

Is a Good Foundation Necessary for Efficient Reinforcement Learning? The Computational Role of the Base Model in Exploration

Dylan J. Foster
Microsoft Research

DYLANFOSTER@MICROSOFT.COM

Zakaria Mhammedi
Google Research

MHAMMEDI@GOOGLE.COM

Dhruv Rohatgi
Massachusetts Institute of Technology

DROHATGI@MIT.EDU

Editors: Nika Haghtalab and Ankur Moitra

Abstract

Language model alignment (or, reinforcement learning) techniques that leverage *active exploration*—deliberately encouraging the model to produce diverse, informative responses—offer the promise of super-human capabilities. However, current understanding of algorithm design primitives for *computationally efficient* exploration with language models is limited. To better understand how to leverage access to powerful pre-trained generative models to improve the efficiency of exploration, we introduce a new computational framework for RL with language models, in which the learner interacts with the model through a *sampling oracle*. Focusing on the *linear softmax* model parameterization, we provide new results that reveal the computational-statistical tradeoffs of efficient exploration:

1. *Necessity of coverage.* Coverage refers to the extent to which the pre-trained model covers near-optimal responses—a form of hidden knowledge. We show that coverage, while not necessary for data efficiency, lower bounds the *runtime* of any algorithm in our framework.
2. *Inference-time exploration.* We introduce a new algorithm, SpannerSampling, which obtains optimal data efficiency and is computationally efficient whenever the pre-trained model enjoys sufficient coverage, matching our lower bound. SpannerSampling leverages inference-time computation with the pre-trained model to reduce the effective search space for exploration.
3. *Insufficiency of training-time interventions.* We contrast (2) by showing that *training-time* interventions that produce *proper* policies cannot achieve similar guarantees in polynomial time.
4. *Computational benefits of multi-turn exploration.* Finally, we show that under additional representational assumptions, one can achieve improved runtime (replacing sequence-level coverage with token-level coverage) through *multi-turn* exploration.

We view this as a step toward a computational theory of decision making with generative models.

Keywords: Language model, reinforcement learning, exploration, computational-statistical tradeoffs

1. Introduction

Language models are rapidly approaching human performance on a vast array of natural language tasks [9, 65, 81, 63, 31], but current models are constrained by the limitations of passively generated human training data. Domains where high-quality feedback is available (e.g., math and code) offer the tantalizing possibility of overcoming these limitations: By iteratively generating new proposals and refining them with human or super-human feedback (e.g., from a formal proof checker), a language model could eventually discover novel, potentially super-human behaviors and capabilities.

The central hurdles to achieving novel capabilities with this template are (1) the amount of feedback—that is, the *data efficiency*—required by alignment/post-training, and (2) the *computational efficiency*. Both metrics are important, but since gathering feedback is often costly or slow (e.g., due to cost of gathering human labels, or high computational overhead of formal proof checkers), data is often more tightly constrained than computation. Unfortunately, the most popular alignment techniques, like PPO [73] and Online DPO [97, 32], are data-inefficient due to their reliance on passive exploration. These techniques treat the pre-trained model as a *policy* and iteratively update it with reinforcement learning, but since there is no explicit mechanism to promote novelty, they are unlikely to generate positive responses (e.g., novel and correct proofs) by chance [94]. In principle, this issue could be mitigated through *active exploration* techniques developed in the theory of reinforcement learning, which deliberately generate diverse, informative responses [38, 2, 39, 27, 26]. However, these techniques—while satisfactorily *data-efficient*—cannot be implemented in a *computationally efficient* fashion in their most general form [20, 41, 30]. Recent attempts to specialize active exploration to language model alignment face the same issue: such methods require either (1) enumeration over the (exponentially large) space of responses [18, 87, 102, 95]; or (2) non-convex training objectives that are not known to be efficiently implementable in even the simplest settings [94, 13].

The role of the base model. Language model alignment features unique structure not present in general reinforcement learning—most prominently, access to a powerful pre-trained base model (the starting point from which alignment proceeds) that encodes substantial prior knowledge (e.g, whether proofs or programs are at least syntactically valid, if not useful).¹ Yet, there is little understanding of what properties of the base model are necessary for novel behaviors to emerge through RL [64, 22], or whether this process can be accelerated through algorithmic interventions (e.g., the idea of directly using the base model to reduce the effective search space has appeared in many empirical works [55, 33, 82, 101, 95, 99]). Meanwhile, the previously-mentioned theoretical works (based on active exploration) only make superficial use of the base model, rendering the lack of computational efficiency perhaps unsurprising. This motivates the central question we explore:

How can we best leverage access to powerful pre-trained generative models to improve computational efficiency of exploration, and how should we evaluate algorithms that do so?

To address this question, we introduce a new computational framework for language reinforcement learning in which access to the model is abstracted away through a *sampling oracle*, and provide new algorithms and fundamental limits which elucidate essential properties—in particular, the notion of *coverage*—of the pre-trained model for computationally efficient learning. In the process, we bring clarity to computational benefits of algorithmic interventions that have been explored empirically but are not yet fully understood, including (i) benefits of *inference-time computation* [8, 78, 93]; and (ii) benefits of *multi-turn* techniques that explore at the per-step level [53, 68, 47, 75, 74, 96, 43].

1.1. Background: Online Alignment from Reward-Based Feedback

To motivate our computational framework, we begin by formally introducing the statistical problem of language model alignment. We adopt a contextual bandit formalism [70, 95] where the language model is a *policy* $\pi : \mathcal{X} \rightarrow \Delta(\mathcal{Y})$ that maps a prompt (context) $x \in \mathcal{X}$ to a response (action) $y \in \mathcal{Y}$ by sampling $y \sim \pi(\cdot \mid x)$. We begin with a reference policy π_{ref} , which is typically obtained through pre-training or supervised fine-tuning. From here, our alignment protocol proceeds as follows: We

1. Other features include (i) deterministic, known transition dynamics, (ii) regularization to the base model.

receive T_{prompt} i.i.d. prompts $x^1, \dots, x^T \sim \rho \in \Delta(\mathcal{X})$. For each prompt x^t , we can select up to N responses $y_1^t, \dots, y_N^t \in \mathcal{Y}$ (the responses may be sampled from π_{ref} or from some alternative sampling procedure [55, 45, 77]), with which we query a **reward oracle** for a reward $r_i^t \in [0, R_{\max}]$. We assume that $\mathbb{E}[r \mid x, y] = r^*(x, y)$, where $r^* : \mathcal{X} \times \mathcal{Y} \rightarrow [0, R_{\max}]$ is the underlying *reward function*, which represents the feedback source (e.g. verifier or human labeler) that the algorithm interacts with. All responses can be chosen adaptively based on prior feedback—this stands in contrast to traditional offline alignment [102, 56, 36], which is the special case of our formulation in which $y_i^t \sim \pi_{\text{ref}}(x^t)$ for all t .² Once data collection concludes, we produce a final policy $\hat{\pi}$ with the aim of achieving high reward. We define $T_{\text{data}} \leq N \cdot T_{\text{prompt}}$ as the total number of reward queries used by the algorithm. Note that in general, we can have $T_{\text{data}} \ll N \cdot T_{\text{prompt}}$, as the algorithm can potentially abstain from querying the reward oracle for a given prompt.

As in prior work on alignment [95, 102, 94], we focus maximizing a *KL-regularized* reward objective. Letting $J(\pi) := \mathbb{E}_{x \sim \rho, y \sim \pi(x)}[r^*(x, y)]$ denote the average reward and $D_{\text{KL}}(\pi \parallel \pi_{\text{ref}}) := \mathbb{E}_{x \sim \rho}[D_{\text{KL}}(\pi(x) \parallel \pi_{\text{ref}}(x))]$ denote KL-divergence, we define for regularization parameter $\beta > 0$:

$$J_{\beta}(\pi) := J(\pi) - \beta \cdot D_{\text{KL}}(\pi \parallel \pi_{\text{ref}}). \quad (1)$$

We measure the quality of the policy $\hat{\pi}$ via regret to the optimal KL-regularized policy: we desire that $J_{\beta}(\pi_{\beta}^*) - J_{\beta}(\hat{\pi}) \leq \varepsilon$, where $\pi_{\beta}^* := \arg \max_{\pi: \mathcal{X} \rightarrow \Delta(\mathcal{Y})} J_{\beta}(\pi)$ is the optimal policy, and $\varepsilon > 0$ is small. This means that $\hat{\pi}$ achieves near-optimal reward, but does not drift too far from the base policy π_{ref} . We view β as a fixed (but potentially small) problem-dependent parameter, so as to allow novel responses that deviate non-trivially from π_{ref} . We abbreviate $\mathbb{E}_{\pi}[\cdot] := \mathbb{E}_{x \sim \rho, y \sim \pi(\cdot|x)}[\cdot]$.

Remark 1.1 (Autoregressive models). *We focus on the abstract setting above, but our motivating example is autoregressive sequence models of length H , where $\mathcal{Y} = \mathcal{A}^H$ represents the space of token sequences over a vocabulary \mathcal{A} . We will return to this specific setting in Section 5.*

Statistical lens: How much reward data do we need? Since the underlying reward function r^* is unknown to the algorithm designer, the total number of reward queries T_{data} used by an algorithm reflects its *data efficiency*, i.e. how much data needs to be collected from the reward oracle to learn a good policy. Collecting high-quality reward signals can be costly or time-consuming (e.g., when human-generated, or when reward evaluation requires computationally intensive code execution or formal verification), so data efficiency is critical. To give provable data efficiency guarantees, typical alignment algorithms [95, 94] take as input a user-specified policy class $\Pi = \{\pi_{\theta} \mid \theta \in \Theta\}$ for a *parameter space* Θ , and invoke the standard statistical assumption that the optimal policy lies in Π .

Assumption 1.1 (Policy realizability). *The policy class Π satisfies $\pi_{\beta}^* \in \Pi$.*

Remark 1.2 (Preference-based feedback). *Our absolute reward formulation for language model alignment has been used in prior work empirically [88, 89, 85, 96] and in theory [107, 84, 96]. This formulation is closely related to the widely-used theoretical model for RLHF where the learner receives preference-based feedback. Our main algorithms use $N = 2$ and readily extend to the preference-based setting, while our lower bounds allow for general N . See Appendix B.1.*

2. Responses need not be chosen according to the order t ; the algorithm can sample $N' < N$ responses for x^t , then sample responses for another $x^{t'}$ before returning x^t and sampling more. This generality makes our lower bounds stronger; our algorithms use $N = 2$ and proceed in order however.

1.2. A Computational Framework for Online Alignment

The response space \mathcal{Y} in the online alignment framework can be exponentially large (e.g., [Remark 1.1](#)). Without assumptions, there is no hope of learning a near-optimal policy without enumerating over \mathcal{Y} . To address this, we assume that the learning algorithm has access to a certain *sampling oracle*.

Informally, we consider two different settings. **(1)** In the *strong oracle* setting, the learner can draw conditional samples from $\pi_\theta(\cdot | x)$ for any prompt $x \in \mathcal{X}$ and parameter $\theta \in \Theta$ (with the convention that $\mathbf{0} \in \Theta$ and $\pi_{\mathbf{0}} = \pi_{\text{ref}}$). **(2)** In the *weak oracle* setting, the learner can draw conditional samples from $\pi_{\text{ref}}(\cdot | x)$ for any prompt $x \in \mathcal{X}$. We let T_{comp} denote the total number of sampling oracle queries used by the algorithm throughout the learning process. See [Section 2](#) for formal details.

Our algorithms only need the weak oracle, but our lower bounds apply even to the strong oracle. We view access to the weak oracle as a minimal assumption: efficient conditional sampling is arguably *the* defining property of autoregressive language models. We use the sampling oracle complexity T_{comp} as an information-theoretic proxy for the computational efficiency of an alignment algorithm, one that parallels the role of oracle/query complexity [\[61, 44\]](#), and is amenable to upper and lower bounds.

As an example, (reward-based) OnLineDPO [\[32\]](#), is a simple online alignment algorithm: For each $t \in [T_{\text{prompt}}]$, the algorithm computes a parameter $\theta^t \in \Theta$ by optimizing a DPO objective ([Eq. \(6\)](#)) with its current dataset \mathcal{D}^t , then samples a pair of responses $y_1^t, y_2^t \sim \pi_{\theta^t}(\cdot | x^t)$, observes corresponding rewards (r_1^t, r_2^t) , and updates $\mathcal{D}^{t+1} \leftarrow \mathcal{D}^t \cup \{(x^t, y_1^t, y_2^t, r_1^t, r_2^t)\}$. This algorithm uses two (strong) sampling oracle queries per round, so the computational cost is no worse than the cost of gathering feedback: $T_{\text{comp}} = T_{\text{data}}$.³ Unfortunately, since OnLineDPO engages in purely passive exploration, the algorithm’s data efficiency itself is unsatisfactory. We make this distinction quantitative below.

1.3. Linear Softmax Policy Parameterization

To understand when we can hope to achieve favorable data efficiency T_{data} (e.g., through active exploration) without entirely sacrificing computational efficiency T_{comp} , we focus on perhaps the simplest concrete choice of policy class: *linearly parametrized softmax policies* [\[95, 13\]](#).

Definition 1.1. Let $\Theta \subset \mathbb{R}^d$ be a convex parameter set and let $\phi : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}^d$ be a feature embedding. The linear-softmax policy class is $\Pi = \{\pi_\theta : \theta \in \Theta\}$, where $\pi_\theta : \mathcal{X} \rightarrow \Delta(\mathcal{Y})$ is defined by

$$\pi_\theta(y | x) \propto \pi_{\text{ref}}(y | x) \cdot \exp(\beta^{-1} \langle \theta, \phi(x, y) \rangle). \quad (2)$$

With this policy class, [Assumption 1.1](#) becomes a natural assumption about the expressivity of the feature embedding ϕ : for example, if the reward function is linear in the features, i.e. $r^*(x, y) = \langle \theta^*, \phi(x, y) \rangle$ for some $\theta^* \in \mathbb{R}^d$, then the optimal regularized policy π_β^* is exactly π_{θ^*} [\[94\]](#), so that [Assumption 1.1](#) is satisfied so long as $\theta^* \in \Theta$. In spite of the simplicity, [Definition 1.1](#) is rich enough to capture autoregressive sequence models where weights for all but the last layer are frozen; there is some evidence [\[58\]](#) that post-training methods with deep models operate in this “lazy” regime.

For sequence modeling ([Remark 1.1](#)), the *strong* sampling oracle can be at odds with [Definition 1.1](#): while the feature dimension d is bounded, \mathcal{Y} is exponentially large, and even if π_{ref} is an autoregressive sequence model, π_θ may not admit an explicit autoregressive factorization for all θ . However, the *weak* sampling oracle is entirely natural for autoregressive sequence modeling; see [Section 5](#).

3. Technically, OnLineDPO also requires observing the log-densities of the observed responses, though this requirement simplifies for the linear softmax policy class that we consider in the sequel. See [Section 2](#) for details.

Tradeoffs between data and computational efficiency. Let $T_{\text{data}}(\varepsilon, \delta)$ and $T_{\text{comp}}(\varepsilon, \delta)$ denote the reward and sampling oracle queries required for an algorithm to ensure $J_\beta(\pi_\beta^*) - J_\beta(\hat{\pi}) \leq \varepsilon$ with probability at least $1 - \delta$. Even for linear softmax policies, all existing algorithms are unsatisfactory with respect to $T_{\text{data}}(\varepsilon, \delta)$ or $T_{\text{comp}}(\varepsilon, \delta)$. On one hand, Xie et al. [94] show that if we define

$$C_{\text{cov}}(\pi_\beta^*) := \sup_{x \in \mathcal{X}, y \in \mathcal{Y}} \frac{\pi_\beta^*(y \mid x)}{\pi_{\text{ref}}(y \mid x)} \quad (3)$$

as the *coverage coefficient* for π_β^* , then the OnlineDPO method in the prequel, while implementable in polynomial time, must suffer $T_{\text{data}}(\varepsilon, \delta) \gtrsim \min\left\{C_{\text{cov}}(\pi_\beta^*), \exp\left(\frac{R_{\max}}{\beta}\right)\right\}$ when $d = O(1)$ and $\varepsilon, \delta = \Omega(1)$. Informally, $C_{\text{cov}}(\pi_\beta^*)$ represents the number of responses one must draw from π_{ref} before high reward is observed by chance [8, 78, 93]. This is a form of hidden knowledge, but its presence in T_{data} reflects passive exploration. On the other hand, Xie et al. [94] introduced a variant of OnlineDPO called XP0 (see Appendix B), which augments OnlineDPO with active exploration. This allows XP0 to achieve polynomial data efficiency, irrespective of whether π_{ref} has favorable coverage:

$$T_{\text{data}}(\varepsilon, \delta) \lesssim \frac{d^2 \log(\delta^{-1})}{\varepsilon^2}. \quad (4)$$

Note that $C_{\text{cov}}(\pi_\beta^*) \gg \text{poly}(d)$ in general, representing a benefit over passive exploration. Like OnlineDPO, XP0 uses the sampling oracle to generate two responses $(y_1^t, y_2^t) \sim \pi_{\theta^t}(\cdot \mid x^t)$ at each iteration. Yet, while the objective XP0 uses to update the policy $\pi_{\theta^{t+1}}$ is amenable to gradient-based methods, it introduces non-convexity not present in the DPO objective, and it is not known whether it can be minimized in polynomial time (nor with $T_{\text{comp}}(\varepsilon, \delta)$ polynomial) for linear softmax policies, even when $|\mathcal{Y}|$ is small. Other active exploration algorithms are similarly unsatisfactory [18, 102, 95, 13].

1.4. Contributions

We develop a sharp understanding of computational-statistical tradeoffs for online alignment with linear softmax policies, highlighting the central role of the base model (policy) π_{ref} in enabling computational efficiency, along with benefits of inference-time computation and multi-step exploration.

The (computational) necessity of coverage (Section 2). The coverage coefficient $C_{\text{cov}}(\pi_\beta^*)$ captures the extent to which π_{ref} covers near-optimal responses—a form of knowledge encoded in the pre-trained model [8, 78, 93]. While coverage is not necessary for data efficiency (e.g., Eq. (4)), we show that it is required for computational efficiency. Formally (Theorem 2.1), for any algorithm in the sampling oracle framework, the number of sampling oracle calls (and runtime) is lower bounded as $T_{\text{comp}}(\varepsilon, \delta) \gtrsim \min\left\{C_{\text{cov}}(\pi_\beta^*), \exp\left(\frac{R_{\max}}{\beta}\right)\right\}$. This serves as a skyline for algorithm design, and contributes to a growing body of work that highlights the computational benefits of coverage [35].

Efficient inference-time exploration (Section 3). We give a new algorithm, SpannerSampling, which (i) achieves near-optimal data efficiency $T_{\text{data}}(\varepsilon, \delta) \lesssim \text{poly}(d, \beta^{-1}, \varepsilon^{-1}, \log(\delta^{-1}))$ for both rewards and prompts, and (ii) runs in polynomial time, achieving minimal oracle efficiency as governed by the lower bound above: $T_{\text{comp}}(\varepsilon, \delta) \lesssim \text{poly}(C_{\text{cov}}(\pi_\beta^*), T_{\text{data}}(\varepsilon, \delta))$ (Theorem 3.1). SpannerSampling leverages inference-time computation to tilt learned policies toward an exploratory distribution, using the base policy π_{ref} to reduce the effective search space for exploration.

Insufficiency of training-time interventions (Section 4). Active exploration algorithms based on “training-time” interventions (e.g., modifications to the DPO objective, as in XPO) are typically *proper* in the sense that they explore using a sequence $\pi_{\theta^1}, \dots, \pi_{\theta^T}$ of iteratively computed linear softmax policies and ultimately output such a policy; meanwhile SpannerSampling, by invoking extra inference-time computation, engages in *improper* exploration. We show (Theorem 4.1) that *no data-efficient proper exploration algorithm can run in polynomial time* (incl. polynomial dependence on $C_{\text{cov}}(\pi_\beta^*)$ and $\exp(R_{\max}/\beta)$). This gives a separation between algorithms based on training-time interventions and algorithms like SpannerSampling that explore improperly through inference-time computation.

Computational benefits of multi-turn exploration (Section 5). The preceding results, when specialized to autoregressive modeling, engage in exploration at the sequence-level. As a final result (Theorem 5.1), we show that under the additional representational condition that π_β^* can be represented as an autoregressive policy, it is possible to achieve substantially improved runtime and oracle complexity T_{comp} (replacing the coverage coefficient $C_{\text{cov}}(\pi_\beta^*)$ with an *token-level* counterpart) by appealing to *multi-turn* exploration at the per-step (token or sub-sequence) level [53, 68, 47, 75, 74, 96, 43]. This is achieved as a special case of a more general result: any MDP where the optimal KL-regularized value function Q_β^* is linear can be efficiently learned in the reset access model.

