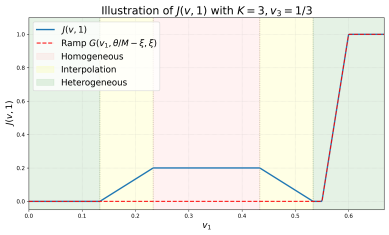


Figure 1: Illustration of Lipschitz reward decoder $J(v, 1)$ with $K = 3, M = 1$. The central region ($\Delta(v) \le \kappa/2$) corresponds to the homogeneous regime, while the edges represent the heterogeneous regime. The transition is smoothed via linear interpolation to maintain Lipschitz continuity.

The following lemma formally verifies that this construction preserves the key properties of the ideal estimator while ensuring the required smoothness.

Lemma 3.7. *For any context $x \in \mathcal{X}$, state $s \in \mathcal{S}_H$, action $a \in [K]$, and feedback y generated by a realized reward $r(x, s, a)$, the estimator J satisfies:*

- *If s is a heterogeneous state:*
  - $r(x, s, a) = \phi^\star(x, y, s) \ge J(h^\star(x, y, s), a)$;
  - $r(x, s, a) = \phi^\star(x, y, s) = J(h^\star(x, y, s), a)$ *if* $a \in \operatorname{argmax}_{a' \in [K]} f^\star(x, s, a')$.
- *If s is a homogeneous state:* $\mathbb{E}[r(x, s, a)] = J(h^\star(x, y, s), a) = c$.

Moreover, $J(v, a)$ is L-Lipschitz in v with respect to the ℓ_∞-norm, with $L = \frac{4}{\kappa} + \frac{1}{\xi}$, where κ and ξ are defined in Lemma 3.4 and Eq. (4) respectively.

The proof is deferred to Appendix C. A crucial consequence of Lemma 3.7 is that the Lipschitz continuity of $J(v, a)$ enables a stable transfer from true posterior $h^\star$ to its empirical estimate $\widehat{h}$, ensuring that $J(\widehat{h}(x, y, s), a)$ will be close to the true estimator $J(h^\star(x, y, s), a)$. In the next section, we leverage this reward estimator to design learning algorithms with provable performance guarantees.

4 Online Policy Learning

Building upon the reward estimator $J(\widehat{h}(x, y, s), a)$, we propose our algorithm for learning the optimal policy. Following the recent framework of realizable contextual learning (Foster and Rakhlin, 2020; Foster and Krishnamurthy, 2021), we assume access to an online square-loss regression oracle AlgSq: at round t, AlgSq predicts $\widehat{f}_t$ from the convex hull of $\mathcal{F}$, and then receives a tuple

Algorithm 1 IGL for Contextual MDPs

1 **Input:** horizon T, function classes $\mathcal{F}$, Φ, $\gamma > 0$, $\varepsilon > 0$, $N_0 > 0$, online regression oracle AlgSq.
2 Set $\delta = 1/(2T^2)$. All stages share the budget of T episodes; stop immediately when it is exhausted.
3 **for** *each* $s \in \mathcal{S}_H$ **do**
4 Run Algorithm 2 with failure probability $\delta/|\mathcal{S}_H|$, N defined in Lemma 3.1, and s, and output $\widehat{\pi}_s$.
5 Evaluate each $\widehat{\pi}_s$ on N fresh episodes to compute $\widehat{p}_s$, and construct $\widehat{\mathcal{S}}_{H,\varepsilon}$ by Eq. (1).
6 Run Algorithm 3 with input $\widehat{\mathcal{S}}_{H,\varepsilon}$, $\{\widehat{\pi}_s\}_{s\in\widehat{\mathcal{S}}_{H,\varepsilon}}$, and N_0, and obtain $\{D_s\}_{s\in\widehat{\mathcal{S}}_{H,\varepsilon}}$.
7 Compute $\widehat{h}_s$ for each $s \in \widehat{\mathcal{S}}_{H,\varepsilon}$ according to Eq. (3) using D_s. Let $\widehat{h} = (\widehat{h}_s)_{s\in\widehat{\mathcal{S}}_{H,\varepsilon}}$.
8 Run Algorithm 4 with input $\widehat{\mathcal{S}}_{H,\varepsilon}$, $\widehat{h}$, $\mathcal{F}$, γ, and AlgSq for the remaining episodes.

$(x_t, s_{t,H}, a_{t,H}, \widetilde{r}_t)$, where $\widetilde{r}_t \triangleq J(\widehat{h}(x_t, y_t, s_{t,H}), a_{t,H})$ serves as the estimated reward, and incurs a squared loss $(\widehat{f}_t(x_t, s_{t,H}, a_{t,H}) - \widetilde{r}_t)^2$. We assume that AlgSq satisfies certain regret bound with respect to squared-loss against the best predictor in $\mathcal{F}$.

Assumption 4.1 (Bounded squared-loss regret)**.** AlgSq guarantees that for any $n \le T$ and any sequence $\{(x_t, s_{t,H}, a_{t,H}, \widetilde{r}_t)\}_{t\in[n]}$, the following holds for a bound $\mathbf{Reg}_{\mathsf{Sq}}$: $\sum_{t=1}^n (\widehat{f}_t(x_t, s_{t,H}, a_{t,H}) - \widetilde{r}_t)^2 - \inf_{f\in\mathcal{F}} \sum_{t=1}^n (f(x_t, s_{t,H}, a_{t,H}) - \widetilde{r}_t)^2 \le \mathbf{Reg}_{\mathsf{Sq}}$.

Many standard function classes admit such an oracle. For instance, for a finite class $\mathcal{F}$, Vovk's aggregation algorithm achieves $\mathbf{Reg}_{\mathsf{Sq}} = \mathcal{O}(\log|\mathcal{F}|)$ (Vovk, 1995). We refer the readers to Foster and Rakhlin (2020) for more examples. We also define the reward estimator based on the ground-truth inverse kinematics $\underline{f}^\star(x, s, a) \triangleq \mathbb{E}_{y|x,s,a}[J(h^\star(x, y, s), a)]$. By Lemma 3.7, this function serves as a lower bound on the true reward $f^\star(x, s, a)$. Following Zhang et al. (2024b), we also assume $\underline{f}^\star \in \mathcal{F}$.

Assumption 4.2. Assume that $\underline{f}^\star \in \mathcal{F}$ where $\underline{f}^\star(x, s, a) \triangleq \mathbb{E}_{y|x,s,a}[J(h^\star(x, y, s), a)]$.

We now present our policy learning algorithm, Algorithm 4 (deferred to Appendix D.1), which builds upon the OMG-CMDP! algorithm proposed by Levy et al. (2023). Designed for realizable contextual MDPs with reward feedback, OMG-CMDP! operates by maintaining estimates for both reward and transition models via online regression oracles. Specifically, at each round t, the algorithm utilizes the history up to $t-1$ to approximate the rewards via a squared-loss oracle and dynamics via a log-loss oracle. Upon observing the current context x_t, it solves an optimization problem to find an occupancy measure that maximizes the estimated reward subject to the estimated dynamics, regularized by a log-barrier term. Finally, the learner executes the policy induced by this optimal occupancy measure.

Compared to OMG-CMDP!, our algorithm incorporates two key modifications tailored to our setting: (1) instead of using a general log-loss regression oracle to estimate the transition, since we assume that the transition is independent of the context, we employ a simpler *smoothed empirical mean estimator* constructed from online visitation counts to estimate the transition dynamics (Line 5). We show in Appendix D.2 that this estimator also enjoys a $\widetilde{\mathcal{O}}(S^2K)$ log-loss regret guarantee satisfying the requirements of the analysis in Levy et al. (2023). (2) Since the true reward is latent, we utilize the learned reward estimator $J(\widehat{h}(x, y, s), a)$ to generate proxy rewards and use them as the input for AlgSq. To ensure accurate reward estimation, we use the reachable set $\widehat{\mathcal{S}}_{H,\varepsilon}$ to filter the input tuples for AlgSq: if $s_{t,H} \in \widehat{\mathcal{S}}_{H,\varepsilon}$, we decode the feedback to obtain a proxy reward $\widetilde{r}_t = J(\widehat{h}(x_t, y_t, s_{t,H}), a_{t,H})$ and use the tuple $(x_t, s_{t,H}, a_{t,H}, \widetilde{r}_t)$ to update the oracle (Line 10 and Line 11); otherwise, we discard these episodes for reward learning. Since any policy visits states outside $\widehat{\mathcal{S}}_{H,\varepsilon}$ with probability at most $\mathcal{O}(S\varepsilon)$, ignoring them contributes at most $\mathcal{O}(TS\varepsilon)$ to regret. Our final full algorithm is shown in Algorithm 1, which combines decoder learning and online policy learning, and our main theoretical result is the following regret guarantee.

Theorem 4.3. *Under Assumptions 2.1, 2.2, 2.3, 4.1, and 4.2, suppose T is sufficiently large as specified in Appendix D.3. Then Algorithm 1, with appropriate choices of parameters guarantees* $\mathbf{Reg} = \widetilde{\mathcal{O}}\big(T^{\frac{3}{4}} S K^{\frac{1}{2}} H^{\frac{1}{4}} L^{\frac{1}{2}} + \sqrt{TSKH\mathbf{Reg}_{\mathsf{Sq}}}\big)$*, where L is defined in Lemma 3.7.*

To our knowledge, this is the first sublinear regret guarantee for contextual MDPs with implicit personalized feedback. Our bound has $\widetilde{\mathcal{O}}(T^{3/4} + \sqrt{T\mathbf{Reg}_{\mathsf{Sq}}})$ dependence on T, which is looser than

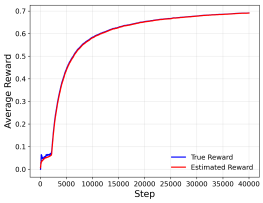
the single-step $\widetilde{\mathcal{O}}(T^{2/3} + \sqrt{T\mathbf{Reg}_{\mathsf{Sq}}})$ regret proven in Zhang et al. (2024b). This gap is caused by terminal states whose maximum reaching probability $p_s^\star$ is small, and it does not arise in the single-step setting, where the only state is reached with probability one. Compared to Levy et al. (2023) where direct reward feedback is available, our approach incurs an additional $\widetilde{\mathcal{O}}(T^{3/4})$ overhead to handle the signal decoding. We leave understanding its minimax regret bound as a future direction.

The full proof of this theorem is deferred to Appendix D.3. The primary analytical challenge lies in managing reward misspecification: our algorithm receives a learned proxy reward $J(\widehat{h}(x, y, s), a)$ rather than the true underlying reward. We address this by leveraging the Lipschitz continuity of J, allowing us to bound the misspecification via the estimation error between the learned posterior $\widehat{h}(x, y, s)$ and the true posterior $h^\star(x, y, s)$. This error is then controlled by the generalization guarantees of $\widehat{h}$, which rely on the dataset providing sufficient coverage of the action space.

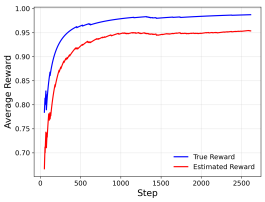
5 Experiment

5.1 Experiments on Synthetic Datasets

Dataset Construction. We construct a synthetic contextual episodic MDP with horizon $H = 3$ and a binary context space $x_t \in \{\texttt{True}, \texttt{False}\}$. At the start of each episode, the context is drawn with probabilities $\mathsf{Prob}(x_t = \texttt{True}) = 0.3$ and $\mathsf{Prob}(x_t = \texttt{False}) = 0.7$. The state space is partitioned into layers $\mathcal{S}_1 = \{s_{1,g}\}$, $\mathcal{S}_2 = \{s_{2,g}, s_{2,b}\}$, and $\mathcal{S}_3 = \{s_{3,g}, s_{3,b}\}$, where indices g and b represent "good" and "bad" states, respectively. The action set is $\mathcal{A} = \{a_1, \ldots, a_5\}$. More configuration details including the transition kernel, reward, and feedback signal setup, are deferred to Appendix E.1 due to space limit. Under this configuration, for either context, $s_{3,g}$ is heterogeneous with $\theta = 0.9$ and $M = 1.3$, while $s_{3,b}$ is homogeneous with constant reward $c = 0$.



(a) Synthetic MDP



(b) Real booking dataset

Figure 2: Avg. reward in policy learning.