2. Sampling Oracle Framework and Necessity of Coverage

In this section, we formally introduce our sampling oracle framework, then prove that coverage for the base policy π_{ref} is necessary for computational efficiency in this framework. Henceforth (until Section 5), we focus on the linear softmax parameterization in Definition 1.1 and make Assumption 1.1. For statistical tractability, we make a (standard) norm bound assumption.

Assumption 2.1. *We assume all $\theta \in \Theta$ satisfy $\|\theta\| \leq B$ for a parameter $B > 0$, and that $\|\phi(x, y)\| \leq 1$ and $\langle \theta^*, \phi(x, y) - \phi(x, y') \rangle \in [-R_{\max}, R_{\max}]$ for all $x \in \mathcal{X}$, $y, y' \in \mathcal{Y}$. Furthermore, we assume that $\mathbf{0} \in \Theta$.*

We assume that $\beta \leq R_{\max} \leq B$ without loss of generality.⁴ We do not explicitly assume that rewards are linear (i.e., $r^*(x, y) = \langle \theta^*, \phi(x, y) \rangle$), but under Assumption 1.1 we have (Lemma G.1):

$$r^*(x, y) - r^*(x, y') = \langle \theta^*, \phi(x, y) - \phi(x, y') \rangle \quad \forall x \in \mathcal{X}, y, y' \in \mathcal{Y}. \quad (5)$$

We write $f = \tilde{O}(g)$ to denote that $f = O(g \cdot \max\{1, \text{polylog}(g)\})$ and $a \lesssim b$ as shorthand for $a = O(b)$. We use $\mathbb{B}_p(r)$ to denote the ℓ_p -ball of radius r , and define $\|x\|_\Sigma^2 = \langle x, \Sigma x \rangle$ for $\Sigma \succ 0$.

2.1. Sampling Oracle Framework

We now formally define our computational framework for the linear softmax policy parameterization. We assume the prompt space \mathcal{X} , response space \mathcal{Y} , and parameter space Θ are given to the alignment protocol, but the feature embedding ϕ and the reference policy π_{ref} are specified only implicitly (i.e., are “unknown” a-priori), and must be accessed through one of the following computational oracles.

Definition 2.1 (Sampling oracles). **Setting I (strong oracle):** *In one query, the learner proposes a prompt $x \in \mathcal{X}$ and parameter $\theta \in \Theta$, and receives a conditional sample $y \sim \pi_\theta(\cdot \mid x)$, as well as the*

4. If $R_{\max} < \beta$, OnlineDPO itself is statistically efficient. Our main upper bounds depend on B only logarithmically.

corresponding feature $\phi(x, y)$ for the sampled response (note that $\pi_0 = \pi_{\text{ref}}$).

Setting II (weak oracle): In one query, the learner proposes a prompt $x \in \mathcal{X}$ and receives a conditional sample $y \sim \pi_{\text{ref}}(\cdot | x)$, as well as the corresponding feature $\phi(x, y)$.

Definition 2.2. An online alignment algorithm in the (strong/weak) setting is an algorithm that, given parameters $\varepsilon, \delta > 0$, produces a policy $\hat{\pi}$ satisfying $J_\beta(\pi_\beta^*) - J_\beta(\hat{\pi}) \leq \varepsilon$ with probability at least $1 - \delta$. We write $T_{\text{data}}(\varepsilon, \delta)$ and $T_{\text{comp}}(\varepsilon, \delta)$ to denote the total number of reward oracle queries and (strong/weak) sampling oracle queries respectively.

Notice, any algorithm operating in our framework (i) must invoke the sampling oracle if it wishes to query the reward oracle with some $y \sim \pi_\theta(\cdot | x^t)$, and (ii) only has knowledge of the features $\phi(x, y)$ that have previously been revealed by the sampling oracle. As an example, given a dataset $\mathcal{D}^t = \{(x^i, y_1^i, y_2^i, r_1^i, r_2^i)\}_{i < t}$ of prompt/response/reward tuples, the (reward-based) OnlineDPO update takes the form

$$\theta^t = \arg \min_{\theta \in \Theta} \sum_{i < t} \left(\beta \log \frac{\pi_\theta(y_1^i | x^i)}{\pi_{\text{ref}}(y_1^i | x^i)} - \beta \log \frac{\pi_\theta(y_2^i | x^i)}{\pi_{\text{ref}}(y_2^i | x^i)} - (r_1^i - r_2^i) \right)^2. \quad (6)$$

Since $\beta \log \frac{\pi_\theta(y_1^i | x^i)}{\pi_{\text{ref}}(y_1^i | x^i)} - \beta \log \frac{\pi_\theta(y_2^i | x^i)}{\pi_{\text{ref}}(y_2^i | x^i)} = \langle \theta, \phi(x^i, y_1^i) - \phi(x^i, y_2^i) \rangle$ and this objective only evaluates $\phi(x, y)$ for previously drawn responses, we see that it can be implemented in the strong setting (Definition 2.2), using the strong sampling oracle to draw $(y_1^t, y_2^t) \sim \pi_{\theta^t}(\cdot | x^t)$.

As we will discuss in Section 5, algorithms that use the weak oracle have important consequences when we specialize our to autoregressive sequence modeling; our main algorithm, SpannerSampling enjoys this property. See Appendix B for connections to classical oracles used in online learning and decision making, and Appendix C for discussion about the framework’s generality.

2.2. Coverage is Necessary for Computational Efficiency

We now present the first of our main results, which shows that the coverage coefficient $C_{\text{cov}}(\pi_\beta^*)$ lower bounds the number of sampling oracle queries (and hence runtime) of any algorithm in our framework.

Theorem 2.1 (Necessity of coverage). *Let $C^*, Y \geq 2$ be given. Let Alg be an online alignment algorithm that uses $T_{\text{data}}(\varepsilon, \delta)$ reward oracle queries and $T_{\text{comp}}(\varepsilon, \delta)$ strong sampling oracle queries whenever (i) the parameter space is $\Theta = \mathbb{B}_2(1)$, (ii) Assumption 2.1 is satisfied with $R_{\text{max}} = B = 1$, (iii) $C_{\text{cov}}(\pi_\beta^*) \leq C^*$, and (iv) the response space has size $Y = |\mathcal{Y}|$. Then, either $T_{\text{data}}(\varepsilon, \delta) \geq Y/8$, or*

$$T_{\text{comp}}(\varepsilon, \delta) \geq \Omega\left(\min\{e^{\beta^2 d/2}, e^{\beta^{-1}/2}, C^*\}\right). \quad (7)$$

For simplicity, consider the regime where $d \geq \beta^{-3}$. Then Theorem 2.1 shows that any algorithm needs $T_{\text{comp}}(\varepsilon, \delta) \geq \Omega(\min\{e^{\beta^{-1}/2}, C_{\text{cov}}(\pi_\beta^*)\})$ to achieve non-trivial data-efficiency $T_{\text{data}}(\varepsilon, \delta) \ll |\mathcal{Y}|$; note that the presence of $e^{\beta^{-1}}$ in the lower bound is fundamental, as we always have $C_{\text{cov}}(\pi_\beta^*) \leq \exp(R_{\text{max}}/\beta)$. We emphasize that this construction uses a single prompt. The intuition for the construction in Theorem 2.1 is as follows. There is a single “hidden” response y^* that the algorithm must discover to achieve high reward. Because the base policy places low probability on this response, we are unlikely to sample $y^* \sim \pi_\theta(\cdot | x)$ unless $\langle \theta, \theta^* \rangle \geq 1 - \beta$. This leaves the algorithm designer with two options: (i) brute-force search over $\theta \in \mathbb{B}_2(1)$ until we find $\langle \theta, \theta^* \rangle \geq 1 - \beta$, which requires an exponential number of oracle queries in the dimension d , or (ii) eat the cost of the coverage coefficient by drawing roughly $C_{\text{cov}}(\pi_\beta^*)$ responses $y \sim \pi_{\text{ref}}(\cdot | x)$ until we observe y^* .

3. Efficient Online Alignment via Inference-Time Exploration

[Theorem 2.1](#) serves as a skyline, showing that coverage (hidden knowledge) for the base policy is essential for computationally efficient alignment; note that various works have shown that existing pre-trained models exhibit favorable coverage for tasks of interest [8, 78, 93]. We now present our main algorithm, *SpannerSampling*, which achieves the computational skyline in [Eq. \(7\)](#) without sacrificing the polynomial data-efficiency achieved by (inefficient) active exploration algorithms such as XPO.

3.1. Algorithm: SpannerSampling

SpannerSampling is displayed in [Algorithm 3](#) (deferred to [Appendix G.1](#) due to space constraints). The algorithm consists of two phases, a *spanner computation phase* and an *exploration phase*.

Spanner phase. Define *relative features* via $\varphi(x, y, y') = \phi(x, y) - \phi(x, y')$; we use these features throughout the algorithm because—per [Eq. \(5\)](#)—the difference in rewards $r^*(x, y) - r^*(x, y')$ is linear under [Assumption 1.1](#). In the first phase, the algorithm aims to compute a *spanner*: a small collection Ψ_{span} of tuples (x, y, y') such that the matrix $\Sigma_{\text{span}} = \lambda I_d + \sum_{(x, y, y') \in \Psi_{\text{span}}} \varphi(x, y, y') \varphi(x, y, y')^T$ covers the feature space in directions that have high probability under the optimal KL-regularized policy π_β^* .⁵ To build the spanner, the algorithm proceeds in T_{prompt} rounds, where at each round $t \in [T_{\text{prompt}}]$, we sample $x^t \sim \rho$, then for each $i \in [T_{\text{span}}]$ sample an independent pair $(y_1^{t,i}, y_2^{t,i}) \sim \pi_{\text{ref}}(\cdot | x^t)$ and check if $\|\varphi(x^t, y_1^{t,i}, y_2^{t,i})\|_{\Sigma_{\text{span}}^{-1}} \geq \nu$ for an accuracy parameter ν ; whenever this occurs, we query the reward oracle for $y_1^{t,i}$ and $y_2^{t,i}$ to receive (r_1^t, r_2^t) and add $(x^t, y_1^{t,i}, y_2^{t,i}, r_1^t, r_2^t)$ to a dataset $\mathcal{D}_{\text{span}}$ for use in the second phase, then proceed to the next round $t + 1$. This process ensures that: (i) the matrix Σ_{span} covers π_β^* well, in the sense that

$$\mathbb{P}_{x \sim \rho, y \sim \pi_\beta^*(\cdot | x), y' \sim \pi_{\text{ref}}(\cdot | x)} \left[\|\varphi(x, y, y')\|_{\Sigma_{\text{span}}^{-1}} > \nu \right] \lesssim \frac{\text{poly}(d, \nu^{-1})}{T_{\text{prompt}}} + \frac{C_{\text{cov}}(\pi_\beta^*)}{T_{\text{span}}}, \quad (8)$$

and (ii) the size of the spanner stays uniformly bounded as $|\Psi_{\text{span}}| \leq \text{poly}(d, \nu^{-1})$. These properties imply that if we estimate θ^* using least squares on any dataset $\mathcal{D} \supset \mathcal{D}_{\text{span}}$, the resulting estimator will have high accuracy on directions covered by π_β^* , up to the error term in [Eq. \(8\)](#). Critically, the size of the spanner—and hence the number of reward queries T_{data} —is uniformly bounded by $\text{poly}(d, \nu^{-1})$, *irrespective* of T_{span} . This means that the second error term in [Eq. \(8\)](#) can be made arbitrarily small by increasing inference-time computation (i.e. T_{span}), without increasing the number of reward oracle queries or prompts.

Exploration phase. In the exploration phase, *SpannerSampling* performs on-policy exploration in order to “fill in” directions that are not well-covered by the spanner. This phase proceeds for T_{exp} rounds, and alternates between (i) computing an estimate θ^t in [Line 16](#) via regression ([Eq. \(49\)](#))⁶ and (ii) updating the dataset $\mathcal{D}_{\text{exp}}^t$, by sampling a pair (y_1^t, y_2^t) (given x^t) from a *truncated softmax policy* parameterized by θ^t and querying the reward oracle for (r_1^t, r_2^t) . The truncated softmax policy $\bar{\pi}_{\theta^t}$ is a new type of exploratory policy which—to our knowledge—is novel to this work, and induces a joint distribution over a pair $(y, y') | x$ via $\bar{\pi}_{\theta^t}(y, y' | x) = \bar{\pi}_{\theta^t}(y | x, y') \pi_{\text{ref}}(y' | x)$, where

$$\bar{\pi}_{\theta^t}(y | x, y') \propto \pi_{\text{ref}}(y | x) \cdot \exp\left(\beta^{-1} \langle \theta^t, \varphi(x, y, y') \rangle \mathbb{I}\{\|\varphi(x, y, y')\|_{\Sigma_{\text{span}}^{-1}} \leq \nu\}\right). \quad (9)$$

5. Our notion of spanner is average-case in nature compared to previous notions in the literature; see [Remark G.1](#).

6. This is equivalent to minimizing the DPO loss: $\sum (\beta \log \frac{\pi_\theta(y_1 | x)}{\pi_{\text{ref}}(y_1 | x)} - \beta \log \frac{\pi_\theta(y_2 | x)}{\pi_{\text{ref}}(y_2 | x)} - (r_1 - r_2))^2$.

Without the indicator in Eq. (9), this coincides with the standard softmax policy $\pi_{\theta^t}(y | x)$, but the indicator “truncates” the reward in directions that are uncertain according to the spanner. Truncation allows SpannerSampling to proceed using only a *weak* sampling oracle: whenever the spanner phase succeeds, we are guaranteed that $\frac{\pi_{\theta^t}(y|x,y')}{\pi_{\text{ref}}(y|x)} \lesssim C_{\text{cov}}(\pi_{\beta}^*)$ for “most” pairs (x, y') (Lemma G.6). This means we can use *rejection sampling* (SoftmaxSampler; Algorithm 1) at inference-time to transform samples from π_{ref} into samples from $\pi_{\theta^t}(y | x, y')$, with computational cost $T_{\text{comp}} = \tilde{O}(C_{\text{cov}}(\pi_{\beta}^*))$ per round.⁷ We write $\hat{\pi}^t(y, y' | x) \approx \pi_{\theta^t}(y, y' | x)$ to denote the distribution induced by rejection sampling with the SoftmaxSampler subroutine, and let $\hat{\pi}^t(y | x) := \mathbb{E}_{y' \sim \pi_{\text{ref}}(\cdot|x)}[\hat{\pi}^t(y | x, y')]$. See Appendix F for detailed background on SoftmaxSampler.

3.2. Guarantee for SpannerSampling

The main guarantee for SpannerSampling is as follows.

Theorem 3.1 (Guarantee for SpannerSampling). *For any $\varepsilon > 0$ and $\delta \in (0, 1)$, by choosing T_{prompt} , T_{span} , T_{exp} appropriately, Algorithm 3 learns a policy with $\mathbb{E}_{\hat{\pi} \sim \text{unif}(\hat{\pi}^1, \dots, \hat{\pi}^{T_{\text{exp}}})} [J_{\beta}(\pi_{\beta}^*) - J_{\beta}(\hat{\pi})] \leq \varepsilon$ with probability at least $1 - \delta$, and achieves the following data efficiency and oracle efficiency bounds:*

$$T_{\text{data}}(\varepsilon, \delta) = \tilde{O}\left(\frac{R_{\text{max}}^2}{\beta}\right) \cdot \frac{d^2 \log^2(\delta^{-1})}{\min\{\varepsilon, \beta\}}, \quad \text{and} \quad T_{\text{comp}}(\varepsilon, \delta) = \tilde{O}\left(C_{\text{cov}}(\pi_{\beta}^*) \cdot \frac{R_{\text{max}}^2}{\beta^2}\right) \cdot T_{\text{data}}(\varepsilon, \delta).$$

The algorithm uses at most $\tilde{O}\left(\frac{R_{\text{max}}^4}{\beta^3}\right) \cdot \frac{d^2 \log^2(\delta^{-1})}{\varepsilon}$ prompts. For any $x \in \mathcal{X}$, one can generate a sample $y \sim \hat{\pi}(\cdot | x)$ from the returned policy using $T_{\text{comp}} = \tilde{O}(C_{\text{cov}}(\pi_{\beta}^))$ weak sampling oracle queries.*

On the computational side, we observe that the number of sampling oracle queries $T_{\text{comp}}(\varepsilon, \delta)$ is controlled by the coverage coefficient $C_{\text{cov}}(\pi_{\beta}^*) \leq \exp(R_{\text{max}}/\beta)$, achieving the lower bound in Theorem 2.1, and the total runtime of the algorithm scales as $\text{poly}(d, C_{\text{cov}}(\pi_{\beta}^*), \varepsilon^{-1}, \beta^{-1}, \log(\delta^{-1}))$.⁸ Furthermore, the algorithm only requires a *weak sampling oracle* (Definition 2.1), and hence can be viewed as performing exploration purely at inference time, with an iteratively updated reward model.

On the statistical side, our bound on $T_{\text{data}}(\varepsilon, \delta)$ matches the minimax rate for linear bandits in terms of dependence on d and R_{max} when $\varepsilon \leq \beta$ [49], and is *independent of the coverage coefficient*, reflecting active exploration. The number of prompts used by the algorithm is also independent of the coverage coefficient, though it is slightly larger than the number of reward queries. We observe that Theorem 3.1 achieves a *fast rate* in the sense that $T_{\text{data}}(\varepsilon, \delta) \lesssim \frac{1}{\beta\varepsilon}$ when $\varepsilon \leq \beta$, improving over the $T_{\text{data}}(\varepsilon, \delta) \lesssim \frac{1}{\varepsilon^2}$ rate for XPO (Eq. (4)) and other prior work [95, 94, 13]; this is a secondary benefit of working with truncated policies (see Eq. (10)), and is facilitated by strong convexity induced by regularization; we view it as analogous to $\frac{1}{\Delta\varepsilon}$ -type rates for bandits with gap- Δ [48, 49]. Concurrent work of Zhao et al. [108] achieves a similar fast rate, but their algorithm is not computationally efficient.

Analysis techniques. The key algorithmic ideas to ensure that $T_{\text{comp}}(\varepsilon, \delta)$ —but not the number of reward queries and prompts—is controlled by $C_{\text{cov}}(\pi_{\beta}^*)$ are: (i) even though T_{span} can grow with $C_{\text{cov}}(\pi_{\beta}^*)$, the size of the spanner Ψ_{span} (and number of reward queries in the spanner phase) is bounded as $\text{poly}(d, \beta^{-1}, R_{\text{max}})$; and (ii) the truncated policy construction ensures we can simulate

7. For a generic parameter θ , we have $C_{\text{cov}}(\pi_{\theta}) \gg C_{\text{cov}}(\pi_{\beta}^*)$. A central insight in our analysis is that we can control the density ratio by $C_{\text{cov}}(\pi_{\beta}^*)$ even when $\theta \neq \theta^*$ by building a spanner and using it to truncate.

8. Indeed, outside of queries to the sampling oracle, the only runtime overhead in SpannerSampling is (i) minimizing the DPO loss (linear least squares), (ii) inverting the second moment matrix, and (iii) evaluating various inner products.

$\bar{\pi}_{\theta^t}$ using rejection sampling with $\tilde{O}(C_{\text{cov}}(\pi_{\beta}^*))$ draws from π_{ref} . Our regret analysis makes use of the following decomposition for truncated softmax policies, which may be of independent interest (see Lemma G.7): Given a parameter θ , define $\varepsilon_{\text{stat}}^2 := \|\theta - \theta^*\|_{\Sigma_{\text{span}}}^2$, and for $\varepsilon > 0$, let

$$\mathcal{X}_{\text{span}}(\varepsilon) := \{x \in \mathcal{X} \mid \mathbb{P}_{(y,y') \sim \pi_{\text{ref}}(x)}[\|\varphi(x, y, y')\|_{\Sigma^{-1}} > \nu] \leq \varepsilon\}.$$

Then under Assumptions 1.1 and 2.1, if $\nu \leq \beta/\varepsilon_{\text{stat}}$, we have that for all $\varepsilon > 0$,

$$\begin{aligned} J_{\beta}(\pi_{\theta^*}) - J_{\beta}(\bar{\pi}_{\theta}) &\leq \frac{1}{\beta} \mathbb{E}_{(y,y') \sim \bar{\pi}_{\theta}(\cdot|x)} \left[\langle \theta - \theta^*, \varphi(x, y, y') \rangle^2 \right] \\ &\quad + O(R_{\max} C_{\text{cov}}(\pi_{\beta}^*)) \cdot \varepsilon + O(R_{\max}) \cdot \mathbb{P}_{x \sim \rho}[x \notin \mathcal{X}_{\text{span}}(\varepsilon)]. \end{aligned} \quad (10)$$

The first term above is controlled by the (on-policy) exploration phase; by virtue of the square, this leads to the fast $\frac{1}{\beta\varepsilon}$ rate. Meanwhile, the last two terms are controlled by the spanner construction: for $\varepsilon_{\text{span}} \approx \frac{1}{T_{\text{span}}}$, we have that $\mathbb{P}_{x \sim \rho}[x \notin \mathcal{X}_{\text{span}}(\varepsilon_{\text{span}})] \lesssim \frac{1}{T_{\text{prompt}}}$.