Results Our empirical results are shown in Figure 2a, which plots the averaged cumulative reward over $T = 40000$ episodes of policy learning. The blue and red curves show the average true reward and decoded reward, respectively. Two observations are as follows. First, the decoded reward stays close to the true one but remains always slightly lower. This observation is consistent with the underestimation property of the ideal decoder in Lemma 3.4. Furthermore, the gap between the two curves decreases toward the end of training, indicating close agreement between the average decoded and true rewards. Second, as shown in Figure 2a, our algorithm steadily improves, approaching a 0.69 final reward (the optimal policy achieves reward 0.729), which shows that our method successfully optimizes the policy with only decoded signals.

5.2 Experiments on Real Datasets

Dataset Construction. We constructed our dataset by merging two task-oriented dialogue corpora: the `dialog_bAbI` dataset (Bordes et al., 2016) for restaurant booking and a movie booking dialogue corpus (Shah et al., 2018). The combined dataset consists of approximately 3,100 samples. We construct two-turn dialogues comprising a clarification query and a final booking action. Besides the optimal ground-truth actions, we employ an LLM to synthesize incorrect candidates including queries lacking sufficient detail and bookings containing inaccurate information. Generation details and the implementation details are deferred to Appendix E.2 and Appendix E.3 respectively.

Results. The performance on the real dataset is shown in Figure 2b. Note that the optimal reward is always 1, corresponding to the correct booking action. We observe that the average reward increases steadily during the training process, eventually reaching 0.98, which shows the effectiveness of our practical instantiation. Similar to Section 5.1, the average decoded reward remains below the average true reward throughout training in this experiment. Additional comparisons with uniform random and naive feedback baselines, together with ablations of Lipschitz smoothing, homogeneous state handling, and reachable state filtering, are provided in Appendix E.4 and Appendix E.5.

AI USE STATEMENT

We used generative AI tools to help create code and construct components of the dialogue-based experimental dataset, including incorrect candidate questions and booking actions, simulated user responses, and feedback, as described in Appendix E.2. Additionally, we used generative AI tools to assist with manuscript formatting. All mathematical proofs were developed and written by the authors. We take responsibility for the final content of this work, including all text, claims, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work studies learning from indirect, personalized feedback. Our experiments use a synthetic MDP and a dialogue-based booking dataset, with LLM-generated interaction components described in Appendix E.2. Such generated responses and feedback may reflect model biases and do not fully represent the diversity of human preferences. Accordingly, performance in these experiments should not be interpreted as evidence of effectiveness or fairness across real user populations. Applications involving actual users should account for privacy, informed consent, and the possibility that inferred rewards misrepresent user intentions. Personalized feedback should support user goals rather than facilitate manipulation or unwanted behavioral influence.

REPRODUCIBILITY STATEMENT

The problem formulation and assumptions are presented in Section 2, and the reward decoder construction and policy learning method are described in Section 3 and Section 4. Supporting algorithms and proofs are provided in Appendix C and Appendix D. Experimental configurations for the synthetic MDP are detailed in Appendix E.1. The dialogue dataset construction procedures, generation prompts, model configurations, and training settings are provided in Appendix E.2 and Appendix E.3. Additional baseline comparisons and ablation studies are presented in Appendix E.4 and Appendix E.5.

REFERENCES

Alekh Agarwal, Miroslav Dudík, Satyen Kale, John Langford, and Robert Schapire. Contextual bandit learning with predictable rewards. In *Artificial Intelligence and Statistics*, pages 19–26. PMLR, 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 *International Conference on Machine Learning*, pages 1638–1646. PMLR, 2014.

Iretiayo Akinola, Zizhao Wang, Junyao Shi, Xiaomin He, Pawan Lapborisuth, Jingxi Xu, David Watkins-Valls, Paul Sajda, and Peter Allen. Accelerated robot learning via human brain signals. In *2020 IEEE international conference on robotics and automation (ICRA)*, pages 3799–3805. IEEE, 2020.

Gábor Bartók and Csaba Szepesvári. Partial monitoring with side information. In *International Conference on Algorithmic Learning Theory*, pages 305–319. Springer, 2012.

Antoine Bordes, Y-Lan Boureau, and Jason Weston. Learning end-to-end goal-oriented dialog. *arXiv preprint arXiv:1605.07683*, 2016.

Dylan Foster and Alexander Rakhlin. Beyond ucb: Optimal and efficient contextual bandits with regression oracles. In *International Conference on Machine Learning*, pages 3199–3210. PMLR, 2020.

Dylan Foster, Alekh Agarwal, Miroslav Dudik, Haipeng Luo, and Robert Schapire. Practical contextual bandits with regression oracles. In *International Conference on Machine Learning*, pages 1539–1548. PMLR, 2018.

Dylan J Foster and Akshay Krishnamurthy. Contextual bandits with surrogate losses: Margin bounds and efficient algorithms. *Advances in Neural Information Processing Systems*, 31, 2018.

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.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Emrul Hasan, Mizanur Rahman, Chen Ding, Jimmy Xiangji Huang, and Shaina Raza. Review-based recommender systems: A survey of approaches, challenges and future perspectives. *ACM Computing Surveys*, 58(1):1–41, 2025.

Xiaoyan Hu, Farzan Farnia, and Ho-fung Leung. An information theoretic approach to interaction-grounded learning. *arXiv preprint arXiv:2401.05015*, 2024.

Chi Jin, Tiancheng Jin, Haipeng Luo, Suvrit Sra, and Tiancheng Yu. Learning adversarial markov decision processes with bandit feedback and unknown transition. In *International Conference on Machine Learning*, pages 4860–4869. PMLR, 2020a.

Chi Jin, Akshay Krishnamurthy, Max Simchowitz, and Tiancheng Yu. Reward-free exploration for reinforcement learning. In *International Conference on Machine Learning*, pages 4870–4879. PMLR, 2020b.

Hyunwoo Kim, Yoonseo Choi, Taehyun Yang, Honggu Lee, Chaneon Park, Yongju Lee, Jin Young Kim, and Juho Kim. Using LLMs to investigate correlations of conversational follow-up queries with user satisfaction. *arXiv preprint arXiv:2407.13166*, 2024.

Johannes Kirschner, Tor Lattimore, and Andreas Krause. Information directed sampling for linear partial monitoring. In *Proceedings of Thirty Third Conference on Learning Theory*, pages 2328–2369, 2020.

John Langford and Tong Zhang. The epoch-greedy algorithm for multi-armed bandits with side information. *Advances in neural information processing systems*, 20, 2007.

Orin Levy and Yishay Mansour. Learning efficiently function approximation for contextual mdp. *arXiv preprint arXiv:2203.00995*, 2022.

Orin Levy and Yishay Mansour. Optimism in face of a context: Regret guarantees for stochastic contextual mdp. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 8510–8517, 2023.

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, Asaf Cassel, Alon Cohen, and Yishay Mansour. Eluder-based regret for stochastic contextual MDPs. In *Forty-first International Conference on Machine Learning*, 2024.

Ying-Chun Lin, Jennifer Neville, Jack W. Stokes, Longqi Yang, et al. Interpretable user satisfaction estimation for conversational systems with large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL)*, 2024.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Jessica Maghakian, Paul Mineiro, Kishan Panaganti, Mark Rucker, Akanksha Saran, and Cheng Tan. Personalized reward learning with interaction-grounded learning (IGL). In *The Eleventh International Conference on Learning Representations*, 2023.

Nicola Neophytou, Bhaskar Mitra, and Catherine Stinson. Revisiting popularity and demographic biases in recommender evaluation and effectiveness. In *European Conference on Information Retrieval*, pages 641–654. Springer, 2022.

Gergely Neu, Andras Gyorgy, and Csaba Szepesvári. The adversarial stochastic shortest path problem with unknown transition probabilities. In *Artificial Intelligence and Statistics*, pages 805–813. PMLR, 2012.

Aviv Rosenberg and Yishay Mansour. Online convex optimization in adversarial markov decision processes. In *International Conference on Machine Learning*, pages 5478–5486. PMLR, 2019.

Pararth Shah, Dilek Hakkani-Tür, Gokhan Tür, Abhinav Rastogi, Ankur Bapna, Neha Nayak, and Larry Heck. Building a conversational agent overnight with dialogue self-play. *arXiv preprint arXiv:1801.04871*, 2018.

David Simchi-Levi and Yunzong Xu. Bypassing the monster: A faster and simpler optimal algorithm for contextual bandits under realizability. *Mathematics of Operations Research*, 2021.

Qwen Team et al. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*, 2(3), 2024.

Vladimir G Vovk. A game of prediction with expert advice. In *Proceedings of the eighth annual conference on Computational learning theory*, pages 51–60, 1995.

Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3):229–256, 1992.

Tengyang Xie, John Langford, Paul Mineiro, and Ida Momennejad. Interaction-grounded learning. In *International Conference on Machine Learning*, pages 11414–11423. PMLR, 2021.

Tengyang Xie, Akanksha Saran, Dylan J Foster, Lekan Molu, Ida Momennejad, Nan Jiang, Paul Mineiro, and John Langford. Interaction-grounded learning with action-inclusive feedback. *Advances in Neural Information Processing Systems*, 35:12529–12541, 2022.

Wanqiao Xu, Allen Nie, Ruijie Zheng, Aditya Modi, Adith Swaminathan, and Ching-An Cheng. Provably learning from language feedback. *arXiv preprint arXiv:2506.10341*, 2025.

Andrea Zanette and Emma Brunskill. Tighter problem-dependent regret bounds in reinforcement learning without domain knowledge using value function bounds. In *Proceedings of the 36th International Conference on Machine Learning*, pages 7304–7312, 2019.

Mengxiao Zhang, Yuheng Zhang, Haipeng Luo, and Paul Mineiro. Efficient contextual bandits with uninformed feedback graphs. *International Conference on Machine Learning*, 2024a.

Mengxiao Zhang, Yuheng Zhang, Haipeng Luo, and Paul Mineiro. Provably efficient interactive-grounded learning with personalized reward. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024b.

Mengxiao Zhang, Yuheng Zhang, Olga Vrousgou, Haipeng Luo, and Paul Mineiro. Practical contextual bandits with feedback graphs. *Advances in Neural Information Processing Systems*, 36, 2024c.

Alexander Zimin and Gergely Neu. Online learning in episodic markovian decision processes by relative entropy policy search. *Advances in neural information processing systems*, 26, 2013.

A Conclusion and Future Works

In this paper, we studied Interaction-Grounded Learning with personalized feedback in contextual episodic MDPs with terminal rewards. We proposed a two-stage algorithm that first learns a reward decoder via inverse kinematics, handling both significant and ambiguous states, and subsequently learns the optimal policy using these decoded signals. We established theoretical regret guarantees, and our experiments on both synthetic and real-world data validate that our decoder accurately recovers ground-truth rewards, enabling near-optimal policy learning without direct supervision. Future work includes extending our framework to context-dependent dynamics and generalizing the approach to handle intermediate personalized feedback.

B Related Works

Interaction-Grounded Learning and Learning from Implicit Feedback The study of Interaction-Grounded Learning was initiated by Xie et al. (2021), who modeled feedback as being conditionally independent of both context and action given the underlying latent reward. This framework was subsequently extended by Xie et al. (2022) to allow feedback to depend on the chosen action, while remaining independent of the context conditioned on the reward and action. Hu et al. (2024) later adopted the original formulation of Xie et al. (2021), introducing an information-theoretic mechanism to explicitly enforce the required conditional independence. Addressing the need for personalization, Maghakian et al. (2023) empirically extended the framework to incorporate user-specific reward structures, while Zhang et al. (2024b) developed the first computationally efficient algorithm with provable sublinear regret in this personalized setting. More recently, Xu et al. (2025) investigated learning with natural language feedback, establishing regret guarantees assuming access to a reliable signal-verification oracle. However, to the best of our knowledge, no prior work addresses the challenge of interaction-grounded learning with personalized feedback in the multi-step MDP setting.

Contextual Bandits and MDPs. Our work also connects to a broad literature on statistically and computationally efficient contextual learning. This includes classical contextual bandits (Langford and Zhang, 2007; Agarwal et al., 2012; 2014; Foster and Krishnamurthy, 2018; Foster et al., 2018; Foster and Rakhlin, 2020; Simchi-Levi and Xu, 2021), where the learner directly observes the reward for the selected action. Extensions of this standard model consider structured observations, such as contextual bandits with feedback graphs (Zhang et al., 2024a;c), where reward visibility is governed by an underlying graph structure, and contextual partial monitoring (Bartók and Szepesvári, 2012; Kirschner et al., 2020), where feedback is provided via a signal matrix or linear operator. Most relevant to our setting is the recent line of work on contextual MDPs (Levy and Mansour, 2022; 2023; Levy et al., 2023; 2024), which studies learning with general function approximation under stochastic or adversarial contexts. However, unlike our work, all these approaches assume access to direct reward observations.