4. Training-Time Interventions Cannot Be Computationally Efficient

SpannerSampling (Algorithm 3) leverages inference-time computation with the pre-trained model to reduce the effective search space for exploration. As a result, the algorithm is *improper* in the sense that it does not use linear softmax policies $\pi_{\theta} \in \Pi$ to draw the responses for which it queries the reward oracle; this is true for the spanner phase (the algorithm samples $y \sim \pi_{\text{ref}}(\cdot|x)$ properly, but adaptively chooses whether or not to query the reward oracle y), and for the exploration phase (due to the use of truncation and rejection sampling). We contrast this with the notion of *proper exploration*.

Definition 4.1 (Proper alignment algorithm). *An online alignment algorithm is proper if, for each $t \in [T_{\text{prompt}}]$ and $i \in [N]$, the algorithm queries the reward oracle with $y_i^t \sim \pi_{\theta_i^t}(\cdot|x^t)$ for $\theta_i^t \in \Theta$.⁹*

Proper algorithms are closely related to the notion of *training-time* interventions for exploration, in the sense that any algorithm that computes exploratory policies π^t by solving $\theta = \arg \min_{\theta \in \Theta} L_{\mathcal{D}}(\theta)$ for some loss function $L_{\mathcal{D}}(\pi)$ that depends on the dataset \mathcal{D} collected so far will inevitably be proper in the sense of Definition 4.1—no matter how clever we are about designing the loss. This includes OnlineDPO [32], XPO [94, 13], and many others [106, 56, 28]. We show that under the Exponential Time Hypothesis (ETH), no such algorithm can be simultaneously data-efficient and computationally efficient.

Theorem 4.1 (Proper algorithms cannot be computationally efficient). *Under the Randomized ETH (Conjecture H.1), there is no proper alignment algorithm, even with a strong oracle, that (i) has $T_{\text{data}}(\varepsilon, \delta) \leq \text{poly}(d, \beta^{-1}, \varepsilon^{-1}, \delta^{-1})$ and $T_{\text{comp}}(\varepsilon, \delta) \leq \text{poly}(d, \exp(\beta^{-1}), \varepsilon^{-1}, \delta^{-1})$ under Assumption 2.1 (with $R_{\max} = 1$, $B = \sqrt{d}$),¹⁰ and (ii) has runtime $\text{poly}(d, \exp(\beta^{-1}), \varepsilon^{-1}, \delta^{-1})$.*

Note that Definition 4.1 does not require the final output policy $\hat{\pi}$ to be proper; the restriction is solely on the policies used to *explore*. For contrast, we recall that since $C_{\text{cov}}(\pi_{\beta}^*) \leq \exp(\beta^{-1})$ when $R_{\max} = 1$, SpannerSampling achieves $T_{\text{data}}(\varepsilon, \delta) \leq \text{poly}(d, \beta^{-1}, \varepsilon^{-1}, \delta^{-1})$ and $T_{\text{comp}}(\varepsilon, \delta) \leq$

9. The parameter θ_i^t may be chosen adaptively based on the previously sampled responses and rewards.

10. Concretely, we use the parameter set $\Theta = \{\theta \in \mathbb{R}^d \mid \|\theta\|_{\infty} \leq 1\}$.

$\text{poly}(d, \exp(\beta^{-1}), \varepsilon^{-1}, \delta^{-1})$ under the conditions of [Theorem 4.1](#), and does so with time complexity $\text{poly}(d, \exp(\beta^{-1}), \varepsilon^{-1}, \delta^{-1})$. [Theorem 4.1](#) shows that no proper alignment algorithm can achieve polynomial runtime in a similar fashion. In particular, while XPO [94] achieves $T_{\text{data}}(\varepsilon, \delta) \leq \text{poly}(d, \beta^{-1}, \varepsilon^{-1}, \delta^{-1})$, [Theorem 4.1](#) implies that it cannot be implemented efficiently for linear softmax policies. This answers a question raised by Xie et al. [94], and gives a separation between training-time interventions and algorithms like `SpannerSampling` that use additional inference-time computation to explore improperly. Like [Theorem 2.1](#), this lower bound uses only a single prompt. To prove [Theorem 4.1](#), we reduce from the Max- k -DNF problem, embedding a k -DNF formula in $\phi(x, y)$ so that responses correspond to clauses, and embedding a maximally satisfying assignment in the hidden parameter θ^* . Our construction ensures that any *proper* exploration policy π_θ places all but a vanishing fraction of mass on a “null” response $y_0 = \mathbf{0}$ (which is useless for gathering reward information) unless θ corresponds to an assignment that satisfies a large fraction of clauses. Directly finding such a θ requires (approximately) maximizing the underlying k -DNF formula, and the assumptions $T_{\text{data}}(\varepsilon, \delta) \leq \text{poly}(d, \beta^{-1}, \varepsilon^{-1}, \delta^{-1})$ and $T_{\text{comp}}(\varepsilon, \delta) \leq \text{poly}(d, \exp(\beta^{-1}), \varepsilon^{-1}, \delta^{-1})$ ensure that we will not sample a non-null response $y_0 \neq \mathbf{0}$ by chance (which could reveal information about θ^*); from here, the hardness follows. Interestingly, we use hardness of approximation for Max- k -DNF in a somewhat non-standard parameter regime ([Appendix H.5](#)).

5. Autoregressive Policies and Computational Benefits of Multi-Turn Exploration

Our motivating example ([Remark 1.1](#)) is the autoregressive setting where $\mathcal{Y} = \mathcal{A}^H$ for a token space \mathcal{A} and horizon H (so that responses $y = (a_1, \dots, a_H)$ correspond to sequences of tokens), and where π_{ref} is explicitly represented as an autoregressive policy of the form $\pi_{\text{ref}}(y \mid x) = \pi_{\text{ref}}(a_{1:H} \mid x) = \prod_{h=1}^H \pi_{h, \text{ref}}(a_h \mid x, a_{1:h-1})$. In what follows, we show how to specialize `SpannerSampling` to this setting, then derive algorithms with *improved* computational efficiency by alternatively viewing this as a reinforcement learning problem in a *token-level MDP* where actions correspond to tokens [71].

5.1. Autoregressive Softmax Policies: Representational Issues and `SpannerSampling`

When the base policy π_{ref} is autoregressive, it is natural to learn a policy with the same autoregressive structure. We consider the class $\Pi_{\text{auto}} := \{\pi_\theta^{\text{auto}} = (\pi_{1, \theta}^{\text{auto}}, \dots, \pi_{H, \theta}^{\text{auto}}) \mid \theta_h \in \Theta_h \forall h\}$ of autoregressive linear softmax policies given by

$$\pi_{h, \theta}^{\text{auto}}(a_h \mid x, a_{1:h-1}) = \frac{\pi_{h, \text{ref}}(a_h \mid x, a_{1:h-1}) \exp(\beta^{-1} \langle \theta_h, \phi_h(x, a_{1:h}) \rangle)}{\sum_{a' \in \mathcal{A}} \pi_{h, \text{ref}}(a' \mid x, a_{1:h-1}) \exp(\beta^{-1} \langle \theta_h, \phi_h(x, a_{1:h-1}, a') \rangle)}, \quad (11)$$

with $\pi_\theta^{\text{auto}}(a_{1:H} \mid x) := \prod_{h=1}^H \pi_{h, \theta}^{\text{auto}}(a_h \mid x, a_{1:h-1})$; we assume $\theta_h \in \Theta_h \subset \mathbb{R}^d$ with $\|\theta_h\| \leq B$ and $\|\phi_h(x, a_{1:h})\| \leq 1$, where each Θ_h is convex. This parameterization corresponds to a standard deep autoregressive model (e.g., GPT-2 architecture) in which the weights for all but the last layer are frozen [69]. For this setting, we use the following computational oracle, which asserts that we can sample from each conditional policy efficiently.

Definition 5.1 (Autoregressive sampling oracle). **Setting I (strong oracle):** *In one query, the learner proposes a prompt $x \in \mathcal{X}$, layer $h \in [H]$, prefix $a_{1:h-1} \in \mathcal{A}^{h-1}$, and parameter $\theta_h \in \Theta_h$, and receives a conditional sample $a_h \sim \pi_{h, \theta}^{\text{auto}}(\cdot \mid x, a_{1:h-1})$ and the corresponding feature $\phi_h(x, a_{1:h})$.* **Setting II (weak oracle):** *In one query, the learner proposes a prompt $x \in \mathcal{X}$, layer $h \in [H]$, and*

prefix $a_{1:h-1} \in \mathcal{A}^{h-1}$, and receives a conditional sample $a_h \sim \pi_{h,\text{ref}}(\cdot \mid x, a_{1:h-1})$ and corresponding feature $\phi_h(x, a_{1:h})$. We let $T_{\text{comp}}^{\text{auto}}$ denote the total number of autoregressive sampling queries.¹¹

Efficient conditional sampling is arguably the defining property of autoregressive models, so we view this as a minimal assumption. As before, our algorithmic results only use the weak oracle (sampling from π_{ref}), but the strong oracle will be useful for discussion. For rewards, we remain in the setup of [Section 1.1](#): For each prompt x^t for $t \in [T_{\text{prompt}}]$, the algorithm can query the reward oracle for up to N responses $y_1^t, \dots, y_N^t \in \mathcal{A}^H$.

Representational issues. To make use of the class Π_{auto} , we need to assume that $\pi_{\beta}^* \in \Pi_{\text{auto}}$ ([Assumption 1.1](#)). Perhaps the simplest setting where we might hope for this is when rewards are linear:

$$r^*(x, y) = \sum_{h=1}^H \langle \theta_h^*, \phi_h(x, a_{1:h}) \rangle \quad (12)$$

for some $\theta_h^* \in \Theta_h$. Here, the optimal KL-regularized policy π_{β}^* under [Eq. \(12\)](#) setting takes the form¹²

$$\pi_{\theta^*}^{\text{seq}}(a_{1:H} \mid x) := \frac{\pi_{\text{ref}}(a_{1:H} \mid x) \exp\left(\beta^{-1} \sum_{h=1}^H \langle \theta_h^*, \phi_h(x, a_{1:h}) \rangle\right)}{\sum_{(a'_1, \dots, a'_H) \in \mathcal{A}^H} \pi_{\text{ref}}(a'_{1:H} \mid x) \exp\left(\beta^{-1} \sum_{h=1}^H \langle \theta_h^*, \phi_h(x, a'_{1:h}) \rangle\right)}.$$

This corresponds to the linear softmax policy in [Definition 1.1](#) with sequence-level feature map $\phi^{\text{seq}}(x, a_{1:H}) := (\phi_1(x, a_1), \dots, \phi_H(x, a_{1:H})) \in \mathbb{R}^{dH}$ and parameter space $\Theta^{\text{seq}} := (\Theta_1, \dots, \Theta_H) \subset \mathbb{R}^{dH}$ (the natural policy class is $\Pi_{\text{seq}} := \{\pi_{\theta}^{\text{seq}} \mid \theta \in \Theta^{\text{seq}}\}$). Unfortunately, *sequence-level* linear softmax policies of this type cannot be represented as autoregressive linear softmax policies in general; that is, there may not exist any $\theta = (\theta_1, \dots, \theta_H)$ such that $\pi_{\theta^*}^{\text{seq}} = \pi_{\theta}^{\text{auto}}$ —see [Proposition I.1](#).¹³

Applying SpannerSampling. Even though autoregressive realizability may not hold under [Eq. \(12\)](#), we can still apply SpannerSampling efficiently under the sequence-level realizability assumption that $\pi_{\beta}^* \in \Pi_{\text{seq}}$ (which is implied by [Eq. \(12\)](#)). In particular, a weak autoregressive oracle ([Definition 5.1](#)) immediately gives a weak sequence-level sampling oracle ([Definition 2.1](#)) with $T_{\text{comp}}^{\text{auto}} \leq H \cdot T_{\text{comp}}$.

Corollary 5.1. Suppose [Assumption 1.1](#) is satisfied for Π_{seq} and $\langle \theta^*, \phi^{\text{seq}}(x, a_{1:H}) \rangle \in [0, R_{\text{max}}]$. SpannerSampling learns a policy with $\mathbb{E}_{\hat{\pi} \sim \text{unif}(\hat{\pi}^1, \dots, \hat{\pi}^{T_{\text{exp}}})} [J_{\beta}(\pi_{\beta}^*) - J_{\beta}(\hat{\pi})] \leq \varepsilon$ with probability at least $1 - \delta$ when configured appropriately, and does so with:

$$T_{\text{data}}(\varepsilon, \delta) = \tilde{O}\left(\frac{R_{\text{max}}^2}{\beta}\right) \cdot \frac{d^2 H^2 \log^2(\delta^{-1})}{\min\{\varepsilon, \beta\}}, \quad \text{and} \quad T_{\text{comp}}^{\text{auto}}(\varepsilon, \delta) = \tilde{O}\left(C_{\text{cov}}(\pi_{\beta}^*) \cdot \frac{H R_{\text{max}}^2}{\beta^2}\right) \cdot T_{\text{data}}^2(\varepsilon, \delta).$$

For this result, the fact that SpannerSampling only uses a *weak* sequence-level sampling oracle ([Definition 2.1](#)) is crucial: due to aforementioned representational issues, a strong sequence-level sampling oracle (sampling from $\pi_{\theta}^{\text{seq}}$ for $\theta \in \Theta^{\text{seq}}$) cannot necessarily be simulated by even a strong autoregressive oracle (sampling from $\pi_{h,\theta}^{\text{auto}}$).

11. Technically, our algorithm results also require query access to ϕ_h for a fixed reference action—see [Remark J.4](#).

12. By the chain rule, the sequence-level KL-regularizer in [Eq. \(1\)](#) is a equivalent to a sum of per-action regularizers ([Eq. \(69\)](#)).

13. π_{β}^* can always be represented as an autoregressive softmax policy applied to a certain *KL-regularized value function* $Q_{h,\beta}^*$ —see [Eq. \(72\)](#)—but is not necessarily a *linear* softmax unless $Q_{h,\beta}^*$ itself is linear.

5.2. Improving Computational Efficiency through Multi-Turn Exploration

The guarantee in [Corollary 5.1](#) depends on the sequence-level coverage coefficient $C_{\text{cov}}(\pi_\beta^*)$ for π_β^* . While various works have shown that existing pre-trained models may exhibit favorable coverage for tasks of interest [\[8, 78, 93\]](#), it is natural to ask whether we can improve the computational efficiency further, perhaps by exploiting the autoregressive structure of π_{ref} . To this end, we will make the autoregressive realizability assumption that $\pi_\beta^* \in \Pi_{\text{auto}}$ (i.e., there exists $\theta^* = (\theta_1^*, \dots, \theta_H^*)$ such that $\pi_{\theta^*}^{\text{auto}} = \pi_\beta^*$). As discussed above, this is not implied by sequence-level realizability in general, but we will show that when it holds, we can achieve runtime guarantees that scale with the following *conditional* (or, token-level/action-level) coverage coefficient:

$$C_{\text{cond}}(\pi_\beta^*) := \max_{h \in [H]} \sup_{x \in \mathcal{X}} \sup_{(a_1, \dots, a_h) \in \mathcal{A}^h} \frac{\pi_{h,\beta}^*(a_h \mid x, a_{1:h-1})}{\pi_{h,\text{ref}}(a_h \mid x, a_{1:h-1})}. \quad (13)$$

This coefficient can exponentially improve $C_{\text{cov}}(\pi_\beta^*)$; we can have $C_{\text{cond}}(\pi_\beta^*) \leq 2$, yet $C_{\text{cov}}(\pi_\beta^*) \geq 2^H$.

MultiTurnSpannerSampling. We introduce a *multi-turn* counterpart to SpannerSampling, MTSS ([Algorithm 4](#) in [Appendix J](#)). MTSS learns a policy in a multi-turn (dynamic programming) fashion by fitting softmax policies for each layer $h = H, \dots, 1$, while growing *core-sets* of informative sub-sequences $(x, a_{1:h})$ for which the algorithm can confidently estimate the parameter θ_h^* (generalizing the notion of spanner used in SpannerSampling). The use of dynamic programming in the algorithm is motivated by the fact that whenever π_β^* is autoregressive (i.e., $\pi_\beta^* \in \Pi_{\text{auto}}$), a certain *KL-regularized* state-action value function $Q_\beta^*(x, a_{1:h})$ is linear up to an action-independent shift. See [Appendices I and J](#) for a detailed overview.

Theorem 5.1 (Guarantee for MTSS; special case of [Theorem J.1](#)). *Suppose [Assumption 1.1](#) is satisfied for the class Π_{auto} . MTSS, when configured appropriately, returns $\hat{\pi}$ such that $J_\beta(\pi_{\theta^*}^{\text{auto}}) - J_\beta(\hat{\pi}_{1:H}) \leq \varepsilon$ with probability at least $1 - \delta$, and does so with $T_{\text{data}}(\varepsilon, \delta) \leq \text{poly}(d, H, B, \varepsilon^{-1}, \log(\delta^{-1}))$ reward queries and $T_{\text{comp}}^{\text{auto}}(\varepsilon, \delta) \leq \text{poly}\left(C_{\text{cond}}(\pi_\beta^*), T_{\text{data}}(\varepsilon, \delta)\right)$ (weak) autoregressive sampling queries.*

As discussed above, the action-level coverage coefficient $C_{\text{cond}}(\pi_\beta^*)$ in this result can be exponentially smaller than the sequence-level coverage coefficient. We view the assumption that $\pi_\beta^* \in \Pi_{\text{auto}}$ as a fairly minimal representational assumption for working with autoregressive policies (i.e., for MTSS to learn efficiently, all we require is that the base policy π_{ref} and the optimal policy π_β^* are autoregressive), analogous to the classical notion of Q^* -realizability in linearly-parameterized RL [\[51, 103, 91\]](#).

We remark that while our exposition focuses on the token-level MDP, the results above also apply to the more realistic setting where each action a_h represents a sub-sequence of tokens (e.g., a lemma in a proof) rather than a single token (e.g., Xiong et al. [\[96\]](#)). Here, the fact that the runtime and sample complexity for MTSS are independent of $|\mathcal{A}|$ is essential.

Connection to reinforcement learning with linear Q^* . MTSS can be applied beyond the token-level MDP formulation above: Our presentation and guarantees for the algorithm in [Part III](#) of the appendix apply to *any* MDP for which π_β^* is an “action-level” linear softmax policy (a generalization of the assumption that the optimal KL-regularized value function Q_β^* is linear), provided that resetting to previously visited states is allowed. In this regard, the algorithm can be viewed as a counterpart to a body of work which shows that MDPs with linear Q^* and state-action gap Δ can be learned under reset access [\[51, 103, 91\]](#); the regularization parameter β plays a role analogous to the gap Δ in facilitating favorable error propagation in our analysis.

See [Part III](#) of the appendix for a formal presentation of the MTSS algorithm and guarantees for learning autoregressive linear softmax policies in general MDPs (generalizing [Theorem 5.1](#))

Future work. Due to space constraints, discussion and future work is deferred to [Appendix A](#).

References

- [1] Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. In *Advances in Neural Information Processing Systems*, 2011.
- [2] Alekh Agarwal, Nan Jiang, Sham M Kakade, and Wen Sun. Reinforcement learning: Theory and algorithms. <https://rltheorybook.github.io/>, 2019. Version: January 31, 2022.
- [3] Baruch Awerbuch and Robert Kleinberg. Online linear optimization and adaptive routing. *Journal of Computer and System Sciences*, 74(1):97–114, 2008.
- [4] James Bagnell, Sham M Kakade, Jeff Schneider, and Andrew Ng. Policy search by dynamic programming. *Advances in neural information processing systems*, 16, 2003.
- [5] Adam Block and Yury Polyanskiy. The sample complexity of approximate rejection sampling with applications to smoothed online learning. In *The Thirty Sixth Annual Conference on Learning Theory*, pages 228–273. PMLR, 2023.
- [6] Avinandan Bose, Zhihan Xiong, Aadirupa Saha, Simon Shaolei Du, and Maryam Fazel. Hybrid preference optimization for alignment: Provably faster convergence rates by combining offline preferences with online exploration. *arXiv preprint arXiv:2412.10616*, 2024.
- [7] Ralph Allan Bradley and Milton E Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952.
- [8] Bradley Brown, Jordan Juravsky, Ryan Ehrlich, Ronald Clark, Quoc V Le, Christopher Ré, and Azalia Mirhoseini. Large language monkeys: Scaling inference compute with repeated sampling. *arXiv:2407.21787*, 2024.
- [9] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, 2020.
- [10] Sébastien Bubeck, Nicolo Cesa-Bianchi, and Sham M Kakade. Towards minimax policies for online linear optimization with bandit feedback. In *Conference on Learning Theory*, pages 41–1. JMLR Workshop and Conference Proceedings, 2012.
- [11] Chris Calabro, Russell Impagliazzo, Valentine Kabanets, and Ramamohan Paturi. The complexity of unique k-sat: An isolation lemma for k-cnfs. *Journal of Computer and System Sciences*, 74(3):386–393, 2008.