C Omitted Details in Section 3

In this section, we present the omitted details in Section 3, including omitted algorithm descriptions in Appendix C.1 and omitted proofs in Appendix C.2.

C.1 Omitted Algorithms in Section 3

We first show the pseudo code that computes a policy that approximates the homing policy to a given state $s \in \mathcal{S}_H$ introduced in Section 3.1.

Algorithm 2 Homing policy learning for state s

Input: failure probability $\delta \in (0,1)$, number of episodes N, target state $s \in \mathcal{S}_H$.
Define a reward function $R_s : \mathcal{S} \to \{0,1\}$ by $R_s(u) = \mathbb{1}\{u = s\}$, independently of the action.
Run EULER on the induced episodic MDP with reward function R_s, horizon H, and failure probability δ for N episodes, ignoring the contexts.
Let $\Psi(s) = \{\pi^{(1)}, \pi^{(2)}, \ldots, \pi^{(N)}\}$ denote the set of policies executed by EULER during these episodes where we set the strategy of the final state s to be $\pi^{(n)}(s) = \frac{1}{K}\mathbf{1}$ for all $n \in [N]$.
Output. Output policy $\widehat{\pi}_s$ that, independently of the context, samples one policy uniformly from $\Psi(s)$ at the beginning of each episode and follows it for the entire episode.

Next, we show the pseudo code for collecting context-action-signal tuples introduced in Section 3.2.

The datasets retain repeated observations as separate samples.

C.2 Omitted Proofs in Section 3

In this section, we show the omitted proofs in Section 3. The first lemma shows that using Algorithm 2, we are able to obtain a policy whose probability of reaching s is close to the maximum achievable among all policies.

Algorithm 3 Tuple Collection

Input: State set $\widehat{\mathcal{S}}_{H,\varepsilon}$, policy set $\{\widehat{\pi}_s\}_{s\in\widehat{\mathcal{S}}_{H,\varepsilon}}$, sample size N_0.

for *each* $s \in \widehat{\mathcal{S}}_{H,\varepsilon}$ **do**
  Initialize $n = 1$ and dataset $D_s = \emptyset$.
  while $n \le N_0$ **do**
    Start a new episode and observe a context, denoted as $x_{s,n}$.
    Run policy $\widehat{\pi}_s$, and observe tuple $(s_n, a_{s,n}, y_{s,n})$ for the final state, action, and feedback for this episode.
    if $s_n = s$ **then**
      $D_s \leftarrow D_s \cup \{(x_{s,n}, s, a_{s,n}, y_{s,n})\}$, $n \leftarrow n+1$

Output: $D \triangleq \bigcup_{s\in\widehat{\mathcal{S}}_{H,\varepsilon}} D_s$.

Lemma 3.1. *Applying Algorithm 2 to each $s \in \mathcal{S}_H$ with failure probability $\delta/|\mathcal{S}_H|$ and $N = \widetilde{\mathcal{O}}\left(\frac{SKH}{\varepsilon^2} + \frac{S^2KH^4}{\varepsilon}\right)$ episodes guarantees that with probability at least $1-\delta$, the output policy set $\{\widehat{\pi}_s\}_{s\in\mathcal{S}_H}$ satisfies $P^{\widehat{\pi}_s}(s) \ge p_s^\star - \varepsilon$ for all $s \in \mathcal{S}_H$. We take $N \ge \log(2S/\delta)/(2\varepsilon^2)$ as well.*

Proof. Fix a target state $s \in \mathcal{S}_H$. According to the full value-dependent guarantee of EULER (Zanette and Brunskill, 2019) (see also Lemma 3.4 of Jin et al. (2020b)), with failure probability $\delta/|\mathcal{S}_H|$ and $N = \widetilde{\mathcal{O}}\left(\frac{SKH}{\varepsilon^2} + \frac{S^2KH^4}{\varepsilon}\right)$ rounds, the executed policies $\Psi(s) = \{\pi_{s,1}, \ldots, \pi_{s,N}\}$ satisfy

$$P^{\widehat{\pi}_s}(s) = \frac{1}{N}\sum_{n=1}^{N} P^{\pi_{s,n}}(s) \ge p_s^\star - \varepsilon.$$

The equality follows from sampling one policy at the beginning of each episode. A union bound over $s \in \mathcal{S}_H$ gives the simultaneous guarantee with probability at least $1-\delta$. □

The next lemma shows that we are able to reliably distinguish between states with high and low maximum reaching probability based solely on the observable empirical visitation frequency $\widehat{p}_s$ for each $s \in \mathcal{S}_H$.

Lemma 3.2. *Let $\beta = \sqrt{\frac{\log(2S/\delta)}{2N}} \le \varepsilon$. Then, with probability at least $1-2\delta$, for every $s \in \mathcal{S}_H$ and any $\tau \in (0,1)$: (i) if $\widehat{p}_s \ge \tau + \beta$, then $p_s^\star \ge P^{\widehat{\pi}_s}(s) \ge \tau$; (ii) if $\widehat{p}_s \le \tau - (\beta+\varepsilon)$, then $p_s^\star \le \tau$.*

Proof. Let $\mathcal{E}_1$ be the event from Lemma 3.1 that for all $s \in \mathcal{S}_H$, $P^{\widehat{\pi}_s}(s) \ge p_s^\star - \varepsilon$, so $\mathsf{Prob}(\mathcal{E}_1) \ge 1-\delta$. Let $\mathcal{E}_2$ be the event that for all $s \in \mathcal{S}_H$, $|\widehat{p}_s - P^{\widehat{\pi}_s}(s)| \le \beta$. Conditional on the learned policies, Hoeffding's inequality bounds each failure probability by $2\exp(-2N\beta^2) = \delta/S$. A union bound therefore gives $\mathsf{Prob}(\mathcal{E}_2) \ge 1-\delta$.

Suppose that $\mathcal{E}_1$ and $\mathcal{E}_2$ hold. If $\widehat{p}_s \ge \tau + \beta$, then $P^{\widehat{\pi}_s}(s) \ge \tau$ holds directly based on $\mathcal{E}_2$ and by definition of $p_s^\star$, we know that $p_s^\star \ge P^{\widehat{\pi}_s}(s) \ge \tau$. For (ii), if $\widehat{p}_s \le \tau - (\beta+\varepsilon)$, according to Lemma 3.1, we know that

$$p_s^\star \le P^{\widehat{\pi}_s}(s) + \varepsilon \le \widehat{p}_s + \varepsilon + \beta \le \tau,$$

where the first inequality is due to $\mathcal{E}_1$ and the second inequality is due to $\mathcal{E}_2$. □

Lemma 3.5. *With probability at least $1-4\delta$, Algorithm 3 stops after $\mathcal{O}\left(\frac{S\,(N_0+\log(S/\delta))}{\varepsilon}\right)$ episodes.*

Proof. Condition on the learned policies and the retained set, and suppose that the event in Lemma 3.2 holds. Every retained state s satisfies $P^{\widehat{\pi}_s}(s) \ge 3\varepsilon$, since $\widehat{p}_s \ge 4\varepsilon$ and $\beta \le \varepsilon$. For a sufficiently large absolute constant C, a Chernoff bound shows that $\lceil C(N_0 + \log(S/\delta))/\varepsilon \rceil$ independent executions of $\widehat{\pi}_s$ contain at least N_0 visits to s except with probability δ/S. A union bound over the retained states, together with the failure probability 2δ of the identification event, gives the claimed bound (in fact with failure probability at most 3δ).

Because each homing policy ignores the context and the transition kernel is independent of the context, the event of reaching s is independent of the current context. Conditional on the learned policies and retained set, the accepted tuples in D_s are therefore independent samples with $x \sim \mathcal{D}$, $a \sim q_{\text{unif}}$, and feedback drawn from its conditional distribution given (x, s, a). This is the distribution used in Lemma 3.6. □

The following lemma is an analog of Lemma 1 in Zhang et al. (2024b) in the context of MDP.

Lemma 3.3. *For any context $x \in \mathcal{X}$ and state $s \in \mathcal{S}_H$, suppose the action a is chosen uniformly from $[K]$. Let r denote the realized (binary) reward and y denote the feedback. If $0 < \sum_{a'=1}^{K} f^\star(x, s, a') < K$, then the posterior distribution of a conditioned on (x, y, s) is given almost surely by*

$$\mathsf{Prob}[a|x, y, s] = \frac{f^\star(x, s, a) \cdot \phi^\star(x, y, s)}{\sum_{a'=1}^{K} f^\star(x, s, a')} + \frac{(1 - f^\star(x, s, a)) \cdot (1 - \phi^\star(x, y, s))}{K - \sum_{a'=1}^{K} f^\star(x, s, a')}.$$

If $\sum_{a'=1}^{K} f^\star(x, s, a') \in \{0, K\}$, this posterior is uniform almost surely.

Proof. Suppose first that $0 < \sum_a f^\star(x, s, a) < K$. Bayes' rule gives the following identities almost surely, with conditional densities in place of probabilities when needed:

$$\begin{aligned}
&\mathsf{Prob}[a|x, y, s] \\
&= \frac{\mathsf{Prob}[a|x, s] \cdot \mathsf{Prob}[y|x, a, s]}{\mathsf{Prob}[y|x, s]} \\
&= \mathsf{Prob}[a|x, s] \frac{\mathsf{Prob}[r = 1|x, a, s] \cdot \mathsf{Prob}[y|x, a, s, r = 1] + \mathsf{Prob}[r = 0|x, a, s] \cdot \mathsf{Prob}[y|x, a, s, r = 0]}{\mathsf{Prob}[y|x, s]} \\
&= \mathsf{Prob}[a|x, s] \frac{f^\star(x, s, a) \cdot \mathsf{Prob}[y|x, s, r = 1] + (1 - f^\star(x, s, a)) \cdot \mathsf{Prob}[y|x, s, r = 0]}{\mathsf{Prob}[y|x, s]} \\
&= \mathsf{Prob}[a|x, s] \frac{f^\star(x, s, a) \cdot \mathsf{Prob}[r = 1|x, y, s]}{\mathsf{Prob}[r = 1|x, s]} + \mathsf{Prob}[a|x, s] \frac{(1 - f^\star(x, s, a)) \cdot \mathsf{Prob}[r = 0|x, y, s]}{\mathsf{Prob}[r = 0|x, s]} \\
&= \mathsf{Prob}[a|x, s] \frac{f^\star(x, s, a) \cdot \mathsf{Prob}[r = 1|x, y, s]}{\sum_{a'=1}^{K} \mathsf{Prob}[a'|x, s]\mathsf{Prob}[r = 1|x, s, a']} \\
&\quad + \mathsf{Prob}[a|x, s] \frac{(1 - f^\star(x, s, a)) \cdot \mathsf{Prob}[r = 0|x, y, s]}{\sum_{a'=1}^{K} \mathsf{Prob}[a'|x, s]\mathsf{Prob}[r = 0|x, s, a']} \\
&= \mathsf{Prob}[a|x, s] \frac{f^\star(x, s, a) \cdot \phi^\star(x, y, s)}{\sum_{a'=1}^{K} \mathsf{Prob}[a'|x, s] f^\star(x, s, a')} + \mathsf{Prob}[a|x, s] \frac{(1 - f^\star(x, s, a)) \cdot (1 - \phi^\star(x, y, s))}{\sum_{a'=1}^{K} \mathsf{Prob}[a'|x, s](1 - f^\star(x, s, a'))},
\end{aligned}$$

where the third equality uses the fact that y is independent of the action given x, s, and r. Since $\mathsf{Prob}[a|x, s] = \frac{1}{K}$ for all $a \in [K]$ given any $x \in \mathcal{X}$ and $s \in \mathcal{S}_H$, we know that

$$\mathsf{Prob}[a|x, y, s] = \frac{f^\star(x, s, a) \cdot \phi^\star(x, y, s)}{\sum_{a'=1}^{K} f^\star(x, s, a')} + \frac{(1 - f^\star(x, s, a))(1 - \phi^\star(x, y, s))}{K - \sum_{a'=1}^{K} f^\star(x, s, a')}.$$

If $\sum_a f^\star(x, s, a) \in \{0, K\}$, all actions have the same deterministic reward. Conditional independence of feedback then implies that y is independent of a given (x, s), so the posterior remains uniform almost surely. □

The next lemma shows how we are able to interpret the true reward from the ground-truth posterior for both homogeneous and heterogeneous states.