- [12] Tongyi Cao and Akshay Krishnamurthy. Disagreement-based combinatorial pure exploration: Sample complexity bounds and an efficient algorithm. In *Conference on Learning Theory*, pages 558–588. PMLR, 2019.
- [13] Shicong Cen, Jincheng Mei, Katayoon Goshvadi, Hanjun Dai, Tong Yang, Sherry Yang, Dale Schuurmans, Yuejie Chi, and Bo Dai. Value-incentivized preference optimization: A unified approach to online and offline rlhf, 2024.
- [14] Siu On Chan. Approximation resistance from pairwise-independent subgroups. *Journal of the ACM (JACM)*, 63(3):1–32, 2016.
- [15] Jonathan D Chang, Wenhao Zhan, Owen Oertell, Kianté Brantley, Dipendra Misra, Jason D Lee, and Wen Sun. Dataset reset policy optimization for rlhf. *arXiv preprint arXiv:2404.08495*, 2024.
- [16] Lijie Chen, Anupam Gupta, Jian Li, Mingda Qiao, and Ruosong Wang. Nearly optimal sampling algorithms for combinatorial pure exploration. In *Conference on Learning Theory*, pages 482–534. PMLR, 2017.
- [17] Ruizhe Chen, Xiaotian Zhang, Meng Luo, Wenhao Chai, and Zuozhu Liu. Pad: Personalized alignment at decoding-time. *arXiv:2410.04070*, 2024.
- [18] Xiaoyu Chen, Han Zhong, Zhuoran Yang, Zhaoran Wang, and Liwei Wang. Human-in-the-loop: Provably efficient preference-based reinforcement learning with general function approximation. In *International Conference on Machine Learning*, pages 3773–3793. PMLR, 2022.
- [19] Varsha Dani, Thomas P Hayes, and Sham M Kakade. Stochastic linear optimization under bandit feedback. In *Conference on Learning Theory (COLT)*, 2008.
- [20] Christoph Dann, Nan Jiang, Akshay Krishnamurthy, Alekh Agarwal, John Langford, and Robert E Schapire. On oracle-efficient PAC RL with rich observations. In *Advances in neural information processing systems*, pages 1422–1432, 2018.
- [21] Nirjhar Das, Souradip Chakraborty, Aldo Pacchiano, and Sayak Ray Chowdhury. Provably sample efficient rlhf via active preference optimization. *arXiv preprint arXiv:2402.10500*, 2024.
- [22] DeepSeek-AI. DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning. 2025. URL https://github.com/deepseek-ai/DeepSeek-R1/blob/main/DeepSeek_R1.pdf.
- [23] Simon S Du, Sham M Kakade, Ruosong Wang, and Lin F Yang. Is a good representation sufficient for sample efficient reinforcement learning? In *International Conference on Learning Representations*, 2020.
- [24] Yihan Du, Anna Winnicki, Gal Dalal, Shie Mannor, and R Srikant. Exploration-driven policy optimization in RLHF: Theoretical insights on efficient data utilization. *arXiv preprint arXiv:2402.10342*, 2024.

- [25] Adam Fisch, Jacob Eisenstein, Vicky Zayats, Alekh Agarwal, Ahmad Beirami, Chirag Nagpal, Pete Shaw, and Jonathan Berant. Robust preference optimization through reward model distillation. *arXiv:2405.19316*, 2024.
- [26] Dylan J Foster and Alexander Rakhlin. Foundations of reinforcement learning and interactive decision making. *arXiv:2312.16730*, 2023.
- [27] Dylan J Foster, Sham M Kakade, Jian Qian, and Alexander Rakhlin. The statistical complexity of interactive decision making. *arXiv:2112.13487*, 2021.
- [28] Zhaolin Gao, Jonathan D Chang, Wenhao Zhan, Owen Oertell, Gokul Swamy, Kianté Brantley, Thorsten Joachims, J Andrew Bagnell, Jason D Lee, and Wen Sun. REBEL: Reinforcement learning via regressing relative rewards. *arXiv:2404.16767*, 2024.
- [29] Zhaolin Gao, Wenhao Zhan, Jonathan D Chang, Gokul Swamy, Kianté Brantley, Jason D Lee, and Wen Sun. Regressing the relative future: Efficient policy optimization for multi-turn rlhf. *arXiv preprint arXiv:2410.04612*, 2024.
- [30] Noah Golowich, Ankur Moitra, and Dhruv Rohatgi. Exploration is harder than prediction: Cryptographically separating reinforcement learning from supervised learning. *arXiv preprint arXiv:2404.03774*, 2024.
- [31] Google. Palm 2 technical report. *arXiv:2305.10403*, 2023.
- [32] Shangmin Guo, Biao Zhang, Tianlin Liu, Tianqi Liu, Misha Khalman, Felipe Llinares, Alexandre Rame, Thomas Mesnard, Yao Zhao, Bilal Piot, Johan Ferret, and Mathieu Blondel. Direct language model alignment from online AI feedback. *arXiv:2402.04792*, 2024.
- [33] Shibo Hao, Yi Gu, Haodi Ma, Joshua Hong, Zhen Wang, Daisy Wang, and Zhiting Hu. Reasoning with language model is planning with world model. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 8154–8173, 2023.
- [34] Elad Hazan and Zohar Karnin. Volumetric spanners: An efficient exploration basis for learning. *The Journal of Machine Learning Research*, 17(1):4062–4095, 2016.
- [35] Audrey Huang, Adam Block, Dylan J Foster, Dhruv Rohatgi, Cyril Zhang, Max Simchowitz, Jordan T Ash, and Akshay Krishnamurthy. Self-improvement in language models: The sharpening mechanism. *arXiv preprint arXiv:2412.01951*, 2024.
- [36] Audrey Huang, Wenhao Zhan, Tengyang Xie, Jason D Lee, Wen Sun, Akshay Krishnamurthy, and Dylan J Foster. Correcting the mythos of KL-regularization: Direct alignment without overoptimization via Chi-squared Preference Optimization. *arXiv:2407.13399*, 2024.
- [37] Xiang Ji, Sanjeev Kulkarni, Mengdi Wang, and Tengyang Xie. Self-play with adversarial critic: Provable and scalable offline alignment for language models. *arXiv:2406.04274*, 2024.
- [38] Nan Jiang, Akshay Krishnamurthy, Alekh Agarwal, John Langford, and Robert E Schapire. Contextual decision processes with low Bellman rank are PAC-learnable. In *International Conference on Machine Learning*, 2017.

- [39] Chi Jin, Qinghua Liu, and Sobhan Miryoosefi. Bellman Eluder dimension: New rich classes of RL problems, and sample-efficient algorithms. *Advances in Neural Information Processing Systems*, 2021.
- [40] Yuu Jinnai, Tetsuro Morimura, Kaito Ariu, and Kenshi Abe. Regularized best-of-n sampling to mitigate reward hacking for language model alignment. *arXiv:2404.01054*, 2024.
- [41] Daniel Kane, Sihan Liu, Shachar Lovett, and Gaurav Mahajan. Computational-statistical gap in reinforcement learning. In *Conference on Learning Theory*, pages 1282–1302. PMLR, 2022.
- [42] Julian Katz-Samuels, Lalit Jain, Kevin G Jamieson, et al. An empirical process approach to the union bound: Practical algorithms for combinatorial and linear bandits. *Advances in Neural Information Processing Systems*, 33, 2020.
- [43] Amirhossein Kazemnejad, Milad Aghajohari, Eva Portelance, Alessandro Sordoni, Siva Reddy, Aaron Courville, and Nicolas Le Roux. Vineppo: Unlocking rl potential for llm reasoning through refined credit assignment. *arXiv preprint arXiv:2410.01679*, 2024.
- [44] Michael Kearns. Efficient noise-tolerant learning from statistical queries. *Journal of the ACM*, 1998.
- [45] Saeed Khaki, JinJin Li, Lan Ma, Liu Yang, and Prathap Ramachandra. Rs-dpo: A hybrid rejection sampling and direct preference optimization method for alignment of large language models. *arXiv preprint arXiv:2402.10038*, 2024.
- [46] Maxim Khanov, Jirayu Burapachee, and Yixuan Li. Args: Alignment as reward-guided search. *arXiv:2402.01694*, 2024.
- [47] Aviral Kumar, Vincent Zhuang, Rishabh Agarwal, Yi Su, John D Co-Reyes, Avi Singh, Kate Baumli, Shariq Iqbal, Colton Bishop, Rebecca Roelofs, et al. Training language models to self-correct via reinforcement learning. *arXiv preprint arXiv:2409.12917*, 2024.
- [48] Tze Leung Lai and Herbert Robbins. Asymptotically efficient adaptive allocation rules. *Advances in Applied Mathematics*, 6(1):4–22, 1985.
- [49] Tor Lattimore and Csaba Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020.
- [50] Tor Lattimore, Csaba Szepesvari, and Gellert Weisz. Learning with good feature representations in bandits and in rl with a generative model. In *International Conference on Machine Learning*, pages 5662–5670. PMLR, 2020.
- [51] Gen Li, Yuxin Chen, Yuejie Chi, Yuantao Gu, and Yuting Wei. Sample-efficient reinforcement learning is feasible for linearly realizable mdps with limited revisiting. *Advances in Neural Information Processing Systems*, 2021.
- [52] Zihao Li, Zhuoran Yang, and Mengdi Wang. Reinforcement learning with human feedback: Learning dynamic choices via pessimism. *arXiv:2305.18438*, 2023.

- [53] Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let’s verify step by step. *arXiv preprint arXiv:2305.20050*, 2023.
- [54] Tianlin Liu, Shangmin Guo, Leonardo Bianco, Daniele Calandriello, Quentin Berthet, Felipe Llinares, Jessica Hoffmann, Lucas Dixon, Michal Valko, and Mathieu Blondel. Decoding-time realignment of language models. *arXiv:2402.02992*, 2024.
- [55] Tianqi Liu, Yao Zhao, Rishabh Joshi, Misha Khalman, Mohammad Saleh, Peter J Liu, and Jialu Liu. Statistical rejection sampling improves preference optimization. *arXiv preprint arXiv:2309.06657*, 2023.
- [56] Zhihan Liu, Miao Lu, Shenao Zhang, Boyi Liu, Hongyi Guo, Yingxiang Yang, Jose Blanchet, and Zhaoran Wang. Provably mitigating overoptimization in RLHF: Your SFT loss is implicitly an adversarial regularizer. *arXiv:2405.16436*, 2024.
- [57] László Lovász and Santosh Vempala. Fast algorithms for logconcave functions: Sampling, rounding, integration and optimization. In *Symposium on Foundations of Computer Science*, 2006.
- [58] Sadhika Malladi, Alexander Wettig, Dingli Yu, Danqi Chen, and Sanjeev Arora. A kernel-based view of language model fine-tuning. In *International Conference on Machine Learning*, pages 23610–23641. PMLR, 2023.
- [59] Zakaria Mhammedi. Sample and oracle efficient reinforcement learning for mdps with linearly-realizable value functions, 2024. URL <https://arxiv.org/abs/2409.04840>.
- [60] Zakaria Mhammedi, Dylan J Foster, and Alexander Rakhlin. The power of resets in online reinforcement learning. *arXiv preprint arXiv:2404.15417*, 2024.
- [61] Arkadii Nemirovski, David Borisovich Yudin, and Edgar Ronald Dawson. *Problem complexity and method efficiency in optimization*. Wiley, 1983.
- [62] Ellen Novoseller, Yibing Wei, Yanan Sui, Yisong Yue, and Joel Burdick. Dueling posterior sampling for preference-based reinforcement learning. In *Conference on Uncertainty in Artificial Intelligence*, pages 1029–1038. PMLR, 2020.
- [63] OpenAI. GPT-4 technical report. *arXiv:2303.08774*, 2023.
- [64] OpenAI. Introducing openai o1. *Blog*, 2024. URL <https://openai.com/o1/>.
- [65] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 2022.
- [66] Aldo Pacchiano, Aadirupa Saha, and Jonathan Lee. Dueling RL: reinforcement learning with trajectory preferences. *arXiv preprint arXiv:2111.04850*, 2021.

- [67] Yury Polyanskiy and Yihong Wu. *Information Theory: From Coding to Learning*. Cambridge University Press, 2025.
- [68] Yuxiao Qu, Tianjun Zhang, Naman Garg, and Aviral Kumar. Recursive introspection: Teaching language model agents how to self-improve. *arXiv preprint arXiv:2407.18219*, 2024.
- [69] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.
- [70] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 2023.
- [71] Rafael Rafailov, Joey Hejna, Ryan Park, and Chelsea Finn. From r to Q^* : Your language model is secretly a Q-function. *arXiv:2404.12358*, 2024.
- [72] Dhruv Rohatgi, Adam Block, Audrey Huang, Akshay Krishnamurthy, and Dylan J. Foster. Computational-statistical tradeoffs at the next-token prediction barrier: Autoregressive and imitation learning under misspecification. *arXiv preprint arXiv:2502.12465*, 2025.
- [73] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv:1707.06347*, 2017.
- [74] Amrith Setlur, Saurabh Garg, Xinyang Geng, Naman Garg, Virginia Smith, and Aviral Kumar. RL on incorrect synthetic data scales the efficiency of llm math reasoning by eight-fold. *arXiv preprint arXiv:2406.14532*, 2024.
- [75] Amrith Setlur, Chirag Nagpal, Adam Fisch, Xinyang Geng, Jacob Eisenstein, Rishabh Agarwal, Alekh Agarwal, Jonathan Berant, and Aviral Kumar. Rewarding progress: Scaling automated process verifiers for llm reasoning. *arXiv preprint arXiv:2410.08146*, 2024.
- [76] Ruizhe Shi, Yifang Chen, Yushi Hu, ALisa Liu, Noah Smith, Hannaneh Hajishirzi, and Simon Du. Decoding-time language model alignment with multiple objectives. *arXiv:2406.18853*, 2024.
- [77] Ruizhe Shi, Runlong Zhou, and Simon S Du. The crucial role of samplers in online direct preference optimization. *arXiv preprint arXiv:2409.19605*, 2024.
- [78] Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling LLM test-time compute optimally can be more effective than scaling model parameters. *arXiv:2408.03314*, 2024.
- [79] Daniil Tiapkin, Denis Belomestny, Daniele Calandriello, Eric Moulines, Alexey Naumov, Pierre Perrault, Michal Valko, and Pierre Menard. Demonstration-regularized rl. In *The Twelfth International Conference on Learning Representations*, 2024.
- [80] Tomasz Tkocz. An upper bound for spherical caps. *The American Mathematical Monthly*, 119(7):606–607, 2012.

- [81] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models. *arXiv:2307.09288*, 2023.
- [82] Hoang Tran, Chris Glaze, and Braden Hancock. Iterative dpo alignment. *Technical report*, 2023. URL <https://huggingface.co/snorkelai/Snorkel-Mistral-PairRM-DPO>.
- [83] Chaoqi Wang, Yibo Jiang, Chenghao Yang, Han Liu, and Yuxin Chen. Beyond reverse kl: Generalizing direct preference optimization with diverse divergence constraints. In *The Twelfth International Conference on Learning Representations*, 2024.
- [84] Haoxiang Wang, Yong Lin, Wei Xiong, Rui Yang, Shizhe Diao, Shuang Qiu, Han Zhao, and Tong Zhang. Arithmetic control of llms for diverse user preferences: Directional preference alignment with multi-objective rewards. *arXiv preprint arXiv:2402.18571*, 2024.
- [85] Haoxiang Wang, Wei Xiong, Tengyang Xie, Han Zhao, and Tong Zhang. Interpretable preferences via multi-objective reward modeling and mixture-of-experts. *arXiv preprint arXiv:2406.12845*, 2024.
- [86] Yuanhao Wang, Ruosong Wang, and Sham M Kakade. An exponential lower bound for linearly-realizable MDPs with constant suboptimality gap. *Neural Information Processing Systems (NeurIPS)*, 2021.
- [87] Yuanhao Wang, Qinghua Liu, and Chi Jin. Is RLHF more difficult than standard RL? *arXiv preprint arXiv:2306.14111*, 2023.
- [88] Zhilin Wang, Yi Dong, Jiaqi Zeng, Virginia Adams, Makesh Narsimhan Sreedhar, Daniel Egert, Olivier Delalleau, Jane Polak Scowcroft, Neel Kant, Aidan Swope, et al. Helpsteer: Multi-attribute helpfulness dataset for steerlm. *arXiv preprint arXiv:2311.09528*, 2023.
- [89] Zhilin Wang, Yi Dong, Olivier Delalleau, Jiaqi Zeng, Gerald Shen, Daniel Egert, Jimmy J Zhang, Makesh Narsimhan Sreedhar, and Oleksii Kuchaiev. Helpsteer2: Open-source dataset for training top-performing reward models. *arXiv preprint arXiv:2406.08673*, 2024.
- [90] Gellert Weisz, Philip Amortila, Barnabás Janzer, Yasin Abbasi-Yadkori, Nan Jiang, and Csaba Szepesvári. On query-efficient planning in mdps under linear realizability of the optimal state-value function. In *Conference on Learning Theory*, pages 4355–4385. PMLR, 2021.

- [91] Gellért Weisz, András György, Tadashi Kozuno, and Csaba Szepesvári. Confident approximate policy iteration for efficient local planning in q^π -realizable mdps. *Advances in Neural Information Processing Systems*, 35:25547–25559, 2022.
- [92] Runzhe Wu and Wen Sun. Making RL with preference-based feedback efficient via randomization. *arXiv preprint arXiv:2310.14554*, 2023.
- [93] Yangzhen Wu, Zhiqing Sun, Shanda Li, Sean Welleck, and Yiming Yang. An empirical analysis of compute-optimal inference for problem-solving with language models. *arXiv:2408.00724*, 2024.
- [94] Tengyang Xie, Dylan J Foster, Akshay Krishnamurthy, Corby Rosset, Ahmed Awadallah, and Alexander Rakhlin. Exploratory preference optimization: Harnessing implicit Q^* -approximation for sample-efficient RLHF. *arXiv:2405.21046*, 2024.
- [95] Wei Xiong, Hanze Dong, Chenlu Ye, Han Zhong, Nan Jiang, and Tong Zhang. Iterative preference learning from human feedback: Bridging theory and practice for RLHF under KL-constraint. *International Conference on Machine Learning (ICML)*, 2024.
- [96] Wei Xiong, Chengshuai Shi, Jiaming Shen, Aviv Rosenberg, Zhen Qin, Daniele Calandriello, Misha Khalman, Rishabh Joshi, Bilal Piot, Mohammad Saleh, et al. Building math agents with multi-turn iterative preference learning. *arXiv preprint arXiv:2409.02392*, 2024.
- [97] Jing Xu, Andrew Lee, Sainbayar Sukhbaatar, and Jason Weston. Some things are more cringe than others: Preference optimization with the pairwise cringe loss. *arXiv preprint arXiv:2312.16682*, 2023.
- [98] Yichong Xu, Ruosong Wang, Lin Yang, Aarti Singh, and Artur Dubrawski. Preference-based reinforcement learning with finite-time guarantees. *Advances in Neural Information Processing Systems*, 33:18784–18794, 2020.
- [99] Xue Yan, Yan Song, Xidong Feng, Mengyue Yang, Haifeng Zhang, Haitham Bou Ammar, and Jun Wang. Efficient reinforcement learning with large language model priors. *arXiv preprint arXiv:2410.07927*, 2024.
- [100] Yuhong Yang and Andrew R Barron. An asymptotic property of model selection criteria. *IEEE Transactions on Information Theory*, 44(1):95–116, 1998.
- [101] Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. *Advances in Neural Information Processing Systems*, 36, 2024.
- [102] Chenlu Ye, Wei Xiong, Yuheng Zhang, Nan Jiang, and Tong Zhang. A theoretical analysis of Nash learning from human feedback under general KL-regularized preference. *Neural Information Processing Systems (NeurIPS)*, 2024.
- [103] Dong Yin, Botao Hao, Yasin Abbasi-Yadkori, Nevena Lazić, and Csaba Szepesvári. Efficient local planning with linear function approximation. In *International Conference on Algorithmic Learning Theory*, 2022.