Lemma 3.4. *For any context $x \in \mathcal{X}$ and state $s \in \mathcal{S}_H$, define $h^\star(x, y, s) = \frac{1}{K}\mathbf{1}$ if $\sum_{a'=1}^{K} f^\star(x, s, a') \in \{0, K\}$. Otherwise, define its components by*

$$h_a^\star(x, y, s) \triangleq \frac{f^\star(x, s, a) \cdot \phi^\star(x, y, s)}{\sum_{a'=1}^{K} f^\star(x, s, a')} + \frac{(1 - f^\star(x, s, a))(1 - \phi^\star(x, y, s))}{K - \sum_{a'=1}^{K} f^\star(x, s, a')}. \tag{2}$$

For any action $a \in [K]$ and feedback y generated by the realized reward $r(x, s, a)$, if s is a heterogeneous state, then:

- $\left\|h^\star(x,y,s) - \frac{1}{K}\mathbf{1}\right\|_\infty \geq \kappa$, *where* $\kappa \triangleq \frac{K\theta - M}{K(K-M)}$;
- $r(x,s,a) = \phi^\star(x,y,s) \geq \mathbb{1}\left\{h_a^\star(x,y,s) \geq \frac{\theta}{M}\right\}$ *for all* $a \in [K]$;
- $r(x,s,a) = \phi^\star(x,y,s) = \mathbb{1}\left\{h_a^\star(x,y,s) \geq \frac{\theta}{M}\right\}$ *when* $a \in \operatorname{argmax}_{a' \in [K]} f^\star(x,s,a')$;

otherwise, $h^\star(x,y,s) = \frac{1}{K}\mathbf{1}$.

Proof. Fix any context $x \in \mathcal{X}$ and heterogeneous state $s \in \mathcal{S}_H$. For conciseness, we denote $f_a \triangleq f^\star(x,s,a)$, $Q \triangleq \sum_{a=1}^K f_a \leq M$, and $a^\star \in \arg\max_{a \in [K]} f_a$, so that $f_{a^\star} \geq \theta$. We lower bound $\left\|h^\star(x,y,s) - \frac{1}{K}\mathbf{1}\right\|_\infty$ in two cases.

Case 1: $\phi^\star(x,y,s) = 1$**.** By Eq. (2), we know that $h_a^\star(x,y,s) = \frac{f_a}{Q}$ for all $a \in [K]$. Moreover, according to Assumption 2.3, we know that $K\theta - M > K\theta - \sigma M = \theta M > 0$. Hence,

$$\left\|h^\star(x,y,s) - \tfrac{1}{K}\mathbf{1}\right\|_\infty \geq h_{a^\star}^\star(x,y,s) - \tfrac{1}{K} = \frac{f_{a^\star}}{Q} - \frac{1}{K} \geq \frac{\theta}{Q} - \frac{1}{K} \geq \frac{\theta}{M} - \frac{1}{K} = \frac{K\theta - M}{KM}.$$

Since $M < \frac{K}{2}$, we have $K - M > M$, and therefore $\frac{K\theta - M}{KM} \geq \frac{K\theta - M}{K(K-M)}$.

Case 2: $\phi^\star(x,y,s) = 0$**.** By Eq. (2), $h_a^\star(x,y,s) = \frac{1 - f_a}{K - Q}$ for all $a \in [K]$. In particular,

$$\left\|h^\star(x,y,s) - \tfrac{1}{K}\mathbf{1}\right\|_\infty \geq \left|\tfrac{1}{K} - h_{a^\star}^\star(x,y,s)\right| \geq \frac{1}{K} - \frac{1 - f_{a^\star}}{K - Q} \geq \frac{1}{K} - \frac{1-\theta}{K-Q} = \frac{K\theta - Q}{K(K-Q)}.$$

The function $u \mapsto \frac{K\theta - u}{K(K-u)}$ is decreasing for $u \in (0, K\theta)$ (since $\theta \leq 1$), and thus using $Q \leq M < K\theta$ yields

$$\frac{K\theta - Q}{K(K-Q)} \geq \frac{K\theta - M}{K(K-M)}.$$

Combining the two cases, we obtain

$$\left\|h^\star(x,y,s) - \tfrac{1}{K}\mathbf{1}\right\|_\infty \geq \frac{K\theta - M}{K(K-M)},$$

which proves the first statement for heterogeneous states.

The remaining two statements for heterogeneous states follow the proof of Lemma 2 in Zhang et al. (2024b). Specifically, if $h_a^\star(x,y,s) \geq \frac{\theta}{M}$, then we must have $\phi^\star(x,y,s) = 1$ because otherwise,

$$h_a^\star(x,y,s) = \frac{1 - f_a}{K - Q} \leq \frac{1}{K - M} < \frac{\theta}{M},$$

where the first inequality uses the fact that $Q \leq M$ and $f_a \geq 0$, and the last inequality is by Assumption 2.3. Moreover, for $a \in \operatorname{argmax}_{a' \in [K]} f^\star(x,s,a')$, we know that $f^\star(x,s,a) \geq \theta$. Therefore, when $\phi^\star(x,y,s) = 1$,

$$h_a^\star(x,y,s) = \frac{f^\star(x,s,a)}{\sum_{a'=1}^K f^\star(x,s,a')} \geq \frac{\theta}{M}.$$

Finally, suppose that s is a homogeneous state. For $c \in (0,1)$, Eq. (2) directly gives $h_a^\star(x,y,s) = 1/K$ for all $a \in [K]$. For $c \in \{0,1\}$, the same identity holds for every y by the endpoint definition of $h^\star$ in Lemma 3.4; it agrees almost surely with the true posterior by Lemma 3.3. Thus,

$$h^\star(x,y,s) = \frac{1}{K}\mathbf{1},$$

and consequently

$$\left\|h^\star(x,y,s) - \frac{1}{K}\mathbf{1}\right\|_\infty = 0.$$

□

Lemma 3.7. *For any context $x \in \mathcal{X}$, state $s \in \mathcal{S}_H$, action $a \in [K]$, and feedback y generated by a realized reward $r(x,s,a)$, the estimator J satisfies:*

- *If s is a heterogeneous state:*
  - $r(x,s,a) = \phi^\star(x,y,s) \geq J(h^\star(x,y,s),a)$;
  - $r(x,s,a) = \phi^\star(x,y,s) = J(h^\star(x,y,s),a)$ *if* $a \in \operatorname{argmax}_{a' \in [K]} f^\star(x,s,a')$.
- *If s is a homogeneous state:* $\mathbb{E}[r(x,s,a)] = J(h^\star(x,y,s),a) = c$.

Moreover, $J(v,a)$ is L-Lipschitz in v with respect to the ℓ_∞-norm, with $L = \frac{4}{\kappa} + \frac{1}{\xi}$, where κ and ξ are defined in Lemma 3.4 and Eq. (4) respectively.

Proof. The proof follows a similar spirit as the proof of Lemma 3.4. Fix any context $x \in \mathcal{X}$, state $s \in \mathcal{S}_H$, action $a \in [K]$, and feedback y. Recall $\Delta(v) = \|v - \frac{1}{K}\mathbf{1}\|_\infty$ and $\xi = \frac{1}{2}(\frac{\theta}{M} - \frac{1}{K-M})$.

When s is a heterogeneous state, by Lemma 3.4, we know that $\Delta(h^\star(x,y,s)) \geq \kappa$. Hence, by Eq. (4),

$$J(h^\star(x,y,s),a) = G\left(h_a^\star(x,y,s), \frac{\theta}{M} - \xi, \xi\right).$$

We then consider two cases.

Case 1: $\phi^\star(x,y,s) = 1$. Then $r(x,s,a) = \phi^\star(x,y,s) = 1$. Since $G\left(h_a^\star(x,y,s), \frac{\theta}{M} - \xi, \xi\right) \in [0,1]$, we know that

$$r(x,s,a) = 1 \geq J(h^\star(x,y,s),a).$$

Moreover, if $a \in \operatorname{argmax}_{a' \in [K]} f^\star(x,s,a')$, then Lemma 3.4 shows that $h_a^\star(x,y,s) \geq \frac{\theta}{M}$. This means that

$$\begin{aligned}
&G\left(h_a^\star(x,y,s), \frac{\theta}{M} - \xi, \xi\right) \\
&= \frac{h_a^\star(x,y,s) - \frac{\theta}{M} + \xi}{\xi} \cdot \mathbb{1}\left\{h_a^\star(x,y,s) \in \left[\frac{\theta}{M} - \xi, \frac{\theta}{M}\right)\right\} + \mathbb{1}\left\{h_a^\star(x,y,s) \geq \frac{\theta}{M}\right\} = 1,
\end{aligned}$$

and therefore $J(h^\star(x,y,s),a) = G\left(h_a^\star(x,y,s), \frac{\theta}{M} - \xi, \xi\right) = 1 = r(x,s,a)$.

Case 2: $\phi^\star(x,y,s) = 0$. Then $r(x,s,a) = 0$. By Eq. (2), we know that $h_a^\star(x,y,s) = \frac{1 - f^\star(x,s,a)}{K - \sum_{a'} f^\star(x,s,a')} \leq \frac{1}{K-M}$. Under Assumption 2.3, we have $\frac{1}{K-M} \leq \frac{\theta}{M} - \xi$, so $h_a^\star(x,y,s) \leq \frac{\theta}{M} - \xi$ and hence $G\left(h_a^\star(x,y,s), \frac{\theta}{M} - \xi, \xi\right) = 0$. Thus

$$J(h^\star(x,y,s),a) = G\left(h_a^\star(x,y,s), \frac{\theta}{M} - \xi, \xi\right) = 0 = r(x,s,a).$$

Combining both cases proves the first statement for heterogeneous states. When $a \in \operatorname{argmax}_{a' \in [K]} f^\star(x,s,a')$, we only need to show that when $\phi^\star(x,y,s) = 1$, $J(h^\star(x,y,s),a) = 1$. Since $f^\star(x,s,a) \geq \theta$ according to Assumption 2.3, we know that

$$h_a^\star(x,y,s) = \frac{f^\star(x,s,a)}{\sum_{a'=1}^K f^\star(x,s,a')} \geq \frac{\theta}{M},$$

meaning that

$$J(h^\star(x,y,s),a) = G\left(h_a^\star(x,y,s), \frac{\theta}{M} - \xi, \xi\right) = 1 = \phi^\star(x,y,s).$$

When s is a homogeneous state, then by Lemma 3.4, we know that $h^\star(x,y,s) = \frac{1}{K}\mathbf{1}$. Hence, $\Delta(h^\star(x,y,s)) = 0 \leq \kappa/2$. Therefore by Eq. (4), we know that $J(h^\star(x,y,s),a) = c$, which equals to $\mathbb{E}[r(x,s,a)]$ by the definition of homogeneous states.

To show that $J(v,a)$ is Lipschitz in v with respect to ℓ_∞ norm, we fix any $a \in [K]$ and let $v, v' \in \Delta_K$. By the triangular inequality, we know that

$$|\Delta(v) - \Delta(v')| \leq \|v - v'\|_\infty. \tag{5}$$

When $\Delta(v)$ and $\Delta(v')$ are both in $[\frac{\kappa}{2}, \kappa)$, we can write

$$J(v,a) = \mu_1(\Delta(v)) \cdot c + \mu_2(\Delta(v)) \cdot G\left(v_a, \frac{\theta}{M} - \xi, \xi\right),$$

where $\mu_1(u) = \frac{2(\kappa-u)}{\kappa}, \mu_2(u) = \frac{2u-\kappa}{\kappa}$. Then for any $u, u' \in [\kappa/2, \kappa]$,

$$|\mu_1(u) - \mu_1(u')| = |\mu_2(u) - \mu_2(u')| \le \frac{2}{\kappa}|u - u'|. \tag{6}$$

Therefore, direct calculation shows that

$$\begin{aligned}
|J(v,a) - J(v',a)| &\le c \cdot |\mu_1(\Delta(v)) - \mu_1(\Delta(v'))| + G\left(v_a, \frac{\theta}{M} - \xi, \xi\right) \cdot |\mu_2(\Delta(v)) - \mu_2(\Delta(v'))| \\
&\quad + \mu_2(\Delta(v'))\left|G\left(v_a, \frac{\theta}{M} - \xi, \xi\right) - G\left(v'_a, \frac{\theta}{M} - \xi, \xi\right)\right| \\
&\le \frac{2c\|v - v'\|_\infty}{\kappa} + \frac{2\|v - v'\|_\infty}{\kappa} + \frac{\|v - v'\|_\infty}{\xi} \\
&\le \left(\frac{4}{\kappa} + \frac{1}{\xi}\right)\|v - v'\|_\infty, \qquad \text{(since } |c| \le 1\text{)}
\end{aligned}$$

where the second inequality uses Eq. (5), Eq. (6), and the fact that

$$\left|G\left(v_a, \tfrac{\theta}{M} - \xi, \xi\right) - G\left(v'_a, \tfrac{\theta}{M} - \xi, \xi\right)\right| \le \frac{1}{\xi}|v_a - v'_a| \le \frac{\|v - v'\|_\infty}{\xi}. \tag{7}$$

Since on the other two regions, $J(v,a)$ is either constant when $\Delta(v) \le \kappa/2$ or equals $G(v_a, \frac{\theta}{M} - \xi, \xi)$ when $\Delta(v) \ge \kappa$, $J(v,a)$ is also $\frac{1}{\xi}$ in the other two regions. Finally, since J is continuous at $\Delta = \kappa/2$ and $\Delta = \kappa$, we know that $J(v,a)$ is Lipschitz over the whole domain with Lipschitz constant $L = \max\{\frac{4}{\kappa} + \frac{1}{\xi}, \frac{1}{\xi}\} = \frac{4}{\kappa} + \frac{1}{\xi}$. □

D Omitted Details in Section 4

In this section, we provide omitted details in Section 4, including the algorithm for policy learning in Appendix D.1, the log-loss regression regret guarantees for empirical transition estimation in Appendix D.2, and the proof of our main theorem Theorem 4.3 in Appendix D.3.

D.1 Policy Learning Algorithm

In this subsection, we provide the algorithm description of IGL policy learning for contextual MDPs. The pseudo code is shown in Algorithm 4, with a budget T_{online} equal to the number of episodes remaining after exploration. Episode indices in this algorithm are local to the online phase; in the main regret proof, they are shifted to the corresponding global indices.

D.2 Guarantee for Empirical Transition Estimation

In this subsection, we provide the performance guarantee for estimating the transition kernel using smoothed empirical mean shown in Line 5 of Algorithm 4. Specifically, we show that this smoothed empirical mean enjoys a $\widetilde{\mathcal{O}}(S^2K)$ log-loss regression regret guarantee, which satisfies the requirements of the analysis in Levy et al. (2023).

Theorem D.1 (Log-loss guarantee for Laplace-smoothed transition estimator)**.** *Let $\{\widehat{P}_t\}_{t=1}^T$ be the sequence of Laplace-smoothed estimators defined in Line 5 and recall that $\mathcal{P}$ is the true transition kernel. Then, the cumulative log-loss regression regret with respect to $\mathcal{P}$ is bounded by:*

$$\sum_{t=1}^{T}\sum_{h=1}^{H-1}\left(-\log \widehat{P}_t(s_{t,h+1} \mid s_{t,h}, a_{t,h})\right) - \sum_{t=1}^{T}\sum_{h=1}^{H-1}\left(-\log \mathcal{P}(s_{t,h+1} \mid s_{t,h}, a_{t,h})\right) \le S^2K\log(TH+S),$$

Algorithm 4 IGL Policy Learning for Contextual MDPs

Input: episode budget T_{online}, state set $\widehat{\mathcal{S}}_{H,\varepsilon}$, decoder $\widehat{h}$, function classes $\mathcal{F}$, $\gamma > 0$, online regression oracle AlgSq.
Initialize: $N_1(s,a) \leftarrow 0$ and $N_1(s,a,s') \leftarrow 0$ for all $h \in [H-1]$, $(s,a) \in \mathcal{S}_h \times \mathcal{A}$, $s' \in \mathcal{S}_{h+1}$.
for $t = 1, \dots, T_{\text{online}}$ **do**
  Receive context x_t and $\widehat{f}_t$ from AlgSq.
  Estimate the transition as $\widehat{P}_t(s' \mid s,a) = \frac{N_t(s,a,s')+1}{N_t(s,a)+|\mathcal{S}_{h+1}|}$ for all $h \in [H-1]$, $(s,a) \in \mathcal{S}_h \times \mathcal{A}$, $s' \in \mathcal{S}_{h+1}$.
  Solve for occupancy measure $\widehat{q}_t$:

$$\widehat{q}_t = \operatorname*{argmax}_{q \in \mu(\widehat{P}_t)} \Big\{ \sum_{s \in \widehat{\mathcal{S}}_{H,\varepsilon},\, a \in \mathcal{A}} q_H(s,a) \widehat{f}_t(x_t,s,a) + \frac{1}{\gamma} \sum_{h,s,a} \log q_h(s,a) \Big\},$$

  where $\mu(P)$ is the set of valid occupancy measures under transition kernel P, and the second sum ranges over $h \in [H]$, $s \in \mathcal{S}_h$, and $a \in \mathcal{A}$.
  For all $h \in [H]$ and $(s,a) \in \mathcal{S}_h \times \mathcal{A}$, set policy as:

$$\pi_t(a \mid s) = \frac{\widehat{q}_{t,h}(s,a)}{\sum_{a'=1}^{K} \widehat{q}_{t,h}(s,a')}.$$

  Execute π_t, get trajectory $\tau_t = (s_{t,1}, a_{t,1}, s_{t,2}, \dots, s_{t,H}, a_{t,H})$ and feedback y_t.
  if $s_{t,H} \in \widehat{\mathcal{S}}_{H,\varepsilon}$ **then**
    $\widetilde{r}_t = J(\widehat{h}(x_t, y_t, s_{t,H}), a_{t,H})$.
    Update AlgSq with $(x_t, s_{t,H}, a_{t,H}, \widetilde{r}_t)$.
  for $h = 1, 2, \dots, H-1$ **do**
    for *each* $(s,a,s') \in \mathcal{S}_h \times \mathcal{A} \times \mathcal{S}_{h+1}$ **do**

$$N_{t+1}(s,a) = N_t(s,a) + \mathbb{1}\{s = s_{t,h}, a = a_{t,h}\}, \tag{8}$$
$$N_{t+1}(s,a,s') = N_t(s,a,s') + \mathbb{1}\{s = s_{t,h}, a = a_{t,h}, s' = s_{t,h+1}\}. \tag{9}$$

Before proving Theorem D.1, we first provide two helpful auxiliary lemmas.

Lemma D.2. *Fix a layer $h \in [H-1]$ and state-action pair $(s,a) \in \mathcal{S}_h \times \mathcal{A}$. Consider the subsequence of global time indices $t_1, \dots, t_B \in [T]$ such that $(s_{t_b,h}, a_{t_b,h}) = (s,a)$ for all $b \in [B]$. Let $z_b \triangleq s_{t_b,h+1} \in \mathcal{S}_{h+1}$ be the next state observed at time t_b. Let $n_{s'} \triangleq \sum_{b=1}^{B} \mathbb{1}\{z_b = s'\}$ count occurrences of $s' \in \mathcal{S}_{h+1}$ in this subsequence. Then the product of the sequential predictive probabilities satisfies:*

$$\prod_{b=1}^{B} \widehat{P}_{t_b}(z_b \mid s,a) = \frac{(S_{h+1}-1)! \prod_{s' \in \mathcal{S}_{h+1}} n_{s'}!}{(S_{h+1}+B-1)!},$$

where we denote $|\mathcal{S}_h| = S_h$ for all $h \in [H]$.

Proof. Recall the estimator definition: $\widehat{P}_{t_b}(z_b \mid s,a) = \frac{N_{t_b}(s,a,z_b)+1}{N_{t_b}(s,a)+S_{h+1}}$. We analyze the numerators and denominators of $\prod_{b=1}^{B} \widehat{P}_{t_b}(z_b \mid s,a)$ separately.

Denominator. Note that $N_{t_b}(s,a)$ counts the visitations to (s,a) prior to time t_b. Thus, as b goes from 1 to B, $N_{t_b}(s,a)$ takes values $0, 1, \dots, B-1$. The product of denominators is:

$$\prod_{b=1}^{B} (N_{t_b}(s,a) + S_{h+1}) = \prod_{k=0}^{B-1} (S_{h+1}+k) = \frac{(S_{h+1}+B-1)!}{(S_{h+1}-1)!}.$$

Numerator. We regroup the product terms based on the outcome state s'. For a fixed state $s' \in \mathcal{S}_{h+1}$, the term $(N_{t_b}(s,a,s')+1)$ appears exactly $n_{s'}$ times. Across these occurrences, the count $N_{t_b}(s,a,s')$

increments sequentially from 0 to $n_{s'} - 1$. Thus, the contribution from state s' is:

$$\prod_{k=0}^{n_{s'}-1} (k+1) = n_{s'}!.$$

Taking the product over all $s' \in \mathcal{S}_{h+1}$ yields $\prod_{s' \in \mathcal{S}_{h+1}} n_{s'}!$.

Combining the calculation of the numerator and the denominator concludes the proof. □

Lemma D.3. *Fix any $h \in [H-1]$ and let $S_{h+1} := |\mathcal{S}_{h+1}|$. Let $\{n_{s'}\}_{s' \in \mathcal{S}_{h+1}}$ be a set of non-negative integers such that $\sum_{s'} n_{s'} = B \geq 1$. Define the distribution $q \in \Delta_{\mathcal{S}_{h+1}}$ such that $q(s') \triangleq n_{s'}/B$ for all $s' \in \mathcal{S}_{h+1}$. With the convention $0 \log 0 = 0$, the following inequality holds:*

$$\log \frac{(S_{h+1}+B-1)!}{(S_{h+1}-1)! \prod_{s'} n_{s'}!} \;\leq\; -B \sum_{s' \in \mathcal{S}_{h+1}} q(s') \log q(s') + (S_{h+1}-1)\log(S_{h+1}+B).$$

Proof. We decompose the term inside the logarithm into the product of a multinomial coefficient and a binomial coefficient:

$$\frac{(S_{h+1}+B-1)!}{(S_{h+1}-1)! \prod_{s'} n_{s'}!} = \underbrace{\frac{B!}{\prod_{s'} n_{s'}!}}_{\text{TERM-A}} \cdot \underbrace{\binom{S_{h+1}+B-1}{S_{h+1}-1}}_{\text{TERM-B}}.$$

Bounding TERM-A. Consider the multinomial expansion of $(\sum_{s'} x_{s'})^B$ with $x_{s'} = q(s') = n_{s'}/B$. Since $\sum_{s'} q(s') = 1$, we have:

$$1 = \left(\sum_{s'} q(s')\right)^B = \sum_{k_1 + \cdots + k_{S_{h+1}} = B} \frac{B!}{\prod_{s'} k_{s'}!} \prod_{s'} q(s')^{k_{s'}}.$$

This sum is lower bounded by the single term where the exponents match the counts (i.e., $k_{s'} = n_{s'}$):

$$1 \geq \frac{B!}{\prod_{s'} n_{s'}!} \prod_{s'} q(s')^{n_{s'}}.$$

Taking the logarithm and rearranging terms yields:

$$\log(\text{TERM-A}) \leq -\sum_{s'} n_{s'} \log q(s') = B \sum_{s'} -\frac{n_{s'}}{B} \log q(s') = -B \sum_{s'} q(s') \log q(s').$$

Bounding TERM-B. Using the inequality $\binom{n}{k} \leq (n+1)^k$:

$$\binom{S_{h+1}+B-1}{S_{h+1}-1} \leq (S_{h+1}+B)^{S_{h+1}-1}.$$

Taking the logarithm gives $\log(\text{TERM-B}) \leq (S_{h+1}-1)\log(S_{h+1}+B)$. Summing the bounds for TERM-A and TERM-B concludes the proof. □

With these technical lemmas established, we are now ready to present the main theorem regarding the log-loss regret of our estimated transition model.