- [104] Dong Yin, Sridhar Thiagarajan, Nevena Lazic, Nived Rajaraman, Botao Hao, and Csaba Szepesvari. Sample efficient deep reinforcement learning via local planning. *arXiv preprint arXiv:2301.12579*, 2023.
- [105] Wenhao Zhan, Masatoshi Uehara, Wen Sun, and Jason D Lee. Provable reward-agnostic preference-based reinforcement learning. *arXiv preprint arXiv:2305.18505*, 2023.
- [106] Shenao Zhang, Donghan Yu, Hiteshi Sharma, Ziyi Yang, Shuohang Wang, Hany Hassan, and Zhaoran Wang. Self-exploring language models: Active preference elicitation for online alignment, 2024.
- [107] Heyang Zhao, Chenlu Ye, Quanquan Gu, and Tong Zhang. Sharp analysis for kl-regularized contextual bandits and rlhf. *arXiv preprint arXiv:2411.04625*, 2024.
- [108] Heyang Zhao, Chenlu Ye, Wei Xiong, Quanquan Gu, and Tong Zhang. Logarithmic regret for online kl-regularized reinforcement learning. *arXiv preprint arXiv:2502.07460*, 2025.
- [109] Yifei Zhou, Andrea Zanette, Jiayi Pan, Sergey Levine, and Aviral Kumar. Archer: Training language model agents via hierarchical multi-turn rl. In *International Conference on Machine Learning*, pages 62178–62209. PMLR, 2024.
- [110] Banghua Zhu, Michael Jordan, and Jiantao Jiao. Principled reinforcement learning with human feedback from pairwise or k-wise comparisons. In *International Conference on Machine Learning*, 2023.
- [111] Yinglun Zhu, Dylan J Foster, John Langford, and Paul Mineiro. Contextual bandits with large action spaces: Made practical. In *International Conference on Machine Learning*, pages 27428–27453. PMLR, 2022.

Contents of Appendix

I	Additional Results and Discussion	25
A	Discussion and Future Work	25
B	Additional Related Work	25
B.1	Comparison to Preference-Based Feedback	27
C	Sampling Oracles: Beyond Linear Policies	28
II	Proofs from Sections 2 through 4	30
D	Technical Tools	30
D.1	Tail Bounds	30
D.2	Elliptic Potential	30
D.3	Miscellaneous Lemmas	31
E	Proofs from Section 2	32
F	SoftmaxSampler Algorithm and Guarantees	36
F.1	Proofs	38
G	SpannerSampling Algorithm and Guarantee (Proof of Theorem 3.1)	43
G.1	SpannerSampling Pseudocode	43
G.2	Technical Lemmas	43
G.3	KL-Regularized Regret Decomposition for Truncated Softmax Policies	46
G.4	Proof of Theorem 3.1 (Guarantee for SpannerSampling)	51
H	Proofs from Section 4	60
H.1	Overview of Proof	60
H.2	Proof of Lemma H.1	62
H.3	Proof of Theorem H.1	66
H.4	Proof of Theorem 4.1	67
H.5	Hardness of Approximation for Max- k -DNF	68
III	Multi-Turn Exploration: Learning Autoregressive Softmax Policies	72
I	Preliminaries for Multi-Turn Exploration	72
I.1	MDP Setting and Multi-Turn Reinforcement Learning Framework	73
I.2	Sample Complexity, Computational Oracles, and Coverage	75
J	MTSS Algorithm and Guarantees	77
J.1	MTSS Pseudocode and Overview	77
J.2	Main Guarantee for MTSS (Generalization of Theorem 5.1)	83
K	Guarantee for UncertainStateAction	85

L	Guarantee for FitValue	88
	L.1 Helper Lemmas for FitValue Guarantee	95
	L.2 Guarantee of FitValue for MTSS	98
M	Proof of Theorem J.1	99
N	Technical Lemmas for Multi-Turn Exploration	109

Part I

Additional Results and Discussion

Appendix A. Discussion and Future Work

Our results—via the sampling oracle framework—reveal the computational, statistical, and representational tradeoffs inherent to language model exploration, highlighting the fundamental role of the base model π_{ref} in enabling computational efficiency. We view our results as an initial step toward a computational foundation for language model exploration, and more broadly, for efficient decision making with generative models. To this end, some natural questions are as follows.

Efficient exploration beyond linear softmax policies. While our lower bounds are relevant beyond the linear softmax parameterization, our algorithms are specialized to this setting. Developing algorithms to support general, nonlinear policy parameterizations is perhaps the most important question left by our work. We expect that the basic principle behind our algorithms—expending inference-time computation to identify “representative” responses with which to explore—to be useful more broadly, but the specific notion of spanner used in our results will need to change.¹⁴

Better representations for exploration. Our results in [Section 4](#) show that training-time interventions that produce softmax policies (e.g., modifications to the DPO loss) are insufficient for computationally efficient exploration. This raises the question of whether there exist training-time interventions that induce different policy representations (e.g., based on alternative forms of regularization [83, 36]) that more readily lend themselves to computationally efficient exploration. Our results in [Section 3](#) show that relatively simple modifications to the linear softmax parameterization (e.g., truncation) have benefits for exploration, but are there more general principles beyond the linear setting?

Appendix B. Additional Related Work

In this section we discuss related work not already covered in detail.

Theoretical algorithms for online alignment. There is a large body of work on theoretical algorithms for exploration in online alignment (as well as the more abstract problem of preference-based contextual bandits and RL), but most prior algorithms are not computationally efficient when the response space \mathcal{Y} is large [98, 62, 66, 92, 105, 24, 21, 18, 87, 102, 95]. As discussed earlier, the XPO algorithm of Xie et al. [94] (see also Cen et al. [13], Zhang et al. [106])¹⁵ is perhaps the closest to a satisfactory solution from prior work, as it achieves optimal data efficiency and only accesses the response space through sampling from policies π_θ . However, our results in [Section 4](#) show that the XPO objective cannot be implemented efficiently in general. More broadly, even if the base model π_{ref} has favorable properties such as coverage (in the sense of [Eq. \(3\)](#)), none of the aforementioned algorithms can take advantage of it for improved computational efficiency. In this regard, we view them as making somewhat superficial use of the base policy (i.e., it does not play a role in algorithm design outside of being used to define the KL-regularized RL objective).

14. We expect it to be fairly straightforward to extend our results to accommodate policy classes with bounded eluder dimension, but it is less clear how to address realistic classes based on, e.g., transformers..

15. Cen et al. [13], Zhang et al. [106] concurrently proposed similar algorithms to XPO, but did not provide non-trivial theoretical guarantees (e.g., guarantees that indicate benefits over purely passive exploration).

Many works consider the complementary problem of alignment in *offline* or *hybrid* settings [110, 52, 95, 28, 15, 56, 13, 25, 37, 36, 107]. These works pay for coverage coefficients similar to $C_{\text{cov}}(\pi_\beta^*)$ *statistically* (i.e., $T_{\text{data}}(\varepsilon, \delta) = \Omega(C_{\text{cov}}(\pi_\beta^*))$), and hence are not data-efficient by our definition. One relevant work here is Bose et al. [6], who give a hybrid variant of XP0 which obtains statistical rates tighter than purely offline or online methods, but is still computationally inefficient.

Algorithms that use additional inference-time computation for exploration (e.g., via rejection sampling) [46, 17, 76, 54, 40, 77] or *multi-turn* techniques that proceed at the per-step (e.g., token or sub-sequence) level [53, 68, 47, 75, 74, 96, 43, 109] have been explored empirically, but most results we are aware of do not enjoy sample complexity guarantees. Shi et al. [77] explore the role of various sampling schemes on top of OnLineDPO, but do not give sample complexity guarantees for our setting. Gao et al. [29] provide a multi-turn algorithm with sample complexity guarantees, but it engages in passive exploration and pays for coverage statistically.

Fast rates for regularized regret. Our algorithm SpannerSampling achieves a *fast rate* in the sense that $T_{\text{data}}(\varepsilon, \delta) \lesssim \frac{1}{\beta\varepsilon}$ when $\varepsilon \leq \beta$, improving over the $T_{\text{data}}(\varepsilon, \delta) \lesssim \frac{1}{\varepsilon^2}$ rates found in prior work [95, 94, 13] by exploiting strong convexity of the KL-regularized regret. Recent work of Zhao et al. [107] achieves a similar fast rate, but requires access to an offline dataset satisfying a stringent uniform coverage assumption (and pays for the coverage coefficient statistically), while concurrent work of Zhao et al. [108] achieves fast rates in the purely online setting, but is not computationally efficient in our framework. Also related is the work of Tiapkin et al. [79], which achieves fast rates for regularized regret in tabular and linear MDPs, but is not efficient when the action space is large.

Connection to optimization oracles in bandits and online learning. There is a large body of work on algorithms for linear contextual bandits with large response spaces \mathcal{Y} in which the response space is accessed through an *optimization oracle* which can solve

$$\arg \max_{y \in \mathcal{Y}} \langle \theta, \phi(x, y) \rangle$$

efficiently for any $x \in \mathcal{X}$ and $\theta \in \Theta$ [19, 10, 34, 16, 12, 42, 111]. Our formulation in Definition 2.1 can be viewed as an alternative, sampling-based computational framework for decision making with large response spaces, one which may be of independent interest. Note that while there is a sense in which sampling and optimization are polynomially equivalent when the set $\{\phi(x, y)\}_{y \in \mathcal{Y}}$ is convex [57], they are not equivalent in general.

Our sampling oracle framework is also related to the work of Huang et al. [35], who used a similar abstraction for the complementary problem of language model self-improvement.

Algorithms for reinforcement learning with linear- Q^* . Our multi-turn algorithm, MTSS, can be viewed as a counterpart to a body of work which shows that MDPs with linear Q^* and state-action gap Δ can be learned under reset access [51, 103, 91, 60, 59]. In particular, prior work has shown that RL with linear- Q^* and an action gap Δ is statistically intractable in the episodic RL protocol, but is tractable under reset access [90, 51]. Our results show that the regularization parameter β plays a similar role to the action gap Δ in enabling favorable error propagation, leading to tractability under reset access. While MTSS draws inspiration from the works above—particularly Mhammedi et al. [60], Mhammedi [59]—it requires fairly substantial modifications, both in design and analysis—to (i) leverage KL regularization, and (ii) achieve computational efficiency in the sampling oracle framework.

B.1. Comparison to Preference-Based Feedback

Much of prior work on online alignment focuses on *preference-based feedback*. Here, the protocol is as follows. At each round $t \in [T_{\text{prompt}}]$, we receive a prompt x^t and sample two responses $(y_1^t, y_2^t) \sim \pi^t(\cdot | x^t)$, where π^t denotes the *exploration policy* for round t ; the exploration policy may be represented as a language model, or may correspond to an alternative sampler [55, 45, 77]. The responses are then labeled as (y_+^t, y_-^t) based on a binary preference $b^t \sim \mathbb{P}(y_1^t \succ y_2^t | x^t)$, and added to the preference dataset via $\mathcal{D}^{t+1} \leftarrow \mathcal{D}^t \cup \{(x^t, y_+^t, y_-^t)\}$, which can then be used to compute an updated policy π^{t+1} . The preference distribution $\mathbb{P}(y_1 \succ y_2 | x)$ represents the underlying verifier or oracle of interest; it is typically assumed that preferences follow the Bradley-Terry model [7], i.e.

$$\mathbb{P}(y_1 \succ y_2 | x) = \frac{\exp(r^*(x, y_1))}{\exp(r^*(x, y_1)) + \exp(r^*(x, y_2))}, \quad (14)$$

for an underlying *reward function* $r^* : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$. As with our setting, the goal is to use the collected data $\mathcal{D}_{\text{pref}}$ to produce a final policy $\hat{\pi}$ with high KL-regularized reward $J_{\beta}(\hat{\pi})$.

When $N = 2$, our absolute reward formulation in Section 1.1 is very closely related to this formulation, and algorithms for one setting can easily be adapted to the other (typically the only change is in the objective used to estimate the reward model). We use the absolute reward formulation and general N (as described Section 1.1) because (i) allowing for $N > 2$ makes our lower bounds/impossibility results stronger, even though our algorithms themselves only use $N = 2$; and (ii) the absolute reward formulation—which has been used in prior work empirically [88, 89, 85, 96] and in theory [107, 84, 96]—is a more realistic model for the motivating problem of learning from a strong oracle/verifier such as a proof checker.

Adapting preference-based algorithms to reward-based feedback. OnlineDPO [32] proceeds iteratively for $t \in [T_{\text{prompt}}]$ as follows:

1. Compute $\pi^t := \pi_{\theta^t}$ by solving the DPO objective:

$$\theta^t \leftarrow \arg \min_{\theta \in \Theta} \sum_{(x, y_+, y_-) \in \mathcal{D}^t} -\log \left[\sigma \left(\beta \log \frac{\pi_{\theta}(y_+ | x)}{\pi_{\text{ref}}(y_+ | x)} - \beta \log \frac{\pi_{\theta}(y_- | x)}{\pi_{\text{ref}}(y_- | x)} \right) \right], \quad (15)$$

where $\sigma(z) := \frac{\exp(z)}{1 + \exp(z)}$ is the sigmoid function.

2. Sample $y_1^t, y_2^t \sim \pi_{\theta^t}(\cdot | x^t)$, then label as (y_+^t, y_-^t) and update $\mathcal{D}^{t+1} \leftarrow \mathcal{D}^t \cup \{(x^t, y_+^t, y_-^t)\}$.

For our reward-based setting, we change Eq. (15) to

$$\theta^t \leftarrow \arg \min_{\theta \in \Theta} \sum_{(x, y_1, y_2, r_1, r_2) \in \mathcal{D}^t} \left(\beta \log \frac{\pi_{\theta}(y_1 | x)}{\pi_{\text{ref}}(y_1 | x)} - \beta \log \frac{\pi_{\theta}(y_2 | x)}{\pi_{\text{ref}}(y_2 | x)} - (r_1 - r_2) \right)^2. \quad (16)$$

It is possible to show that this algorithm obtains $T_{\text{data}}(\varepsilon, \delta) = \text{poly}(d, R_{\text{max}}, C_{\text{cov}}(\pi_{\beta}^*), \varepsilon^{-1}, \log(\delta^{-1}))$ for our linear softmax setting through standard arguments.

Similarly, at each step t , XPO [94] minimizes the objective

$$\theta^t \leftarrow \arg \min_{\theta \in \Theta} \left\{ \alpha \sum_{i < t} \log \pi_{\theta}(y_2^i | x^i) + \sum_{(x, y_1, y_2, r_1, r_2) \in \mathcal{D}^t} -\log \left[\sigma \left(\beta \log \frac{\pi_{\theta}(y_+ | x)}{\pi_{\text{ref}}(y_+ | x)} - \beta \log \frac{\pi_{\theta}(y_- | x)}{\pi_{\text{ref}}(y_- | x)} \right) \right] \right\},$$

for an optimism parameter $\alpha > 0$, then samples $y^t \sim \pi_{\theta^t}(\cdot | x^t)$ and $y_2^t \sim \pi_{\text{ref}}(\cdot | x^t)$ and updates $\mathcal{D}^{t+1} \leftarrow \mathcal{D}^t \cup \{(x^t, y_+^t, y_-^t)\}$. To adapt XP0 to the reward-based setting, we analogously change the objective to

$$\theta^t \leftarrow \arg \min_{\theta \in \Theta} \left\{ \alpha \sum_{i < t} \log \pi_{\theta}(y_2^i | x^i) + \sum_{(x, y_1, y_2, r_1, r_2) \in \mathcal{D}^t} \left(\beta \log \frac{\pi_{\theta}(y_1 | x)}{\pi_{\text{ref}}(y_1 | x)} - \beta \log \frac{\pi_{\theta}(y_2 | x)}{\pi_{\text{ref}}(y_2 | x)} - (r_1 - r_2) \right)^2 \right\}. \quad (17)$$

The sample complexity bound $T_{\text{data}}(\varepsilon, \delta) \lesssim \text{poly}(R_{\max}) \cdot \frac{d^2 \log(\delta^{-1})}{\varepsilon^2}$ claimed in Eq. (4) follows by (i) specializing the SEC-based bound in Xie et al. [94] to the linear softmax policy class, and (ii) noting that the $\exp(R_{\max})$ factor in the sample complexity guarantee in Xie et al. [94] can be removed under reward-based feedback (as it arises due to converting between the logistic loss for the Bradley-Terry model and the square loss); these calculations can be found in Theorem J.1 and Lemmas J.4 and J.5 of Huang et al. [35].

Adapting SpannerSampling to preference-based feedback. To adapt SpannerSampling to preference-based feedback, the only change required is to switch the reward estimation step in Eq. (49) to the following DPO-like objective: $\theta^t \leftarrow \arg \min_{\theta \in \Theta} \sum_{(x, y_+, y_-) \in \mathcal{D}_{\text{exp}}^t \cup \mathcal{D}_{\text{span}}^t} -\log \sigma(\langle \theta, \varphi(x, y_+, y_-) \rangle)$. This leads to identical guarantees, except that the sample complexity will pay for a $\exp(R_{\max})$ factor due to conversion from logistic loss to square loss (e.g., Lemma C.8 in Xie et al. [94]).

Appendix C. Sampling Oracles: Beyond Linear Policies

We expect that our sampling oracle abstraction for the language model alignment problem will be of use beyond the linear softmax policy parameterization we focus on. In this section, we briefly discuss possibilities for extending Definition 2.1 beyond the linear setting, as well as challenges this entails.

Features versus log-probabilities. Recall that the sampling oracle in Definition 2.1 reveals the features $\phi(x, y)$ for responses y sampled from the oracle, but does not reveal the log-probabilities $\log \pi_{\theta}(y | x)$ themselves. As highlighted in Section 2, the observed features are closely related, as they can be used to evaluate $\beta \log \frac{\pi_{\theta}(y|x)}{\pi_{\text{ref}}(y|x)} - \beta \log \frac{\pi_{\theta}(y'|x)}{\pi_{\text{ref}}(y'|x)} = \langle \theta, \phi(x, y) - \phi(x, y') \rangle$, but they cannot be used to compute $\log \pi_{\theta}(y | x)$ itself in general. We adopt this formalism because it simplifies the coverage-based lower bounds in Section 2.2; our algorithmic results only make use of the features $\phi(x, y)$, and hence fall into the framework of Definition 2.1. But to move beyond linear policies, it is more natural to directly allow the learner to query the log-probabilities.

Definition C.1 (Generalized sampling oracle framework). *In one query, the learner proposes a prompt $x \in \mathcal{X}$ and parameter $\theta \in \Theta$, and receives a conditional sample $y \sim \pi_{\theta}(\cdot | x)$, as well as the corresponding log-probability $\log \pi_{\theta}(y | x)$ for the sampled response (note that $\pi_0 = \pi_{\text{ref}}$).*

There is a technical subtlety here (and in Definition 2.1) as far as the learner’s a-priori knowledge. In the linear softmax setting, if ϕ is known a-priori, then ruling out algorithms that enumerate over the response space \mathcal{Y} requires additionally assuming that each query y_i^t made to the reward oracle is a response that has previously been revealed by the sampling oracle. It seems more natural to consider the features (and, in general, the parametrization $\theta \mapsto \pi_{\theta}$) to be unknown a-priori.

Proving lower bounds like Theorem 2.1 may be more challenging in the framework in Definition C.1, as the log-probabilities can potentially provide more information than the features

themselves. On the other hand, more assumptions are likely required to derive efficient algorithms. For example, it is not clear that one can efficiently minimize the DPO objective under [Definition C.1](#), and so it might be necessary to assume an additional oracle for minimizing the objective. We leave a detailed understanding for future work.

Part II

Proofs from Sections 2 through 4

Appendix D. Technical Tools

For a pair of probability measures \mathbb{P} and \mathbb{Q} , we define the total variation distance as $D_{\text{TV}}(\mathbb{P}, \mathbb{Q}) = \frac{1}{2} \int |\mathrm{d}\mathbb{P} - \mathrm{d}\mathbb{Q}|$, and define Hellinger distance by $D_{\text{H}}^2(\mathbb{P}, \mathbb{Q}) = \int (\sqrt{\mathrm{d}\mathbb{Q}} - \sqrt{\mathrm{d}\mathbb{P}})^2$. We define KL divergence by $D_{\text{KL}}(\mathbb{P} \parallel \mathbb{Q}) = \int \mathrm{d}\mathbb{P} \log(\frac{\mathrm{d}\mathbb{P}}{\mathrm{d}\mathbb{Q}})$ if $\mathbb{P} \ll \mathbb{Q}$ and $D_{\text{KL}}(\mathbb{P} \parallel \mathbb{Q}) = +\infty$ otherwise.

D.1. Tail Bounds

Lemma D.1 (Azuma-Hoeffding). *Let $(X_t)_{t \leq T}$ be a sequence of real-valued random variables adapted to a filtration $(\mathcal{F}_t)_{t \leq T}$. If $|X_t| \leq R$ almost surely, then with probability at least $1 - \delta$, for all $T' \leq T$,*

$$\left| \sum_{t=1}^{T'} X_t - \mathbb{E}_{t-1}[X_t] \right| \leq R \cdot \sqrt{8T' \log(2\delta^{-1})}.$$

Lemma D.2 (Freedman's inequality). *Let $(X_t)_{t \leq T}$ be a real-valued martingale difference sequence adapted to a filtration $(\mathcal{F}_t)_{t \leq T}$. If $|X_t| \leq R$ almost surely, then for any $\eta \in (0, 1/R)$, with probability at least $1 - \delta$, for all $T' \leq T$,*

$$\sum_{t=1}^{T'} X_t \leq \eta \sum_{t=1}^{T'} \mathbb{E}_{t-1}[X_t^2] + \frac{\log(\delta^{-1})}{\eta}.$$

The next result is a standard consequence of [Lemma D.2](#) (e.g., Foster et al. [27]).

Lemma D.3. *Let $(X_t)_{t \leq T}$ be a sequence of random variables adapted to a filtration $(\mathcal{F}_t)_{t \leq T}$. If $0 \leq X_t \leq R$ almost surely, then with probability at least $1 - \delta$, for all $T' \leq T$,*

$$\sum_{t=1}^{T'} X_t \leq \frac{3}{2} \sum_{t=1}^{T'} \mathbb{E}_{t-1}[X_t] + 4R \log(2\delta^{-1}), \quad (18)$$

and

$$\sum_{t=1}^{T'} \mathbb{E}_{t-1}[X_t] \leq 2 \sum_{t=1}^{T'} X_t + 8R \log(2\delta^{-1}). \quad (19)$$

D.2. Elliptic Potential

Lemma D.4 (e.g. Lemma 19.4 in Lattimore and Szepesvári [49]). *Let $v_1, \dots, v_T \in \mathbb{R}^d$ satisfy $\|v_t\|_2 \leq 1$ for all $t \in [T]$. Fix $\lambda > 0$, and let $V_t = \lambda I + \sum_{i < t} v_i v_i^\top$. Then*

$$\sum_{t=1}^T \|v_t\|_{V_t^{-1}}^2 \wedge 1 \leq 2 \sum_{t=1}^T \log(1 + \|v_t\|_{V_t^{-1}}^2) \leq 2d \log(1 + \lambda^{-1}T/d). \quad (20)$$

As a consequence, we have

$$\sum_{t=1}^T \|v_t\|_{V_t^{-1}} \wedge 1 \leq \sqrt{2dT \log(1 + \lambda^{-1}T/d)}. \quad (21)$$

D.3. Miscellaneous Lemmas

Lemma D.5 (Sequential union bound). *Let $T, H \in \mathbb{N}$ and $\delta \in (0, 1)$ be given. Further, let \mathcal{B}_1 be an algorithm that runs in $T \in \mathbb{N}$ iterations. At each iteration, \mathcal{B}_1 makes a sequence of H calls to a subroutine \mathcal{B}_2 . Let \mathfrak{S} denote the state space of algorithm \mathcal{B}_1 ; the space capturing the values of all the internal variables of \mathcal{B}_1 . Let $\mathbf{S}_{h,-}^t \in \mathfrak{S}$ denote the random state of \mathcal{B}_1 immediately before the h th call to \mathcal{B}_2 during the t th iteration; further, let $\mathbf{S}_{h,+}^t \in \mathfrak{S}$ denote the random state of \mathcal{B}_1 immediately after this call to \mathcal{B}_2 . Suppose that for any $\mathbf{S}_{h,-}^t \in \mathfrak{S}$, there is an event $\mathcal{E}_h^t(\mathbf{S}_{h,-}^t) \subset \mathfrak{S}$ such that $\mathbb{P}[\mathbf{S}_{h,+}^t \in \mathcal{E}_h^t(\mathbf{S}_{h,-}^t)] \geq 1 - \delta$. Then, with probability at least $1 - \delta HT$, for all $t \in [T]$ and $h \in [H]$, we have $\mathbf{S}_{h,+}^t \in \mathcal{E}_h^t(\mathbf{S}_{h,-}^t)$.*

Proof. Let \mathcal{E} be the event defined by

$$\mathcal{E} := \left\{ \prod_{t=1}^T \prod_{h=1}^H \mathbb{I}\{\mathbf{S}_{h,+}^t \in \mathcal{E}_h^t(\mathbf{S}_{h,-}^t)\} = 1 \right\}. \quad (22)$$

We need to show that $\mathbb{P}[\mathcal{E}] \geq 1 - \delta HT$. To this end, we note that by the chain rule of probability, we have

$$\begin{aligned} \mathbb{P}[\mathcal{E}] &= \prod_{t=1}^T \prod_{h=1}^H \mathbb{E} [\mathbb{P}[\mathbf{S}_{h,+}^t \in \mathcal{E}_h^t(\mathbf{S}_{h,-}^t) \mid \mathbf{S}_{h,-}^t]] , \\ &\geq \prod_{t=1}^T \prod_{h=1}^H (1 - \delta) , \\ &\geq 1 - TH\delta, \end{aligned} \quad (23)$$

where (23) follows by the fact that $\mathbb{P}[\mathbf{S}_{h,+}^t \in \mathcal{E}_h^t(\mathbf{S}_{h,-}^t)] \geq 1 - \delta$ for all $\mathbf{S}_{h,-}^t \in \mathfrak{S}$, and the last inequality follows by the fact that for any sequence $x_1, \dots, x_T \in (0, 1)$, $\prod_{i \in [T]} (1 - x_i) \geq 1 - \sum_{i \in [T]} x_i$. \square

Lemma D.6. *If $x \geq 1$ satisfies $x \leq a \log(1 + bx)$ for $a, b \geq 3$, then $x \leq 2a \log(1 + ab)$.*

Proof of Lemma D.6. First, note that by reparameterizing $x \leftarrow bx$ and $c \leftarrow ab$, it suffices to show that $x \leq c \log(1 + x)$ for $x \geq 1, c \geq 3$ implies $x \leq 2c \log(1 + c)$. Toward proving the latter statement, we first note that if $x \geq c$, then $x \mapsto x$ increases faster than $x \mapsto c \log(1 + x)$, so any point $x \geq c$ for which $x > 2c \log(1 + x)$ gives a valid upper bound. Let us choose $x = 2c \log(1 + c)$. Then we have $c \log(1 + x) \leq c \log(1 + 2c \log(1 + c)) < c \log(1 + c^2) \leq c \log((1 + c)^2) \leq 2c \log(1 + c) = x$ as desired, where the strict inequality uses that $2c \log(1 + c) \leq c^2$ for $c \geq 3$. \square

Appendix E. Proofs from Section 2

In this section we restate and prove [Theorem 2.1](#).

Theorem 2.1 (Necessity of coverage). *Let $C^*, Y \geq 2$ be given. Let Alg be an online alignment algorithm that uses $T_{\text{data}}(\varepsilon, \delta)$ reward oracle queries and $T_{\text{comp}}(\varepsilon, \delta)$ strong sampling oracle queries whenever (i) the parameter space is $\Theta = \mathbb{B}_2(1)$, (ii) [Assumption 2.1](#) is satisfied with $R_{\max} = B = 1$, (iii) $C_{\text{cov}}(\pi_\beta^*) \leq C^*$, and (iv) the response space has size $Y = |\mathcal{Y}|$. Then, either $T_{\text{data}}(\varepsilon, \delta) \geq Y/8$, or*

$$T_{\text{comp}}(\varepsilon, \delta) \geq \Omega\left(\min\{e^{\beta^2 d/2}, e^{\beta^{-1}/2}, C^*\}\right). \quad (7)$$

Proof of Theorem 2.1. If $\beta > 1/2$ then the lower bound on $T_{\text{comp}}(\varepsilon, \delta)$ is vacuously true, so we may assume henceforth that $\beta \leq 1/2$. Similarly, we may assume without loss of generality that $Y \geq 9$. Let \mathcal{S} be an arbitrary set of size $Y - 1$. We take prompt space $\mathcal{X} = \{\perp\}$ (and henceforth omit all dependencies on the prompt \perp). We take response space $\mathcal{Y} = \{0\} \cup \mathcal{S}$. We take parameter space $\Theta = \mathbb{B}_2(1)$.

For each $\theta^* \in \mathbb{B}_2(1)$ and $y^* \in \mathcal{S}$, we define an instance $\mathcal{I}^{\theta^*, y^*}$ of the online alignment problem (with linear softmax policy class) as follows:

- The reference policy is $\pi_{\text{ref}}^{y^*} \in \Delta(\mathcal{Y})$ defined by $\pi_{\text{ref}}^{y^*}(0) = 1 - \varepsilon_{\text{ref}}$ and $\pi_{\text{ref}}^{y^*}(y^*) = \varepsilon_{\text{ref}}$ where $\varepsilon_{\text{ref}} := \max\{1/C^*, e^{-\beta^{-1}/2}\}$.
- The feature mapping $\phi^{\theta^*, y^*} : \mathcal{Y} \rightarrow \mathbb{R}^d$ is defined by

$$\phi^{\theta^*, y^*}(y) = \begin{cases} \theta^* & \text{if } y = y^* \\ 0 & \text{if } y \neq y^* \end{cases}.$$

- The reward function $r^{\theta^*, y^*} : \mathcal{Y} \rightarrow [0, 1]$ is defined by $r^{\theta^*, y^*}(y) = \langle \theta^*, \phi^{\theta^*, y^*}(y) \rangle = \mathbb{1}[y = y^*]$.

Note that \mathcal{X} , \mathcal{Y} , and Θ are fixed, and do not depend on the choice of (θ^*, y^*) .

We make the following observations about $\mathcal{I}^{\theta^*, y^*}$. Since $r^{\theta^*, y^*}(y) = \langle \theta^*, \phi^{\theta^*, y^*}(y) \rangle$ and $\theta^* \in \mathbb{B}_2(1) = \Theta$, [Assumption 1.1](#) is satisfied. In particular, the optimal KL-regularized policy is $\pi_{\theta^*}(y) \propto \pi_{\text{ref}}(y) \exp(\beta^{-1} \langle \theta^*, \phi^{\theta^*, y^*}(y) \rangle)$. It is straightforward to check that [Assumption 2.1](#) is satisfied with $R_{\max} = B = 1$. From [Eq. \(3\)](#), we have

$$C_{\text{cov}}(\pi_{\theta^*}) = C_{\text{cov}}(\pi_{\theta^*}) \leq \max_{y \in \{0, y^*\}} \frac{1}{\pi_{\text{ref}}(y)} \leq \frac{1}{\varepsilon_{\text{ref}}} \leq C^*$$

where the second inequality uses the fact that $1 - \varepsilon_{\text{ref}} \geq \varepsilon_{\text{ref}}$ (which holds since $C^* \geq 2$ and $\beta \leq 1/2$). Finally, $|\mathcal{Y}| = Y$ by construction. From the theorem assumptions, we conclude that for all instances $\mathcal{I}^{\theta^*, y^*}$, Alg uses $T_{\text{data}}(\varepsilon, \delta)$ queries to the reward oracle and $T_{\text{comp}}(\varepsilon, \delta)$ queries to the strong sampling oracle, and with probability at least $1 - \delta$ returns a policy $\hat{\pi}$ satisfying $J_\beta(\pi_{\theta^*}) - J_\beta(\hat{\pi}) \leq \varepsilon$. Assume, for the sake of contradiction, that $T_{\text{data}} := T_{\text{data}}(1/4, 1/4) < Y/8$ and $T_{\text{comp}} := T_{\text{comp}}(1/4, 1/4) < c_0 \cdot \min\{e^{\beta^2 d/2}, e^{\beta^{-1}/2}, C^*\}$ for a universal constant $c_0 > 0$ to be determined.

Now, consider the distribution over problem instances induced by sampling θ^* uniformly from the unit sphere in \mathbb{R}^d , and independently sampling $y^* \sim \text{Unif}(\mathcal{S})$. Then execute Alg on instance

$\mathcal{I}^{\theta^*, y^*}$ with error tolerance $\varepsilon = 1/4$ and failure probability $\delta = 1/4$. For the purposes of analysis, for each $q \geq 0$, let $\overline{\text{Alg}}^{[q]}$ denote a modified version of algorithm where the first q oracle queries are answered with $0 \in \mathcal{Y}$ (for the sampling oracle) or $0 \in \mathbb{R}$ (for the reward oracle), and the algorithm is run unmodified for subsequent steps. Let $\hat{\pi}^{[q]}$ denote the output of $\overline{\text{Alg}}^{[q]}$. On the one hand, since $\overline{\text{Alg}}^{[0]} = \text{Alg}$, we know that

$$\mathbb{P}[J_\beta(\pi_{\theta^*}) - J_\beta(\hat{\pi}^{[0]}) \leq \varepsilon] \geq 1 - \delta \quad (24)$$

where the probability is over the random choice of (θ^*, y^*) and the randomness of $\overline{\text{Alg}}^{[0]}$ (and its oracle calls). On the other hand, since the problem parameters $(\mathcal{X}, \mathcal{Y}, \Theta)$ are independent of (θ^*, y^*) and all queries made by $\overline{\text{Alg}}^{[T_{\text{comp}} + T_{\text{data}}]}$ are answered independently of (θ^*, y^*) , we have that $\hat{\pi}^{[T_{\text{comp}} + T_{\text{data}}]}$ is independent of (θ^*, y^*) . We use this to prove the following lower bound on the regret for $\hat{\pi}^{[T_{\text{comp}} + T_{\text{data}}]}$.

Lemma E.1. *For $\varepsilon = \delta = 1/4$, it holds that*

$$\mathbb{P}[J_\beta(\pi_{\theta^*}) - J_\beta(\hat{\pi}^{[T_{\text{comp}} + T_{\text{data}}]}) \leq \varepsilon] \leq 1/2. \quad (25)$$

Proof of Lemma E.1. For any fixed (θ^*, y^*) , in instance $\mathcal{I}^{\theta^*, y^*}$, we have

$$\begin{aligned} J_\beta(\pi_{\theta^*}) &= \pi_{\theta^*}(y^*) - \beta D_{\text{KL}}(\pi_{\theta^*} \parallel \pi_{\text{ref}}) \\ &= \beta \log(1 - \varepsilon_{\text{ref}} + \varepsilon_{\text{ref}} e^{\beta^{-1}}) \\ &\geq \beta \log(\varepsilon_{\text{ref}} e^{\beta^{-1}}) \\ &= 1 - \beta \log(1/\varepsilon_{\text{ref}}) \\ &\geq 1/2, \end{aligned}$$

where the first equality is by definition of J_β and the reward function r^{θ^*, y^*} ; the second equality is by explicit calculation; and the final inequality is $\varepsilon_{\text{ref}} \geq \frac{1}{2} e^{-\beta^{-1}/2}$. But we also have

$$J_\beta(\hat{\pi}^{[T_{\text{comp}} + T_{\text{data}}]}) \leq \hat{\pi}^{[T_{\text{comp}} + T_{\text{data}}]}(y^*)$$

by definition of J_β and non-negativity of KL-divergence. Since $\hat{\pi}^{[T_{\text{comp}} + T_{\text{data}}]}$ is independent of y^* and y^* is uniformly distributed in \mathcal{S} , we know that

$$\mathbb{E}[\hat{\pi}^{[T_{\text{comp}} + T_{\text{data}}]}(y^*)] \leq 1/|\mathcal{S}|,$$

and so by Markov's inequality and the fact that $|\mathcal{S}| = Y - 1 \geq 8$, we have

$$\mathbb{P}[\hat{\pi}^{[T_{\text{comp}} + T_{\text{data}}]}(y^*) \geq 1/4] \leq 1/2.$$

Recalling that $\varepsilon = 1/4$, it follows that

$$\mathbb{P}[J_\beta(\pi_a) - J_\beta(\hat{\pi}^{[T_{\text{comp}} + T_{\text{data}}]}) \leq \varepsilon] \leq \mathbb{P}[J_\beta(\pi_a) - \hat{\pi}^{[T_{\text{comp}} + T_{\text{data}}]}(y^*) \leq \varepsilon] \leq 1/2.$$

□

From here, we proceed by relating the regret of $\hat{\pi}^{[0]}$ to that of $\hat{\pi}^{[T_{\text{comp}} + T_{\text{data}}]}$. Fix $0 \leq q < T_{\text{comp}} + T_{\text{data}}$. The probability that $\overline{\text{Alg}}^{[q]}$ deviates from $\overline{\text{Alg}}^{[q+1]}$ is at most the probability that the response to the $(q+1)$ -th oracle query by $\overline{\text{Alg}}^{[q]}$ is non-zero. Since all previous oracle queries by $\overline{\text{Alg}}^{[q]}$ were answered independently of (θ^*, y^*) , the $(q+1)$ -th query (though not its answer) is independent as well. Condition on this query; we distinguish two cases.

1. If it is a sampling oracle query $\theta \in \Theta$, then the probability that the execution of $\overline{\text{Alg}}^{[q+1]}$ deviates from $\overline{\text{Alg}}^{[q]}$ (in the optimal coupling of their executions) is precisely the probability that the sampling oracle $y \sim \pi_\theta$ yields a non-zero answer $y \neq y^*$, which is precisely $\pi_\theta(y^*)$. Moreover, we can bound the expectation (over all randomness) of this probability:

$$\begin{aligned} \mathbb{E}[\pi_\theta(y^*)] &= \mathbb{E} \left[\frac{\varepsilon_{\text{ref}} \exp(\beta^{-1} \langle \theta, \phi^{\theta^*, y^*}(y^*) \rangle)}{(1 - \varepsilon_{\text{ref}}) \exp(\beta^{-1} \langle \theta, \phi^{\theta^*, y^*}(0) \rangle) + \varepsilon_{\text{ref}} \exp(\beta^{-1} \langle \theta, \phi^{\theta^*, y^*}(y^*) \rangle)} \right] \\ &= \mathbb{E} \left[\frac{\varepsilon_{\text{ref}} \exp(\beta^{-1} \langle \theta, \theta^* \rangle)}{1 - \varepsilon_{\text{ref}} + \varepsilon_{\text{ref}} \exp(\beta^{-1} \langle \theta, \theta^* \rangle)} \right] \\ &\leq \mathbb{E} \left[\frac{\varepsilon_{\text{ref}} \exp(\beta^{-1} \max(\theta_1^*, 0))}{1 - \varepsilon_{\text{ref}} + \varepsilon_{\text{ref}} \exp(\beta^{-1} \max(\theta_1^*, 0))} \right] \leq O(\varepsilon_{\text{ref}} + \exp(-\beta^2 d/2)), \end{aligned}$$

where the first inequality uses the fact that θ is independent of θ^* and hence $\langle \theta, \theta^* \rangle$ is stochastically dominated by $\max(\theta_1^*, 0)$, and the final inequality uses [Lemma E.2](#) (stated and proven in the sequel).

2. If it is a reward query $y \in \mathcal{Y}$, then the probability that the execution of $\overline{\text{Alg}}^{[q+1]}$ deviates from $\overline{\text{Alg}}^{[q]}$ is precisely the probability that the reward oracle yields a non-zero answer, which is $r^{\theta^*, y^*}(y) = \mathbb{1}[y = y^*]$. Since y is independent of y^* , we have $\mathbb{P}[r^{\theta^*, y^*}(y) \neq 0] \leq 1/|\mathcal{S}|$.