Proof of Theorem D.1. We partition the transitions according to the state-action pairs. Fix a layer $h \in [H-1]$ and a state-action pair $(s,a) \in \mathcal{S}_h \times \mathcal{A}$. Let $t_1, \ldots, t_{m_{s,a}}$ be the subsequence of episode indices where $(s_{t_j,h}, a_{t_j,h}) = (s,a)$, and let $z_j \triangleq s_{t_j,h+1} \in \mathcal{S}_{h+1}$ be the observed next states. Pairs with $m_{s,a} = 0$ contribute zero and can be omitted. For $m_{s,a} \geq 1$, let $S_{h+1} = |\mathcal{S}_{h+1}|$ and $q(s') := n_{s'}/m_{s,a}$ be the empirical distribution of outcomes. Then, following Lemma D.2 and Lemma D.3, we can bound $\sum_{j=1}^{m_{s,a}} \left(-\log \widehat{P}_{t_j}(z_j \mid s,a)\right) - \sum_{j=1}^{m_{s,a}} (-\log \mathcal{P}(z_j \mid s,a))$ as follows:

$$\sum_{j=1}^{m_{s,a}} \left(-\log \widehat{P}_{t_j}(z_j \mid s,a)\right) - \sum_{j=1}^{m_{s,a}} (-\log \mathcal{P}(z_j \mid s,a))$$

$$= -\log\left(\prod_{j=1}^{m_{s,a}} \widehat{P}_{t_j}(z_j \mid s,a)\right) + \sum_{s'\in\mathcal{S}_{h+1}} n_{s'}\log\mathcal{P}(s' \mid s,a)$$

$$= \log\left(\frac{(S_{h+1}+m_{s,a}-1)!}{(S_{h+1}-1)!\prod_{s'} n_{s'}!}\right) + m_{s,a}\sum_{s'} q(s')\log\mathcal{P}(s' \mid s,a)$$

(by Lemma D.2 and $n_{s'} = m_{s,a}q(s')$)

$$\leq \left[-m_{s,a}\sum_{s'} q(s')\log q(s') + C_{s,a}\right] + m_{s,a}\sum_{s'} q(s')\log\mathcal{P}(s'|s,a) \qquad \text{(By Lemma D.3)}$$

$$= -m_{s,a}\sum_{s'} q(s')\left(\log q(s') - \log\mathcal{P}(s'|s,a)\right) + C_{s,a} \qquad \text{(rearranging terms)}$$

$$\leq (S_{h+1}-1)\log(S_{h+1}+m_{s,a}), \qquad \text{(since KL-distance is non-negative)}$$

where we define $C_{s,a} \triangleq (S_{h+1}-1)\log(S_{h+1}+m_{s,a})$.

Taking summation over all state-action pairs and all layers, we know that

$$\sum_{t=1}^{T}\sum_{h=1}^{H-1}\left(-\log\widehat{P}_t(s_{t,h+1} \mid s_{t,h},a_{t,h})\right) - \sum_{t=1}^{T}\sum_{h=1}^{H-1}\left(-\log\mathcal{P}(s_{t,h+1} \mid s_{t,h},a_{t,h})\right)$$

$$= \sum_{h=1}^{H-1}\sum_{(s,a)\in\mathcal{S}_h\times\mathcal{A}}\sum_{j=1}^{m_{s,a}}\left(-\log\widehat{P}_{t_j}(z_j \mid s,a)\right) - \sum_{h=1}^{H-1}\sum_{(s,a)\in\mathcal{S}_h\times\mathcal{A}}\sum_{j=1}^{m_{s,a}}\left(-\log\mathcal{P}(z_j \mid s,a)\right)$$

$$\leq \sum_{h=1}^{H-1} S_h S_{h+1} K\log(S+TH) \qquad \text{(since } m_{s,a}\leq TH \text{ and } S_{h+1}\leq S\text{)}$$

$$\leq \left(\sum_{h=1}^{H-1} S_h\right)\left(\sum_{h=1}^{H-1} S_{h+1}\right) K\log(S+TH) \leq S^2 K\log(S+TH).$$

□

D.3 Proof of Theorem 4.3

In this subsection, we prove our main result, Theorem 4.3. We set $\delta = 1/(2T^2)$ as in Algorithm 1. Each homing call uses failure probability $\delta/|\mathcal{S}_H|$, and the simultaneous concentration, sampling, and ERM guarantees in Section 3 each use δ. Their joint failure probability is $\mathcal{O}(\delta)$, contributing at most $\mathcal{O}(T\delta) = \mathcal{O}(1/T)$ to expected regret.

We first consider $0 < \varepsilon < 1/4$; the remaining case is handled after the parameter choice below. For analysis, let N_{exp} be the number of episodes needed to complete exploration without the budget cutoff, and let $\mathcal{E}$ denote the joint success event above. The actual algorithm uses only the first T episodes, so define $T_{\text{start}} = \min\{N_{\text{exp}}+1, T+1\}$. Its online phase is empty when exploration exhausts the budget. We decompose the expected regret as

$$\mathbf{Reg} \leq \underbrace{\mathbb{E}\left[\mathbb{1}\{\mathcal{E}\}\sum_{t=1}^{T_{\text{start}}-1}(V^\star - V(\pi_t))\right]}_{\mathbf{Reg}_{\text{EXPLORE}}} + \underbrace{\mathbb{E}\left[\mathbb{1}\{\mathcal{E}\}\sum_{t=T_{\text{start}}}^{T}(V^\star - V(\pi_t))\right]}_{\mathbf{Reg}_{\text{ONLINE}}} + \mathcal{O}(T\delta).$$

For the online analysis, we condition on a completed exploration history in $\mathcal{E}$ with $N_{\text{exp}} < T$. The retained set and decoder are then fixed, while future contexts, transitions, rewards, and feedback retain their specified conditional laws, including the fixed kernel in Assumption 2.1. The resulting bounds hold uniformly over these histories and are averaged to bound $\mathbf{Reg}_{\text{ONLINE}}$; all other histories contribute zero to that term.

D.3.1 BOUNDING $\mathbf{Reg}_{\text{EXPLORE}}$

The exploration phase consists of learning and evaluating the homing policies, followed by collecting tuples for retained states. On $\mathcal{E}$, its regret is at most $\min\{T, N_{\text{exp}}\} \le N_{\text{exp}}$, so the bounds on the untruncated exploration cost remain valid under the budget cutoff.

For the first part, Lemma 3.1 bounds the cost of learning homing policies for all $s \in \mathcal{S}_H$ by $\widetilde{\mathcal{O}}\left(\frac{S^2KH}{\varepsilon^2} + \frac{S^3KH^4}{\varepsilon}\right)$ episodes. Evaluating each policy on N fresh episodes adds the same order of cost.

For the second part, according to Lemma 3.5, under the high-probability events, collecting N_0 samples for all reachable states $s \in \widehat{\mathcal{S}}_{H,\varepsilon}$ using Algorithm 3 requires $\widetilde{\mathcal{O}}(\frac{SN_0}{\varepsilon})$ episodes, leading to an $\widetilde{\mathcal{O}}(\frac{SN_0}{\varepsilon})$ regret. Thus, $\mathbf{Reg}_{\text{explore}}$ is bounded as:

$$\mathbf{Reg}_{\text{EXPLORE}} \le \widetilde{\mathcal{O}}\left(\frac{S^2KH}{\varepsilon^2} + \frac{S^3KH^4}{\varepsilon} + \frac{SN_0}{\varepsilon}\right). \tag{10}$$

D.3.2 BOUNDING $\mathbf{Reg}_{\text{ONLINE}}$

We analyze the online phase using a surrogate MDP with the same context distribution and transition kernel as the original MDP, but with terminal mean reward

$$g^\star(x,s,a) \triangleq \underline{f}^\star(x,s,a)\mathbb{1}\{s \in \widehat{\mathcal{S}}_{H,\varepsilon}\}, \qquad \underline{f}^\star(x,s,a) = \mathbb{E}_{y|x,s,a}[J(h^\star(x,y,s),a)].$$

All intermediate rewards are zero. The retained set and hence $g^\star$ are fixed after exploration. For any terminal mean reward f, let $V(\pi; f) \triangleq \mathbb{E}_\pi[f(x, s_H, a_H)]$, where the expectation includes $x \sim \mathcal{D}$, and define $V_g^\star \triangleq \max_{\pi\in\Pi} V(\pi; g^\star)$.

We first relate the original regret to regret in this surrogate MDP. By Lemma 3.7, $\underline{f}^\star \le f^\star$, with equality at every terminal action maximizing $f^\star(x,s,\cdot)$. Without loss of generality, choose $\pi^\star$ to take such an action at every terminal state: replacing its terminal actions in this way cannot reduce its value. Consequently, $V(\pi^\star; f^\star) = V(\pi^\star; \underline{f}^\star)$. Moreover, Lemma 3.2 gives $p_s^\star \le 6\varepsilon$ for every excluded terminal state, so any policy reaches the excluded set with probability at most $6S\varepsilon$. Since $\underline{f}^\star \in [0,1]$, it follows that

$$V^\star = V(\pi^\star; \underline{f}^\star) \le V(\pi^\star; g^\star) + 6S\varepsilon \le V_g^\star + 6S\varepsilon.$$

For the learner, $V(\pi_t; f^\star) \ge V(\pi_t; g^\star)$. Combining these inequalities gives

$$\mathbf{Reg}_{\text{ONLINE}} \le \mathbb{E}\left[\sum_{t=T_{\text{start}}}^{T} \left(V_g^\star - V(\pi_t; g^\star)\right)\right] + 6TS\varepsilon. \tag{11}$$

The reward prediction used in the occupancy optimization of Algorithm 4 is

$$\widehat{g}_t(x,s,a) \triangleq \widehat{f}_t(x,s,a)\mathbb{1}\{s \in \widehat{\mathcal{S}}_{H,\varepsilon}\}$$

at terminal states, and zero at intermediate states. Thus, both the true surrogate reward and its prediction vanish outside the retained terminal set. This ensures that the reward error needed for the surrogate MDP analysis is exactly the error controlled by the regression oracle on its update subsequence.

We define the cumulative expected estimation errors for the reward and dynamics as follows:

$$\mathcal{E}_{\text{rew}} \triangleq \mathbb{E}\left[\sum_{t=T_{\text{start}}}^{T} \left(\widehat{f}_t(x_t, s_{t,H}, a_{t,H}) - \underline{f}^\star(x_t, s_{t,H}, a_{t,H})\right)^2 \cdot \mathbb{1}\{s_{t,H} \in \widehat{\mathcal{S}}_{H,\varepsilon}\}\right],$$

$$\mathcal{E}_{\text{dyn}} \triangleq \mathbb{E}\left[\sum_{t=T_{\text{start}}}^{T} \sum_{h=1}^{H-1} \left(-\log \widehat{P}_t(s_{t,h+1} \mid s_{t,h}, a_{t,h}) + \log \mathcal{P}(s_{t,h+1} \mid s_{t,h}, a_{t,h})\right)\right].$$

In particular, extending $g^\star$ and $\widehat{g}_t$ by zero at intermediate states yields the identity

$$\mathbb{E}\left[\sum_{t=T_{\text{start}}}^{T} \sum_{h=1}^{H} \left(\widehat{g}_t(x_t, s_{t,h}, a_{t,h}) - g^\star(x_t, s_{t,h}, a_{t,h})\right)^2\right] = \mathcal{E}_{\text{rew}}.$$

There is no reward prediction error on excluded states. We do not require $g^\star \in \mathcal{F}$: the oracle is compared with $\underline{f}^\star \in \mathcal{F}$ only on the retained-state update subsequence, as shown below. For the dynamics, the conditional expected log-loss difference equals the KL divergence from $\mathcal{P}(\cdot \mid s,a)$ to $\widehat{P}_t(\cdot \mid s,a)$, which upper bounds their squared Hellinger distance. Thus, the cumulative expected squared Hellinger error is at most $\mathcal{E}_{\text{dyn}}$.

Applying the value-difference bounds in Lemmas 10–12 of Levy et al. (2023) to this surrogate MDP and the predictions $(\widehat{g}_t, \widehat{P}_t)$, and then taking expectations, gives

$$\mathbb{E}\left[\sum_{t=T_{\text{start}}}^{T}\left(V_g^\star - V(\pi_t; g^\star)\right)\right] \le \widetilde{\mathcal{O}}\left(\frac{HSKT}{\gamma} + \gamma\mathcal{E}_{\text{rew}} + \gamma H^4(\mathcal{E}_{\text{dyn}} + H) + \sqrt{TH(\mathcal{E}_{\text{dyn}} + H)} + \sqrt{TH\mathcal{E}_{\text{rew}}}\right).$$

Substituting the bound $\mathcal{E}_{\text{dyn}} \le \widetilde{\mathcal{O}}(S^2K)$ according to Theorem D.1, we further obtain that:

$$\mathbb{E}\left[\sum_{t=T_{\text{start}}}^{T}\left(V_g^\star - V(\pi_t; g^\star)\right)\right] \le \widetilde{\mathcal{O}}\left(\frac{HSKT}{\gamma} + \gamma\mathcal{E}_{\text{rew}} + \gamma H^4S^2K + \sqrt{THS^2K} + \sqrt{TH\mathcal{E}_{\text{rew}}}\right). \tag{12}$$

Next, we bound the reward estimation error $\mathcal{E}_{\text{rew}}$. Let $\mathcal{T}_{\text{reach}} \triangleq \{t \in \{T_{\text{start}}, T_{\text{start}}+1, \ldots, T\} : s_{t,H} \in \widehat{\mathcal{S}}_{H,\varepsilon}\}$ denote the set of time indices where the reachable states are visited. For brevity, let $\zeta_t \triangleq (x_t, s_{t,H}, a_{t,H})$. For $s \in \widehat{\mathcal{S}}_{H,\varepsilon}$, define $\bar{f}(x,s,a) \triangleq \mathbb{E}_{y|x,s,a}[J(\widehat{h}(x,y,s),a)]$, and set $\bar{f}(x,s,a) = 0$ otherwise. For expressions involving $\widehat{h}$ outside the retained set, extend $\widehat{h}$ there by the uniform vector; this extension is only notational and is never used by the algorithm.

Invoking the regression oracle guarantee (Assumptions 4.1 and 4.2) over the subsequence $\mathcal{T}_{\text{reach}}$, we have:

$$\sum_{t\in\mathcal{T}_{\text{reach}}}(\widehat{f}_t(\zeta_t) - \widetilde{r}_t)^2 - \sum_{t\in\mathcal{T}_{\text{reach}}}(\underline{f}^\star(\zeta_t) - \widetilde{r}_t)^2 \le \mathbf{Reg}_{\mathsf{Sq}}. \tag{13}$$

Using the identity $(a-b)^2 - (c-b)^2 = (a-c)^2 + 2(a-c)(c-b)$ with $a = \widehat{f}_t(\zeta_t)$, $b = \widetilde{r}_t$, and $c = \underline{f}^\star(\zeta_t)$, we can rewrite Eq. (13) as follows:

$$\sum_{t\in\mathcal{T}_{\text{reach}}}(\widehat{f}_t(\zeta_t) - \underline{f}^\star(\zeta_t))^2 + 2\sum_{t\in\mathcal{T}_{\text{reach}}}(\widehat{f}_t(\zeta_t) - \underline{f}^\star(\zeta_t))(\underline{f}^\star(\zeta_t) - \widetilde{r}_t) \le \mathbf{Reg}_{\mathsf{Sq}}. \tag{14}$$

Let $\mathcal{H}_{t-1}$ denote the history before episode t, including all exploration data. Conditional on $\mathcal{H}_{t-1}$ and the current context and trajectory (x_t, τ_t), the oracle prediction, $\widehat{h}$, and the event $t \in \mathcal{T}_{\text{reach}}$ are fixed. By Assumption 2.1, on this event the conditional mean of $\widetilde{r}_t$ is $\bar{f}(\zeta_t)$. In the next display, $\mathbb{E}_{y_t}$ denotes expectation under this conditioning, so

$$\begin{aligned}\mathbb{E}_{y_t}\left[(\widehat{f}_t(\zeta_t) - \underline{f}^\star(\zeta_t))(\underline{f}^\star(\zeta_t) - \widetilde{r}_t)\right] &= (\widehat{f}_t(\zeta_t) - \underline{f}^\star(\zeta_t))(\underline{f}^\star(\zeta_t) - \mathbb{E}_{y_t}[\widetilde{r}_t]) \\ &= (\widehat{f}_t(\zeta_t) - \underline{f}^\star(\zeta_t))(\underline{f}^\star(\zeta_t) - \bar{f}(\zeta_t)).\end{aligned} \tag{15}$$

Finally, we lower bound this term using the AM-GM inequality variation $ab \ge -\frac{1}{4}a^2 - b^2$ (by setting $a = \widehat{f}_t(\zeta_t) - \underline{f}^\star(\zeta_t)$ and $b = \underline{f}^\star(\zeta_t) - \bar{f}(\zeta_t)$):

$$(\widehat{f}_t(\zeta_t) - \underline{f}^\star(\zeta_t)) \cdot (\underline{f}^\star(\zeta_t) - \bar{f}(\zeta_t)) \ge -\frac{1}{4}(\widehat{f}_t(\zeta_t) - \underline{f}^\star(\zeta_t))^2 - (\underline{f}^\star(\zeta_t) - \bar{f}(\zeta_t))^2. \tag{16}$$

Taking the total expectation and rearranging the terms, we can obtain that:

$$\mathcal{E}_{\text{rew}} = \mathbb{E}\left[\sum_{t\in\mathcal{T}_{\text{reach}}}(\widehat{f}_t(\zeta_t) - \underline{f}^\star(\zeta_t))^2\right]$$

$$\leq \mathbf{Reg}_{\mathsf{Sq}} - 2\mathbb{E}\left[\sum_{t\in\mathcal{T}_{\text{reach}}} (\widehat{f}_t(\zeta_t) - \underline{f}^\star(\zeta_t))(\underline{f}^\star(\zeta_t) - \widetilde{r}_t)\right] \qquad \text{(according to Eq. (14))}$$

$$\leq \mathbf{Reg}_{\mathsf{Sq}} + \frac{1}{2}\mathbb{E}\left[\sum_{t\in\mathcal{T}_{\text{reach}}} (\widehat{f}_t(\zeta_t) - \underline{f}^\star(\zeta_t))^2\right] + 2\mathbb{E}\left[\sum_{t\in\mathcal{T}_{\text{reach}}} (\underline{f}^\star(\zeta_t) - \bar{f}(\zeta_t))^2\right]$$

(according to Eq. (16))

$$= \mathbf{Reg}_{\mathsf{Sq}} + \frac{1}{2}\mathcal{E}_{\text{rew}} + 2\mathbb{E}\left[\sum_{t\in\mathcal{T}_{\text{reach}}} (\underline{f}^\star(\zeta_t) - \bar{f}(\zeta_t))^2\right].$$

Rearranging the terms leads to the following:

$$\mathcal{E}_{\text{rew}} \leq 2\mathbf{Reg}_{\mathsf{Sq}} + 4\mathbb{E}\left[\sum_{t\in\mathcal{T}_{\text{reach}}} (\underline{f}^\star(\zeta_t) - \bar{f}(\zeta_t))^2\right]. \tag{17}$$

We now bound the second term on the RHS of Eq. (17). Using the definitions of $\underline{f}^\star$ and $\bar{f}$, we apply Jensen's inequality and the L-Lipschitz property of $J(\cdot, a_{t,H})$ with respect to the ℓ_∞-norm:

$$\begin{aligned}
(\underline{f}^\star(\zeta_t) - \bar{f}(\zeta_t))^2 \cdot \mathbb{1}\{s_{t,H} \in \widehat{\mathcal{S}}_{H,\varepsilon}\} &= \mathbb{1}\{s_{t,H} \in \widehat{\mathcal{S}}_{H,\varepsilon}\} \cdot \left(\mathbb{E}_{y_t}[J(h^\star(\zeta_t'), a_{t,H}) - J(\widehat{h}(\zeta_t'), a_{t,H})]\right)^2 \\
&\leq \mathbb{1}\{s_{t,H} \in \widehat{\mathcal{S}}_{H,\varepsilon}\} \cdot \mathbb{E}_{y_t}\left[(J(h^\star(\zeta_t'), a_{t,H}) - J(\widehat{h}(\zeta_t'), a_{t,H}))^2\right] \\
&\leq L^2\mathbb{E}_{y_t}\left[\|h^\star(\zeta_t') - \widehat{h}(\zeta_t')\|_\infty^2\right] \cdot \mathbb{1}\{s_{t,H} \in \widehat{\mathcal{S}}_{H,\varepsilon}\},
\end{aligned}$$

where we used the shorthand $\zeta_t' \triangleq (x_t, y_t, s_{t,H})$.

To invoke Lemma 3.6, we must account for both the terminal action distribution and the context-dependent selection of the terminal state. For each $s \in \widehat{\mathcal{S}}_{H,\varepsilon}$, define

$$e_s(x) \triangleq \mathbb{E}_{\substack{a\sim q_{\text{unif}} \\ y|x,s,a}}\left[\|\widehat{h}_s(x,y) - h_s^\star(x,y)\|_2^2\right].$$

By Lemma 3.6, $\mathbb{E}_{x\sim\mathcal{D}}[e_s(x)] \leq \widetilde{\mathcal{O}}(1/N_0)$ for every fixed retained state s. Conditioning on the online history, let $p_t(s \mid x)$ denote the probability that π_t reaches s given context x. Since the terminal-action density ratio relative to q_{unif} is at most K and $\|\cdot\|_\infty^2 \leq \|\cdot\|_2^2$, we have

$$\begin{aligned}
\mathbb{E}\left[\sum_{t\in\mathcal{T}_{\text{reach}}} (\underline{f}^\star(\zeta_t) - \bar{f}(\zeta_t))^2\right] &\leq L^2K \sum_{t=T_{\text{start}}}^{T} \mathbb{E}_{x\sim\mathcal{D}}\left[\sum_{s\in\widehat{\mathcal{S}}_{H,\varepsilon}} p_t(s \mid x) e_s(x)\right] \\
&\leq L^2K \sum_{t=T_{\text{start}}}^{T} \sum_{s\in\widehat{\mathcal{S}}_{H,\varepsilon}} \mathbb{E}_{x\sim\mathcal{D}}[e_s(x)] \\
&\leq \widetilde{\mathcal{O}}\left(\frac{TSKL^2}{N_0}\right).
\end{aligned}$$

Plugging the above back to Eq. (17), we conclude that

$$\mathcal{E}_{\text{rew}} \leq 2\mathbf{Reg}_{\mathsf{Sq}} + \widetilde{\mathcal{O}}\left(\frac{TSKL^2}{N_0}\right).$$

Substituting the bound for $\mathcal{E}_{\text{rew}}$ back into Eq. (12), we obtain that

$$\mathbb{E}\left[\sum_{t=T_{\text{start}}}^{T} \left(V_g^\star - V(\pi_t; g^\star)\right)\right]$$
$$\leq \widetilde{\mathcal{O}}\left(\frac{HSKT}{\gamma} + \gamma\mathcal{E}_{\text{rew}} + \gamma H^4S^2K + \sqrt{THS^2K} + \sqrt{TH\mathcal{E}_{\text{rew}}}\right)$$

$$\leq \widetilde{\mathcal{O}}\left(\frac{HSKT}{\gamma} + \gamma \mathbf{Reg}_{\mathsf{Sq}} + \frac{\gamma TSKL^2}{N_0} + \gamma H^4 S^2 K + \sqrt{THS^2K} + \sqrt{TH\mathbf{Reg}_{\mathsf{Sq}}} + TL\sqrt{\frac{SKH}{N_0}}\right). \tag{18}$$

Final Bound. Combining the exploration regret bound (Eq. (10)) with the online regret bounds (Eq. (11)) and (Eq. (18)), the total regret is bounded by:

$$\mathbf{Reg} \leq \widetilde{\mathcal{O}}\Bigg(\frac{S^2KH}{\varepsilon^2} + \frac{S^3KH^4}{\varepsilon} + \frac{SN_0}{\varepsilon} + TS\varepsilon + \frac{HSKT}{\gamma} + \gamma\mathbf{Reg}_{\mathsf{Sq}} + \frac{\gamma TSKL^2}{N_0} + \gamma H^4S^2K + \sqrt{THS^2K} + \sqrt{TH\mathbf{Reg}_{\mathsf{Sq}}} + TL\sqrt{\frac{SKH}{N_0}}\Bigg).$$

We choose $\gamma = \min\left\{\sqrt{\frac{TSKH}{\mathbf{Reg}_{\mathsf{Sq}}}}, \sqrt{\frac{T}{SH^3}}, T^{\frac{1}{4}}K^{\frac{1}{2}}H^{\frac{3}{4}}L^{-\frac{1}{2}}\right\}$, $N_0 = \max\left\{L\sqrt{\frac{TKH}{S}}, \gamma^{\frac{2}{3}}T^{\frac{1}{3}}K^{\frac{2}{3}}L^{\frac{4}{3}}\right\}$, and $\varepsilon = \max\left\{\sqrt{\frac{N_0}{T}}, \left(\frac{SKH}{T}\right)^{\frac{1}{3}}, \frac{S\sqrt{K}H^2}{\sqrt{T}}\right\}$, where the first branch in the definition of γ is omitted when $\mathbf{Reg}_{\mathsf{Sq}} = 0$. Substituting these choices gives

$$\mathbf{Reg} = \widetilde{\mathcal{O}}\Big(T^{\frac{3}{4}}SK^{\frac{1}{2}}H^{\frac{1}{4}}L^{\frac{1}{2}} + T^{\frac{2}{3}}S^{\frac{4}{3}}K^{\frac{1}{3}}H^{\frac{1}{3}} + T^{\frac{1}{2}}S^2K^{\frac{1}{2}}H^2 + T^{\frac{1}{2}}S^{\frac{3}{2}}KH^{\frac{5}{2}} + \sqrt{TSKH\mathbf{Reg}_{\mathsf{Sq}}}\Big).$$

If the chosen $\varepsilon \geq 1/4$, the budget cutoff gives $\mathbf{Reg} \leq T \leq 4TS\varepsilon$. The same algebra bounds $TS\varepsilon$ by the terms in the preceding display, so this display also holds in that case. Finally, when $T \geq \max\left\{1, \frac{S^4H}{K^2L^6}, \frac{S^4H^7}{L^2}, \frac{S^2K^2H^9}{L^2}\right\}$, the first term dominates the remaining non-oracle polynomial terms, which yields the bound stated in Theorem 4.3.