Therefore by the data processing inequality,

$$\begin{aligned} \mathbb{P}[\hat{\pi}^{[q]} \neq \hat{\pi}^{[q+1]}] &\leq D_{\text{TV}}(\text{Law}(\overline{\text{Alg}}^{[q]}), \text{Law}(\overline{\text{Alg}}^{[q+1]})) \\ &\leq O(\varepsilon_{\text{ref}} + \exp(-\beta^2 d/2)) \cdot \mathbb{P}[(q+1)\text{-th query by Alg}^{[q]} \text{ is sampling}] \\ &\quad + \frac{1}{|\mathcal{S}|} \cdot \mathbb{P}[(q+1)\text{-th query by Alg}^{[q]} \text{ is reward}] \\ &= O(\varepsilon_{\text{ref}} + \exp(-\beta^2 d/2)) \cdot \mathbb{P}[(q+1)\text{-th query by Alg}^{[T_{\text{comp}}+T_{\text{data}}]} \text{ is sampling}] \\ &\quad + \frac{1}{|\mathcal{S}|} \cdot \mathbb{P}[(q+1)\text{-th query by Alg}^{[T_{\text{comp}}+T_{\text{data}}]} \text{ is reward}] \end{aligned} \tag{26}$$

where the equality uses the fact that the executions of $\text{Alg}^{[q]}$ and $\text{Alg}^{[T_{\text{comp}}+T_{\text{data}}]}$ are identically distributed up to and including the $(q+1)$ -th query. We conclude that

$$\begin{aligned} \mathbb{P}[J_\beta(\pi_{\theta^*}) - J_\beta(\hat{\pi}^{[0]}) \leq \varepsilon_{\text{ref}}] &\leq \mathbb{P}[J_\beta(\pi_{\theta^*}) - J_\beta(\hat{\pi}^{[T_{\text{comp}}+T_{\text{data}}]}) \leq \varepsilon_{\text{ref}}] + \sum_{q=0}^{T_{\text{comp}}+T_{\text{data}}-1} \mathbb{P}[\hat{\pi}^{[q]} \neq \hat{\pi}^{[q+1]}] \\ &\leq \frac{1}{2} + O(\varepsilon_{\text{ref}} + \exp(-\beta^2 d/2))T_{\text{comp}} + \frac{1}{|\mathcal{S}|}T_{\text{data}} < 3/4, \end{aligned}$$

where the second inequality is by [Eqs. \(25\) and \(26\)](#), and the third inequality is by the assumed bounds on $T_{\text{comp}}, T_{\text{data}}$ and holds so long as $c_0 > 0$ is a sufficiently small constant. This contradicts [Eq. \(24\)](#), so it must be that either $T_{\text{data}} \geq Y/8$ or $T_{\text{comp}} \geq c_0 \cdot \min\{e^{\beta^2 d/2}, e^{-\beta^{-1}/2}, C^*\}$. \square

Lemma E.2. Fix $\epsilon \in (0, 1/2)$, $\beta > 0$, and $d \in \mathbb{N}$. Let $X \sim \text{Unif}(\mathbb{S}^{d-1})$ where \mathbb{S}^{d-1} is the unit sphere in \mathbb{R}^d . Then

$$\mathbb{E} \left[\frac{\epsilon \cdot e^{\beta^{-1} \max(X_1, 0)}}{1 - \epsilon + \epsilon \cdot e^{\beta^{-1} \max(X_1, 0)}} \right] \lesssim \epsilon + e^{-\beta^2 d/2}.$$

Proof of Lemma E.2. The quantity inside the expectation is always at most 1. Moreover, if $X_1 \leq \beta$, then $\frac{\epsilon \cdot e^{\beta^{-1} \max(X_1, 0)}}{1 - \epsilon + \epsilon \cdot e^{\beta^{-1} \max(X_1, 0)}} \lesssim \epsilon$. It follows that

$$\mathbb{E} \left[\frac{\epsilon \cdot e^{\beta^{-1} \max(X_1, 0)}}{1 - \epsilon + \epsilon \cdot e^{\beta^{-1} \max(X_1, 0)}} \right] \lesssim \epsilon + \mathbb{P}[X_1 > \beta] \leq \epsilon + e^{-\beta^2 d/2}$$

by a standard bound on the volume of a spherical cap [80]. □

Appendix F. SoftmaxSampler Algorithm and Guarantees

Algorithm 1 SoftmaxSampler $_{\beta, M, \delta}(f; x, \pi_{\text{ref}})$

input: Function f , prompt x , base policy π_{ref} , parameter $\beta > 0$, rejection threshold $M > 0$, failure probability $\delta \in (0, 1)$.

- 1: Let $N := 4M \log(4\delta^{-1})$.
 /* Estimate normalization constant */
- 2: Sample $y_1, \dots, y_N \sim \pi_{\text{ref}}(\cdot | x)$ i.i.d.
- 3: Set $\hat{Z} := \frac{1}{N} \sum_{i=1}^N \exp(\beta^{-1} f(x, y_i))$.
 /* Rejection sampling */
- 4: **for** iteration $i = 1, 2, \dots, N$ **do**
- 5: Sample $y \sim \pi_{\text{ref}}(\cdot | x)$ and $\xi \sim \text{Ber}(\exp(\beta^{-1} f(x, y)) / \hat{Z})$.
- 6: **If** $\xi = 1$, **return** y .
- 7: **return** $y \sim \pi_{\text{ref}}(\cdot | x)$. // Failure event; occurs with low probability.

In this section, we give self-contained guarantees for the SoftmaxSampler algorithm (Algorithm 1) used within SpannerSampling, as well as a slightly more general version of the algorithm, SoftmaxSamplerDensity (Algorithm 2), which is used within MTSS (Appendix J). Both algorithms take as input a function $f(x, y)$ and use rejection sampling to generate samples from the softmax policy

$$\pi_f(y | x) \propto \pi_{\text{ref}}(y | x) \exp(\beta^{-1} f(x, y)) \quad (27)$$

given sample access to π_{ref} . SoftmaxSamplerDensity only differs from SoftmaxSampler in that, in addition to using rejection sampling to generate samples from (27), it also returns an estimate for density ratio $\frac{\pi_f(y|x)}{\pi_{\text{ref}}(y|x)}$ for the sampled response; since the SoftmaxSampler algorithm already estimates the normalization constant for the target policy, which is the only non-trivial part of the density ratio to compute, this requires no change outside of explicitly returning the density ratio estimate.

Algorithm overview. Let us briefly describe the algorithm. Line 4 of SoftmaxSampler and SoftmaxSamplerDensity applies vanilla rejection sampling to generate samples from π_f , sampling multiple responses from π_{ref} and using the density ratio to decide whether to accept each response. The only subtlety is that the density ratio

$$\frac{\pi_f(y | x)}{\pi_{\text{ref}}(y | x)} = \frac{\exp(\beta^{-1} f(x, y))}{\mathbb{E}_{y' \sim \pi_{\text{ref}}}[\exp(\beta^{-1} f(x, y'))]},$$

depends on the normalization constant $Z(x) := \mathbb{E}_{y' \sim \pi_{\text{ref}}}[\exp(\beta^{-1} f(x, y'))]$, which is unknown. To address this, Line 3 estimates the normalization constant via sampling from π_{ref} and computing the empirical mean. The estimated normalization constant is then used to set the rejection threshold.

The main guarantee for SoftmaxSamplerDensity is as follows.

Theorem F.1 (Guarantee for SoftmaxSamplerDensity). *Let $f : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$, $x \in \mathcal{X}$, and $\beta > 0$ be given, and define*

$$\pi_f(\cdot | x) \propto \pi_{\text{ref}}(\cdot | x) \exp(\beta^{-1} f(x, \cdot)), \quad \text{and} \quad C_\infty := 1 \vee \left\| \frac{\pi_f(\cdot | x)}{\pi_{\text{ref}}(\cdot | x)} \right\|_\infty. \quad (28)$$

Algorithm 2 SoftmaxSamplerDensity

input: Function $f : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$, prompt x , base policy $\pi_{\text{ref}} : \mathcal{X} \rightarrow \Delta(\mathcal{Y})$, parameter $\beta > 0$, rejection threshold $M > 0$, failure probability $\delta \in (0, 1)$.

- 1: Let $N := 4M \log(4\delta^{-1})$.
- /* Estimate normalization constant */*
- 2: Sample $y_1, \dots, y_N \sim \pi_{\text{ref}}(\cdot | x)$ i.i.d.
- 3: Set $\hat{Z} := \frac{1}{N} \sum_{i=1}^N \exp(\beta^{-1} f(x, y_i))$.
- /* Rejection sampling */*
- 4: **for** iteration $i = 1, 2, \dots, N$ **do**
- 5: Sample $y \sim \pi_{\text{ref}}(\cdot | x)$ and $\xi \sim \text{Ber}\left(\frac{\exp(\beta^{-1} f(x, y))}{\hat{Z}M}\right)$.
- 6: Set $\rho \leftarrow \frac{\exp(\beta^{-1} f(x, y))}{\hat{Z}}$. *// $\rho \approx \frac{\exp(\beta^{-1} f(x, y))}{\mathbb{E}_{y \sim \pi_{\text{ref}}(\cdot | x)}[\exp(\beta^{-1} f(x, y))]}$*
- 7: If $\xi = 1$, **return** (y, ρ) .
- 8: Sample $y \sim \pi_{\text{ref}}(\cdot | x)$ and set $\rho \leftarrow \frac{\exp(\beta^{-1} f(x, y))}{\hat{Z}}$.
- 9: **return** (y, ρ) . *// Failure event; occurs with low probability.*

Fix $\delta \in (0, 1)$, and suppose that $M \geq 4C_\infty$. There is an event $\mathcal{E}_{\text{accept}}$ with $\mathbb{P}(\mathcal{E}_{\text{accept}}) \geq 1 - \delta$ under which the output (y, ρ) of $\text{SoftmaxSamplerDensity}_{\beta, M, \delta}(f; x, \pi_{\text{ref}})$ satisfies

$$\mathbb{P}(y = \cdot | \mathcal{E}_{\text{accept}}) = \pi_f(\cdot | x) \quad (29)$$

and

$$\mathbb{I}\{M \geq 4C_\infty^2\} \cdot \left| \log \rho - \log \frac{\pi_f(y | x)}{\pi_{\text{ref}}(y | x)} \right| \leq C_\infty \cdot \sqrt{\frac{2}{M}}. \quad (30)$$

Furthermore, if $|f(\cdot, \cdot)| \leq R_{\max}$, then $\rho \in [e^{-2R_{\max}/\beta}, e^{2R_{\max}/\beta}]$ with probability 1. The total number of sampling queries $y \sim \pi_{\text{ref}}(\cdot | x)$ used by the algorithm is at most $T_{\text{comp}} = 8M \log(4\delta^{-1}) + 1$.

We now state the guarantee for `SoftmaxSampler`, which follows immediately from [Theorem F.1](#).

Theorem F.2 (Guarantee for `SoftmaxSampler`). *Let $f : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$, $x \in \mathcal{X}$, and $\beta > 0$ be given, and define*

$$\pi_f(y | x) \propto \pi_{\text{ref}}(y | x) \exp(\beta^{-1} f(x, y)), \quad \text{and} \quad C_\infty := \left\| \frac{\pi_f(\cdot | x)}{\pi_{\text{ref}}(\cdot | x)} \right\|_\infty. \quad (31)$$

Fix $\delta \in (0, 1)$, and suppose that $M \geq 4C_\infty$. There is an event $\mathcal{E}_{\text{accept}}$ with $\mathbb{P}(\mathcal{E}_{\text{accept}}) \geq 1 - \delta$ such that the response $y \sim \text{SoftmaxSampler}_{\beta, M, \delta}(f; x, \pi_{\text{ref}})$ satisfies

$$\mathbb{P}(y = \cdot | \mathcal{E}_{\text{accept}}) = \pi_f(\cdot | x). \quad (32)$$

The total number of sampling queries $y \sim \pi_{\text{ref}}(\cdot | x)$ used by the algorithm is at most $T_{\text{comp}} = 8M \log(4\delta^{-1}) + 1$.

Further guarantees. We now state some additional results, both of which are fairly straightforward consequences of [Theorems F.1](#) and [F.2](#).

Lemma F.1. *Let $\hat{\pi}_f(\cdot | x)$ denote the distribution over $y \sim \text{SoftmaxSampler}_{\beta, M, \delta}(f; x, \pi_{\text{ref}})$. Suppose that $|f(x, y)| \leq R_{\max}$. Then under the conditions of [Theorem F.2](#), it holds that*

$$D_{\text{TV}}(\hat{\pi}_f(x), \pi_f(x)) \leq \delta, \quad D_{\text{H}}^2(\hat{\pi}_f(x), \pi_f(x)) \leq 2\delta, \quad \text{and} \quad D_{\text{KL}}(\hat{\pi}_f(x) \| \pi_f(x)) \leq 4 \left(\frac{R_{\max}}{\beta} + \log N \right) \delta.$$

In addition,

$$D_{\text{KL}}(\hat{\pi}_f(x) \| \pi_{\text{ref}}(x)) - D_{\text{KL}}(\pi_f(x) \| \pi_{\text{ref}}(x)) \leq 6 \left(\frac{R_{\max}}{\beta} + \log N \right) \delta$$

and

$$\frac{\hat{\pi}_f(y | x)}{\pi_f(y | x)} \leq \exp(R_{\max}/\beta) \cdot N.$$

Lemma F.2. *Let $f : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$, $x \in \mathcal{X}$, and $\beta > 0$ be given, and define*

$$\pi_f(\cdot | x) \propto \pi_{\text{ref}}(\cdot | x) \exp(\beta^{-1} f(x, \cdot)), \quad \text{and} \quad C_{\infty} := \left\| \frac{\pi_f(\cdot | x)}{\pi_{\text{ref}}(\cdot | x)} \right\|_{\infty}. \quad (33)$$

Fix $\delta \in (0, 1)$, and suppose that $M \geq 4C_{\infty}^2$. Consider a call to $\text{SoftmaxSamplerDensity}_{\beta, M, \delta}(f; x, \pi_{\text{ref}})$. Let (y, ρ) denote its random output and let $\hat{\pi}_f(\cdot | x)$ denote the probability distribution of y . Then, we have

$$\left| \mathbb{E}[\log \rho] - \mathbb{E}_{y' \sim \hat{\pi}_f(\cdot | x)} \left[\log \frac{\hat{\pi}_f(y' | x)}{\pi_{\text{ref}}(y' | x)} \right] \right| \leq C_{\infty} \cdot \sqrt{\frac{2}{M}} + 4 \left(\frac{2B}{\beta} + \log(4M \log(4\delta^{-1})) \right) \delta. \quad (34)$$

Finally, we have the following change-of-measure guarantee.

Lemma F.3. *For any function $g(x, y) \in [0, 1]$, let $\hat{\pi}(x) := \text{SoftmaxSamplerDensity}_{\beta, M, \delta}(f; x, \pi_{\text{ref}})$ denote the distribution over responses induced by [Algorithm 2](#). Then for any $\rho \in \Delta(\mathcal{X})$,*

$$\begin{aligned} & \left| \mathbb{E}_{x \sim \rho, y \sim \pi_f(x)}[g(x, y)] - \mathbb{E}_{x \sim \rho, y \sim \hat{\pi}(x)}[g(x, y)] \right| \\ & \leq \delta + \mathbb{P}_{x \sim \rho}[M < 4C_{\text{cond}}(\pi_f | x)], \end{aligned}$$

where $C_{\text{cond}}(\pi_f | x) := \sup_{y \in \mathcal{Y}} \frac{\pi_f(y | x)}{\pi_{\text{ref}}(y | x)}$.

F.1. Proofs

Proof of [Theorem F.1](#). Our starting point is the following standard guarantee for rejection sampling.

Lemma F.4 (Rejection sampling [[5](#)]). *Let $\mu \in \Delta(\mathcal{Y})$ be a proposal distribution, and let ν denote a target distribution that we wish to sample from. Suppose that $\left\| \frac{d\nu}{d\mu} \right\|_{\infty} \leq M$. Consider the algorithm which, for $i = 1, 2, \dots, N$, samples $X_i \sim \mu$, then samples $\xi_i \in \{0, 1\}$ such that $\mathbb{P}(\xi_i = 1 | X_i) = \frac{1}{M} \frac{d\nu}{d\mu}$, and returns X_i if $\xi_i = 1$; we return \perp if $\xi_i = 0$ for all i . If $N \geq M \log(\delta^{-1})$, then $\xi_i = 1$ for some i with probability at least $1 - \delta$, and we have*

$$\mathbb{P}(X_i \in A | \xi_i = 1) = \nu(A).$$

In what follows, we omit dependence on x . Let $Z := \mathbb{E}_{y \sim \pi_{\text{ref}}} [\exp(\beta^{-1} f(y))]$ denote the “true” normalization constant for π_f , and observe that we have

$$C_\infty = \frac{\max_{y \in \mathcal{Y}} \exp(\beta^{-1} f(y))}{Z}. \quad (35)$$

We begin by giving a guarantee for the estimated normalization constant \hat{Z} . We observe that by [Lemma D.3](#), there is an event \mathcal{E} of probability at least $1 - \delta/2$, under which

$$\hat{Z} \leq \frac{3}{2}Z + \frac{4 \max_{y \in \mathcal{Y}} \exp(\beta^{-1} f(y)) \log(4\delta^{-1})}{N},$$

and

$$Z \leq 2\hat{Z} + \frac{8 \max_{y \in \mathcal{Y}} \exp(\beta^{-1} f(y)) \log(4\delta^{-1})}{N}.$$

Using [Eq. \(35\)](#), we can equivalently write this as

$$\hat{Z} \leq \frac{3}{2}Z + \frac{4C_\infty \log(4\delta^{-1})}{N} \cdot Z, \quad \text{and} \quad Z \leq 2\hat{Z} + \frac{8C_\infty \log(4\delta^{-1})}{N} \cdot Z.$$

It follows that as long as $N \geq 16C_\infty \log(4\delta^{-1})$ (or $N \geq 4M \log(4\delta^{-1})$ if $M \geq 4C_\infty$), we have that

$$\frac{1}{4}Z \leq \hat{Z} \leq 2Z. \quad (36)$$

Let us condition on \mathcal{E} until otherwise stated. To proceed, we observe that the for loop in [Line 4](#) can be interpreted as applying the rejection sampling algorithm in [Lemma F.4](#) with $\mu = \pi_{\text{ref}}(\cdot)$, $\nu = \pi_f(\cdot)$, and threshold

$$M' := M \cdot \frac{\hat{Z}}{Z}. \quad (37)$$

Hence, as long as $M' \geq \left\| \frac{\pi_f(\cdot)}{\pi_{\text{ref}}(\cdot)} \right\|_\infty = C_\infty$ and $N \geq M' \log(2\delta^{-1})$, with probability at least $1 - \delta/2$, there will be some i such that $\xi_i = 1$ and $\mathbb{P}(y = \cdot \mid \xi_i = 1) = \pi_f(\cdot)$. Note that under [Eq. \(36\)](#), we have

$$M' = M \cdot \frac{\hat{Z}}{Z} \in \left[\frac{M}{4}, 2M \right].$$

so setting $M \geq 4C_\infty$ and $N \geq 2M \log(2\delta^{-1})$ suffices to prove that

$$\mathbb{P}(y = \cdot \mid \mathcal{E}_{\text{accept}}) = \pi_f(\cdot \mid x), \quad (38)$$

under \mathcal{E} . We now prove the second claim on the approximation of the density ratio.

Estimating the density ratio. We no longer condition on \mathcal{E} . First, note that

$$\frac{\pi_f(\cdot)}{\pi_{\text{ref}}(\cdot)} = \frac{e^{\beta^{-1} f(\cdot)}}{Z}. \quad (39)$$

On the other hand, the output ρ of `SoftmaxSamplerDensity` satisfies $\rho = \frac{e^{\beta^{-1}f(y)}}{\hat{Z}}$. Thus, to get our desired bound on $\left| \log \rho - \log \frac{\pi_f(y)}{\pi_{\text{ref}}(y)} \right|$, it suffices to show that

$$\left| \log \frac{\hat{Z}}{Z} \right| \leq 4C_\infty \cdot \sqrt{\frac{2 \log(4\delta^{-1})}{N}}. \quad (40)$$

By [Lemma D.1](#) (Hoeffding inequality), there is an event \mathcal{E}' of probability at least $1 - \delta/2$, under which

$$\begin{aligned} \left| \hat{Z} - Z \right| &\leq \max_{y \in \mathcal{Y}} \exp(\beta^{-1}f(y)) \cdot \sqrt{\frac{8 \log(4\delta^{-1})}{N}}, \\ &= C_\infty Z \cdot \sqrt{\frac{8 \log(4\delta^{-1})}{N}}. \end{aligned} \quad (41)$$

We now condition on \mathcal{E}' . Rearranging (41) and dividing by Z , we get that

$$\frac{\hat{Z}}{Z} \leq 1 + C_\infty \cdot \sqrt{\frac{8 \log(4\delta^{-1})}{N}}, \quad \text{and} \quad \frac{\hat{Z}}{Z} \geq 1 - C_\infty \cdot \sqrt{\frac{8 \log(4\delta^{-1})}{N}}. \quad (42)$$

Therefore, using that $N \geq 16C_\infty^2 \log(4\delta^{-1})$ together with the facts that $\log(1+x) \leq x$ and $\log(1-x) \geq 1-2x$, for $x \in [0, 1/2]$, we get

$$-2C_\infty \cdot \sqrt{\frac{8 \log(4\delta^{-1})}{N}} \leq \log \frac{\hat{Z}}{Z} \leq C_\infty \cdot \sqrt{\frac{8 \log(4\delta^{-1})}{N}}. \quad (43)$$

This shows the desired bound on the log ratio $\log \frac{\hat{Z}}{Z}$ after plugging-in the choice $N = 4M \log(4\delta^{-1})$. Now, by the union bound, the probability of the event $\mathcal{E} \cap \mathcal{E}'$ is at least $1 - \delta$. Thus, the event $\mathcal{E}_{\text{accept}} = \mathcal{E} \cap \mathcal{E}'$ satisfies the desired properties.