E Omitted Details in Section 5

E.1 Implementation Details in Section 5.1

In this subsection, we present the omitted experiment configuration and implementation details for experiments in Section 5.1. The dynamics and rewards are defined as follows:

- **Transitions ($h < 3$):** Transitions depend on the current state quality. From a good state $s_{h,g}$, action a_1 maintains the good state ($s_{h+1,g}$) with high probability $1-p$ and transitions to the bad state ($s_{h+1,b}$) with probability p. Actions $a_2, \ldots, a_5$ have the reverse effect, transitioning to the bad state with probability $1-p$ and to the good state with probability p. From a bad state $s_{h,b}$, all actions deterministically transition to the next bad state $s_{h+1,b}$.
- **Rewards ($h = 3$):** Rewards are binary and realized only at the final step. At the good terminal state $s_{3,g}$, action a_1 yields reward 1 with probability $1 - p_{\text{reward}}$, while $a_2, \ldots, a_5$ yield reward 1 with probability p_{reward}. At the bad terminal state $s_{3,b}$, the reward is deterministically 0.
- **Feedback Signal:** The learner observes a flipped reward as signal depending on the context. Specifically, $y_t = r_t$ if $x_t = \texttt{True}$, and $y_t = 1 - r_t$ if $x_t = \texttt{False}$.

In our experiments, we set parameters $p = p_{\text{reward}} = 0.1$. Our implementation follows the two-stage pipeline detailed in Section 3 and Section 4.

In the decoder-learning phase, we first run Algorithm 2 for each candidate terminal state $s \in \mathcal{S}_3$ with $\delta = 0.05$ and $N = 5000$ to obtain the homing policies $\widehat{\pi}_s$. We then run each homing policy $\widehat{\pi}_s$ for 1000 episodes and keep the tuples from episodes that end in s, which yields 622 tuples for $s_{3,g}$ and 936 for $s_{3,b}$. To approximate the posterior distribution $\widehat{h}$, we implement the function class $\mathcal{H}$ by parameterizing both the decoder class Φ and the reward function class $\mathcal{F}$ as two-layer fully-connected neural networks. We then solve the ERM problem by training a separate classifier for each terminal state using the collected tuples. Finally, we convert the learned probabilities into a reward estimator using the Lipschitz construction in Eq. (4), with its ramp replaced by $G(v_a, 0.265, 0.05)$.

In the policy-learning phase, we apply Algorithm 4 with a learning rate schedule $\gamma_t = H\sqrt{|\mathcal{A}|t}$ over a total horizon of $T = 40,000$ episodes. The regression oracle is implemented by online gradient descent with learning rate 0.05. Crucially, the learner updates its policy solely based on the decoded rewards; ground-truth rewards are used exclusively for performance evaluation and plotting.

E.2 Dataset Generation Details in Section 5.2

In this subsection, we present the omitted data generation details in Section 5.2. Each episode involves a two-turn dialogue between the agent and the user, corresponding to a **horizon** $H = 2$. In the first turn, the agent responds to a user request by asking a clarification question. The ground-truth question (taken from the dataset) is the most informative, containing all necessary details to complete the booking (e.g., cuisine type and party size for restaurants; genre and location for movies). Conversely, we synthesize two incorrect candidate questions using an LLM. While contextually relevant, these questions fail to include sufficient information to fulfill the request.

In the second turn, the agent finalizes the booking based on the information gathered. Alongside the correct booking action from the dataset, we used an LLM to generate four incorrect booking options, each containing details that deviate from the ground truth. After the agent provides the booking information, we use an LLM to simulate user feedback, indicating whether the user is satisfied or unsatisfied based on the correctness and sufficiency of the information. We provide the specific prompts used for data generation as follows.

Clarification Questions. We present the prompt used to generate incorrect clarification questions, which correspond to the **intermediate actions** a_h. The "`initial_request`" represents the input from the booking dataset, serving as the **context** x in our formulation.

Prompt

You are helping design clarification questions for a movie ticket booking assistant. User's initial request: {initial_request}. Generate one question in English that is about movie booking and sounds natural, but does NOT help the assistant gather the key information needed to complete the booking correctly. For example, ask about irrelevant details like movie reviews, snack preferences, parking, or general theater facilities instead of essential booking details (movie, theater, time, date, tickets). Keep it short and specific to the movie booking context. Return ONLY the question text.

Prompt

You are helping design clarification questions for a restaurant reservation assistant. User's initial request: {initial_request}. Generate one question in English that is about dining/reservations and sounds natural, but does NOT help the assistant gather the key information needed to answer the user correctly. Keep it short and specific to the restaurant context. Return ONLY the question text.

User Responses. Below, we present the prompt used to generate the user's response to the clarification question. The generated response, concatenated with the dialogue history, serves as the **state** s_h.

Prompt

User's initial request: {initial_request}. System asks: {clarification_question}. Simulate how the user would answer this question in English. Return ONLY the user's brief answer (1 sentence), without any explanation or meta-text.

Booking Actions. We present the prompt used to generate incorrect booking actions below. The "`correct_answer`", corresponding to the **final action** a_H, is taken directly from the booking dataset.

Prompt

User request: {initial_request}. User's answer to clarification: {user_answer}. Correct system response: {correct_answer}. Generate one DIFFERENT but related system response for movie booking. Change ONE key detail from the correct answer: either movie name (use a different movie), OR theater name (use a different theater), OR showtime (use a different time), OR date (use a different date), OR number of tickets. Keep the same professional tone and format. Return ONLY the system's response without any quotes or explanations.

Prompt

User request: {initial_request}. User's answer to clarification: user_answer. Correct system response: {correct_answer}. Generate one different but related system response. Change ONE aspect from the correct answer: either cuisine type (try French, Mexican, Spanish, Thai, Greek, Indian, Korean - use DIVERSE types), OR location (different city), OR number of people, OR price range. Keep the EXACT same format with spaces, not underscores. Return ONLY the API call without any quotes or explanations.

User Feedback. We present the prompts used for generating the user's positive and negative feedback, respectively. The **feedback** corresponds to y in our formulation and is generated based on the context x.

Prompt

The user is SATISFIED with this conversation because the system asked the right question and provided the correct information. Generate ONLY the user's direct positive feedback (1-2 sentences). Do NOT include any meta-text like 'Here is', 'Sure', or 'The user says'. Start directly with the user's words.

Prompt

The user is NOT SATISFIED with this conversation because the system's response was wrong or irrelevant. Generate ONLY the user's direct negative feedback expressing dissatisfaction (1-2 sentences). Do NOT include any meta-text like 'Here is', 'Sure', or 'The user says'. Start directly with the user's words.

All synthetic generations were performed using Llama-3-8B-Instruct (Grattafiori et al., 2024).

E.3 Implementation Details in Section 5.2

In this subsection, we present the omitted implementation details in Section 5.2. While Algorithm 1 constructs homing policies for each latent state, this is computationally infeasible in our experimental setting, where the state corresponds to the entire dialogue history. Instead, we adopt a simplified exploration strategy: for the first 500 user requests, we randomly select clarification questions and final actions to collect data, and then train the inverse kinematics predictor $\widehat{h}$ on these samples. The predictor is initialized with Qwen2.5-3B-Instruct (Team et al., 2024) and fine-tuned with a learning rate of 10^{-4} and a batch size of 16. We then learn the policy on the remaining 2,600 samples. Each request is evaluated before being used for the subsequent policy update, so the reported performance follows an online evaluation protocol rather than a separate post-training evaluation. The policy model is initialized with Qwen2.5-3B-Instruct (Team et al., 2024) and employs inverse gap weighting (Foster and Rakhlin, 2020) with $\gamma_t = \sqrt{|\mathcal{A}|t}$ for action selection. We use the REINFORCE algorithm (Williams, 1992) to update it with a learning rate of 5×10^{-5} based on the predicted rewards from the inverse kinematics. Both models utilize the AdamW optimizer (Loshchilov and Hutter, 2017) for parameter updates. All the experiments are conducted on a single NVIDIA H200 GPU. We then present the specific prompts used for model training as follows.

Prompt Used in Learning the Inverse Kinematics Model. The prompt we use in learning the inverse kinematic model is as follows. The final answer corresponds to a_H and the user feedback corresponds to y in our formulation.

Prompt

Evaluate if the User's Final Feedback is consistent with the given conversation. User Initial Request: {initial_request}. System Question: {question}. User Answer: {user_answer}. System Final Action: {final_answer}. User Final Feedback: {feedback}. Is the feedback consistent with this final action? (Yes/No)

Prompt Used in Learning the Policy. The prompt we use in learning the policy is as follows.

Prompt

You are a decision making agent. Based on the conversation, decide whether to take the following final action. Respond with Yes or No only. User Initial Request: {initial_request}. System Clarification Question: {question}. User Clarification Answer: {user_answer}. Candidate Final Action: {action}.

E.4 Baseline Comparisons

We compare our method with a uniform random policy and a naive baseline that directly uses feedback as reward. The uniform random policy samples actions uniformly at each step. On the synthetic MDP, the naive baseline uses the observed binary feedback directly as reward. For booking, it maps positive and negative sentiment in the final feedback to rewards 1 and 0, using only the feedback text. Single-step IGL cannot be applied separately at each stage because intermediate feedback is unavailable, while treating deterministic state-to-action policies as bandit actions gives an exponentially large action space.

As shown in Table 1, our method outperforms the naive baseline by 47.1 percentage points on the synthetic MDP and 4.7 points on booking. These improvements demonstrate the benefit of learning a personalized reward decoder over directly treating feedback as reward.

Table 1: Baseline comparison in terms of true reward (%).

Method	Synthetic MDP	Booking
Uniform random	1.85	20.2
Naive feedback	22.0	94.0
Ours	**69.1**	**98.7**

E.5 Ablation Study

Lipschitz Smoothing. We use experimental ramp parameters $(\beta, \lambda) = (0.265, 0.05)$ for the synthetic MDP and $(0.65, 0.05)$ for Booking. To ablate smoothing, we replace the ramp $G(\alpha, \beta, \lambda)$ by the hard threshold $\mathbb{1}\{\alpha \geq \beta\}$. Table 2 reports the resulting true reward.

Table 2: Lipschitz smoothing ablation in terms of true reward (%).

Task	With Smoothing	Without Smoothing
Synthetic MDP	69.1	8.2
Booking	98.7	96.5

Homogeneous State Handling. To evaluate the role of homogeneous state handling, we remove this component and apply the decoder for heterogeneous states to every retained terminal state, keeping the remaining pipeline unchanged. On the original synthetic MDP, this reduces true reward from 69.1% to 60.7%. Explicitly handling homogeneous states thus improves policy learning by accounting for terminal states whose rewards are constant across actions.

Reachable State Filtering. To examine the effect of filtering a rarely visited terminal state, we augment the synthetic MDP with a terminal state $s_{3,r}$ whose reward is always zero. At $s_{2,g}$, action a_2 transitions to $s_{3,r}$, $s_{3,g}$, and $s_{3,b}$ with probabilities 0.15, 0.75, and 0.10, respectively; all other dynamics remain unchanged. Feedback at $s_{3,r}$ is an independent Bernoulli(0.5) signal. This state is visited in approximately 6% of the homing trials, below the 20% filtering threshold. On this modified instance, filtering achieves 72.2% true reward, compared with 60.1% without filtering, with all other components fixed.

Table 3: Ablations of state handling in terms of true reward (%). Homogeneous state handling is evaluated on the original synthetic MDP, and reachable state filtering on the modified instance described above.

Component	**With Component**	**Without Component**
Homogeneous state handling	69.1	60.7
Reachable state filtering	72.2	60.1