Now, we no longer condition on \mathcal{E}' . Note that ρ is of the form

$$\frac{e^{\beta^{-1}f(y)}}{\frac{1}{N} \sum_{i=1}^N e^{\beta^{-1}f(y_i)}}. \quad (44)$$

Thus, when $|f(\cdot, \cdot)| \leq R_{\max}$, we immediately have that $\rho \in [e^{-2R_{\max}/\beta}, e^{2R_{\max}/\beta}]$ as desired. \square

Proof of [Lemma F.1](#). It is an immediate consequence of [Theorem F.2](#) that $D_{\text{TV}}(\hat{\pi}_f(x), \pi_f(x)) \leq \delta$ and $D_{\text{H}}^2(\hat{\pi}_f(x), \pi_f(x)) \leq 2\delta$. We begin by writing

$$\begin{aligned} &D_{\text{KL}}(\hat{\pi}_f(x) \parallel \pi_{\text{ref}}(x)) - D_{\text{KL}}(\pi_f(x) \parallel \pi_{\text{ref}}(x)) \\ &= \mathbb{E}_{y \sim \hat{\pi}_f(x)} [\log(\pi_f(y \mid x) / \pi_{\text{ref}}(y \mid x))] - \mathbb{E}_{y \sim \pi_f(x)} [\log(\pi_f(y \mid x) / \pi_{\text{ref}}(y \mid x))] \\ &\quad + \mathbb{E}_{y \sim \hat{\pi}_f(x)} [\log(\hat{\pi}_f(y \mid x) / \pi_f(y \mid x))] \\ &\leq \frac{2R_{\max}\delta}{\beta} + \mathbb{E}_{y \sim \hat{\pi}_f(x)} [\log(\hat{\pi}_f(y \mid x) / \pi_f(y \mid x))], \end{aligned}$$

where the inequality uses that $|\log(\pi_f(y \mid x) / \pi_{\text{ref}}(y \mid x))| \leq R_{\max}/\beta$. To proceed, we bound

$$D_{\text{KL}}(\hat{\pi}_f(x) \parallel \pi_f(x)) = \mathbb{E}_{y \sim \hat{\pi}_f(x)} [\log(\hat{\pi}_f(y \mid x) / \pi_f(y \mid x))]$$

We first note that

$$\frac{\hat{\pi}_f(y | x)}{\pi_f(y | x)} \leq \exp(R_{\max}/\beta) \cdot \frac{\hat{\pi}_f(y | x)}{\pi_{\text{ref}}(y | x)} \leq \exp(R_{\max}/\beta) \cdot N.$$

The latter inequality is a standard property of rejection sampling: If we let \mathcal{Y}_N denote the set of responses the algorithm considers accepting, then we have $\hat{\pi}_f(y | x) = \mathbb{E}_{\mathcal{Y}_N}[\hat{\pi}_f(y | x, \mathcal{Y}_N)] \leq \mathbb{E}_{\mathcal{Y}_N}[\mathbb{I}\{y \in \mathcal{Y}_N\}] \leq N \cdot \pi_{\text{ref}}(y | x)$. From here, it follows from Lemma A.10 of Foster et al. [27] that $D_{\text{KL}}(\hat{\pi}_f(x) \| \pi_f(x)) \leq 2(\frac{R_{\max}}{\beta} + \log N)D_{\text{H}}^2(\hat{\pi}_f(x), \pi_f(x)) \leq 4(\frac{R_{\max}}{\beta} + \log N)\delta$. \square

Proof of Lemma F.2. Let (y, ρ) be the random output of `SoftmaxSamplerDensity`. First, by Jensen's inequality, we have that

$$\left| \mathbb{E}[\log \rho] - \mathbb{E}\left[\log \frac{\pi_f(y | x)}{\pi_{\text{ref}}(y | x)}\right] \right| \leq \mathbb{E}\left[\left|\log \rho - \log \frac{\pi_f(y | x)}{\pi_{\text{ref}}(y | x)}\right|\right],$$

and so, by letting $\mathcal{E}_{\text{accept}}$ be the event $\mathcal{E}_{\text{accept}}$ in Theorem F.1, we have

$$\begin{aligned} &= \mathbb{E}\left[\mathbb{I}\{\mathcal{E}_{\text{accept}}\} \cdot \left|\log \rho - \log \frac{\pi_f(y | x)}{\pi_{\text{ref}}(y | x)}\right|\right] \\ &\quad + \mathbb{E}\left[(1 - \mathbb{I}\{\mathcal{E}_{\text{accept}}\}) \cdot \left|\log \rho - \log \frac{\pi_f(y | x)}{\pi_{\text{ref}}(y | x)}\right|\right], \\ &\leq C_{\infty} \cdot \sqrt{\frac{2}{M}} + \frac{4B\delta}{\beta}, \end{aligned} \tag{45}$$

where the last inequality follows from Theorem F.1 and that ρ and $\frac{\pi_f(y|x)}{\pi_{\text{ref}}(y|x)}$ are in $[e^{2B/\beta}, e^{-2B/\beta}]$.

On the other hand, we have that

$$\mathbb{E}\left[\log \frac{\pi_f(y | x)}{\pi_{\text{ref}}(y | x)}\right] = \mathbb{E}_{y \sim \hat{\pi}_f(\cdot | x)}\left[\log \frac{\pi_f(y | x)}{\pi_{\text{ref}}(y | x)}\right] \tag{46}$$

$$= \mathbb{E}_{y \sim \hat{\pi}_f(\cdot | x)}\left[\log \frac{\hat{\pi}_f(y | x)}{\pi_{\text{ref}}(y | x)}\right] - D_{\text{KL}}(\hat{\pi}_f(\cdot | x) \| \pi_f(\cdot | x)). \tag{47}$$

Now, by Lemma F.1, we have that $D_{\text{KL}}(\hat{\pi}_f(\cdot | x) \| \pi_f(\cdot | x)) \leq 4(\frac{R_{\max}}{\beta} + \log N)\delta$. Combining this with (45) and the triangle inequality, we get the desired result. \square

Proof of Lemma F.3. By Jensen's inequality and the triangle inequality, we have that

$$\begin{aligned} &|\mathbb{E}_{x \sim \rho, y \sim \pi_f}[g(x, y)] - \mathbb{E}_{x \sim \rho, y \sim \hat{\pi}}[g(x, y)]| \\ &\leq \mathbb{E}_{x \sim \rho}\left[\left|\mathbb{E}_{y \sim \pi_f(\cdot | x)}[g(x, y)] - \mathbb{E}_{y \sim \hat{\pi}(\cdot | x)}[g(x, y)]\right| \cdot \mathbb{I}\{M \geq 4C_{\text{cond}}(\pi_f | x)\}\right] \\ &\quad + \mathbb{P}_{x \sim \rho}[M < 4C_{\text{cond}}(\pi_f | x)], \\ &\leq \delta + \mathbb{P}_{x \sim \rho}[M < 4C_{\text{cond}}(\pi_f | x)], \end{aligned} \tag{48}$$

where the last inequality follows from the fact that

$$D_{\text{TV}}(\widehat{\pi}(\cdot \mid x), \pi_f(\cdot \mid x)) \leq \delta,$$

for all x such that $M \geq 4C_{\text{cond}}(\pi_f \mid x)$, thanks to [Theorem F.2](#). This completes the proof. \square

Appendix G. SpannerSampling Algorithm and Guarantee (Proof of Theorem 3.1)

In this section of the appendix, we present the pseudocode for SpannerSampling (Appendix G.1), present various intermediate technical results (Appendices G.2 and G.3), then combine these results to prove the main guarantee for SpannerSampling, Theorem 3.1 (Appendix G.4).

G.1. SpannerSampling Pseudocode

Note that within Algorithm 3, we define $\hat{\pi}^t(y, y' \mid x)$ as the distribution over $(y, y') \mid x$, induced by sampling $y' \sim \pi_{\text{ref}}(\cdot \mid x)$ and

$$y \sim \text{SoftmaxSampler}_{\beta, M_{\text{rej}}, \delta_{\text{rej}}}(r^t(x, \cdot, y'); x, \pi_{\text{ref}}).$$

We further define,

$$\hat{\pi}^t(y \mid x) := \mathbb{E}_{y' \sim \pi_{\text{ref}}(\cdot \mid x)}[\hat{\pi}^t(y, y' \mid x)]$$

as the marginal distribution over $y \mid x$ under this process.

Remark G.1 (Average-case vs. uniform spanners). *Our usage of the term “spanner” is inspired but technically different from the notion of an optimal design or barycentric spanner, which has been widely used in the linear bandit literature [3, 34, 50]. These notions provide a small collection of responses for which second moment matrix Σ achieves uniform coverage in the sense that $\max_{x, y, y'} \|\varphi(x, y, y')\|_{\Sigma^{-1}} \leq \text{poly}(d)$ or similar. For computational reasons, we cannot hope to achieve such a uniform guarantee, and instead settle for average-case coverage with respect to π_{θ^*} .*

Remark G.2 (Anchor responses). *Algorithm 3 can be slightly simplified as follows: Instead of sampling $y_2^t \sim \pi_{\text{ref}}(\cdot \mid x^t)$, we can set $y_2^t = \mathbf{y} \forall t$ for an arbitrary fixed “anchor response” \mathbf{y} . This leads to the same guarantee, but does not technically fall into the sampling oracle framework in Definition 2.1, as it requires observing the features $\phi(x^t, \mathbf{y})$ for all t . However, we use the anchor technique within our multi-turn algorithm (MTSS; Algorithm 4) to simplify presentation.*

G.2. Technical Lemmas

G.2.1. BASIC RESULTS

Lemma G.1 (Differences in rewards are linear). *If Assumption 1.1 holds, then for all $x \in \mathcal{X}$ and $y, y' \in \mathcal{Y}$,*

$$r^*(x, y) - r^*(x, y') = \langle \theta^*, \phi(x, y) - \phi(x, y') \rangle. \quad (51)$$

Proof of Lemma G.1. If Assumption 1.1 holds, then for all $x \in \mathcal{X}$, $\pi_{\beta}^*(y \mid x) = \pi_{\theta^*}(y \mid x)$, where $\pi_{\beta}^*(y \mid x) \propto \pi_{\text{ref}}(y \mid x) \exp(\beta^{-1} r^*(x, y))$ is the optimal KL-regularized policy. Taking logarithms, this implies that for all $x \in \mathcal{X}$, $y \in \mathcal{Y}$,

$$\beta \log \frac{\pi_{\beta}^*(y \mid x)}{\pi_{\text{ref}}(y \mid x)} = r^*(x, y) - \log Z_{r^*}(x) = \beta \log \frac{\pi_{\theta^*}(y \mid x)}{\pi_{\text{ref}}(y \mid x)} = \langle \theta^*, \phi(x, y) \rangle - \log Z_{\theta^*}(x),$$

where $Z_{r^*}(x) := \mathbb{E}_{y \sim \pi_{\text{ref}}(x)}[\exp(\beta^{-1} r^*(x, y))]$ and $Z_{\theta^*}(x) := \mathbb{E}_{y \sim \pi_{\text{ref}}(x)}[\exp(\beta^{-1} \langle \theta^*, \phi(x, y) \rangle)]$. Picking any $y, y' \in \mathcal{Y}$ and take the difference then implies that

$$r^*(x, y) - r^*(x, y') = \langle \theta^*, \phi(x, y) - \phi(x, y') \rangle.$$

Algorithm 3 SpannerSampling

input: Base policy π_{ref} , KL-regularization parameter $\beta > 0$, number of spanner rounds $T_{\text{span}} \in \mathbb{N}$, number of exploration rounds $T_{\text{exp}} \in \mathbb{N}$, failure probability $\delta \in (0, 1)$.

- 1: Define $\varepsilon_{\text{stat}} := c \cdot \sqrt{dR_{\text{max}}^2 \log(BR_{\text{max}}^{-1}\delta^{-1}T_{\text{exp}})}$ for abs. constant $c > 0$.
- 2: Set $\lambda \leftarrow (R_{\text{max}}/B)^2$ and $\nu := \beta/\varepsilon_{\text{stat}}$. // Spanner params.
- 3: Set $M_{\text{rej}} := 8e^2 \cdot C_{\text{cov}}(\pi_{\beta}^*)$ and $\delta_{\text{rej}} := T_{\text{exp}}^{-1}$. // Rejection sampling params.
- /* Spanner construction phase */
- 4: Initialize dataset $\mathcal{D}_{\text{span}} \leftarrow \{\emptyset\}$ and $\Psi_{\text{span}} \leftarrow \{\emptyset\}$ and set $\Sigma_{\text{span}} \leftarrow \lambda I_d$.
- 5: **for** iteration $t = 1, 2, \dots, T_{\text{prompt}}$ **do**
- 6: Observe prompt $x^t \sim \rho$.
- 7: **for** iteration $i = 1, 2, \dots, T_{\text{span}}$ **do**
- 8: Sample $(y_1^{t,i}, y_2^{t,i}) \sim \pi_{\text{ref}}(\cdot \mid x^t)$.
- 9: **if** $\|\varphi(x^t, y_1^{t,i}, y_2^{t,i})\|_{\Sigma_{\text{span}}^{-1}} > \nu$ **then** // $\varphi(x, y_1, y_2) := \phi(x, y_1) - \phi(x, y_2)$.
- 10: Observe rewards (r_1^t, r_2^t) for $(x^t, y_1^{t,i}, y_2^{t,i})$.
- 11: Update $\mathcal{D}_{\text{span}} \leftarrow \mathcal{D}_{\text{span}} \cup \{(x^t, y_1^{t,i}, y_2^{t,i}, r_1^t, r_2^t)\}$ and $\Psi_{\text{span}} \leftarrow \Psi_{\text{span}} \cup \{(x^t, y_1^{t,i}, y_2^{t,i})\}$.
- 12: $\Sigma_{\text{span}} \leftarrow \Sigma_{\text{span}} + \varphi(x^t, y_1^{t,i}, y_2^{t,i})\varphi(x^t, y_1^{t,i}, y_2^{t,i})^\top$.
- 13: **break**
- /* Exploration phase */
- 14: Initialize dataset $\mathcal{D}_{\text{exp}}^1 = \{\emptyset\}$.
- 15: **for** iteration $t = 1, 2, \dots, T_{\text{exp}}$ **do**
- /* Estimate policy and reward model */
- 16: Fit reward model via regression:

$$\theta^t \leftarrow \arg \min_{\theta \in \Theta} \sum_{(x, y_1, y_2, r_1, r_2) \in \mathcal{D}_{\text{exp}}^t \cup \mathcal{D}_{\text{span}}} (\langle \theta, \varphi(x, y_1, y_2) \rangle - (r_1 - r_2))^2. \quad (49)$$

- /* Sample responses and update dataset */
- 17: Define truncated reward function:

$$r^t(x, y, y') := \langle \theta^t, \varphi(x, y, y') \rangle \mathbb{I}\{\|\varphi(x, y, y')\|_{\Sigma_{\text{span}}^{-1}} \leq \nu\}. \quad (50)$$

- 18: Observe prompt $x^t \sim \rho$. Sample $y_2^t \sim \pi_{\text{ref}}(\cdot \mid x^t)$ and observe reward r_2^t .
- // Defines policy $\hat{\pi}^t(\cdot \mid x) \sim \text{SoftmaxSampler}_{\beta, M_{\text{rej}}, \delta_{\text{rej}}}(r^t(x, \cdot, y'); x, \pi_{\text{ref}})$ **for** $y' \sim \pi_{\text{ref}}(\cdot \mid x)$.
- 19: Sample $y_1^t \sim \text{SoftmaxSampler}_{\beta, M_{\text{rej}}, \delta_{\text{rej}}}(r^t(x^t, \cdot, y_2^t); x^t, \pi_{\text{ref}})$ and observe reward r_1^t .
- 20: Update dataset: $\mathcal{D}_{\text{exp}}^{t+1} \leftarrow \mathcal{D}_{\text{exp}}^t \cup \{(x^t, y_1^t, y_2^t, r_1^t, r_2^t)\}$.
- 21: **return** $\hat{\pi} \sim \text{unif}(\hat{\pi}^1, \dots, \hat{\pi}^{T_{\text{exp}}})$.

as claimed. \square

Lemma G.2 (Density ratio bound for softmax policies). *For a function $f : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$, let*

$$\pi_f(y | x) \propto \pi_{\text{ref}}(y | x) \exp(\beta^{-1} f(x, y)), \quad (52)$$

Then for all $x \in \mathcal{X}$, it holds that

$$\left\| \frac{\pi_f(\cdot | x)}{\pi_{\text{ref}}(\cdot | x)} \right\|_{\infty} \leq \exp \left(\beta^{-1} \left(\max_{y \in \mathcal{Y}} f(x, y) - \mathbb{E}_{y \sim \pi_{\text{ref}}(x)}[f(x, y)] \right) \right). \quad (53)$$

Proof of Lemma G.2. For any $y \in \mathcal{Y}$, we can use Jensen's inequality to bound

$$\begin{aligned} \frac{\pi_f(y | x)}{\pi_{\text{ref}}(y | x)} &= \frac{\exp(\beta^{-1} f(x, y))}{\mathbb{E}_{y' \sim \pi_{\text{ref}}(x)}[\exp(\beta^{-1} f(x, y'))]} \\ &= \frac{\exp(\beta^{-1} (f(x, y) - \mathbb{E}_{y'' \sim \pi_{\text{ref}}(x)}[f(x, y'')]))}{\mathbb{E}_{y' \sim \pi_{\text{ref}}(x)}[\exp(\beta^{-1} (f(x, y') - \mathbb{E}_{y'' \sim \pi_{\text{ref}}(x)}[f(x, y'')]))]} \\ &\leq \frac{\exp(\beta^{-1} (f(x, y) - \mathbb{E}_{y'' \sim \pi_{\text{ref}}(x)}[f(x, y'')]))}{\exp(\beta^{-1} (\mathbb{E}_{y' \sim \pi_{\text{ref}}(x)}[f(x, y')] - \mathbb{E}_{y'' \sim \pi_{\text{ref}}(x)}[f(x, y'')]))} \\ &= \exp(\beta^{-1} (f(x, y) - \mathbb{E}_{y'' \sim \pi_{\text{ref}}(x)}[f(x, y'')])). \end{aligned}$$

as claimed. \square

G.2.2. GUARANTEES FOR LEAST SQUARES

The following result presents a standard guarantee for least squares with dependent data.

Lemma G.3. *Consider a sequentially generated dataset $\{(x^t, y_1^t, y_2^t, r_1^t, r_2^t)\}_{t \in [T]}$ in which for all t ,*

$$\mathbb{E}[r_1^t - r_2^t | x^t, y_1^t, y_2^t, \mathcal{F}^{t-1}] = \langle \theta^*, \phi(x^t, y_1^t) - \phi(x^t, y_2^t) \rangle,$$

where $\mathcal{F}^{t-1} := \sigma(\{(x^i, y_1^i, y_2^i, r_1^i, r_2^i)\}_{i < t})$. Define the least-squares estimator

$$\theta^t = \arg \min_{\theta \in \Theta} \sum_{i < t} (\langle \phi(x^i, y_1^i) - \phi(x^i, y_2^i), \theta \rangle - (r_1^i - r_2^i))^2,$$

and let $\Sigma^t := \sum_{i < t} (\phi(x^i, y_1^i) - \phi(x^i, y_2^i))(\phi(x^i, y_1^i) - \phi(x^i, y_2^i))^{\top}$. Assume that $r_1^t, r_2^t \in [0, R_{\max}]$ almost surely, that $\theta^ \in \Theta$, and that [Assumption 2.1](#) holds with parameter B . Define $\lambda = \frac{R_{\max}^2}{B^2}$. Then with probability at least $1 - \delta$, for all $t \in [T]$,*

$$\|\theta^t - \theta^*\|_{\Sigma^t + \lambda I_d}^2 \leq O(d R_{\max}^2 \log(B R_{\max}^{-1} \delta^{-1} T)).$$

Proof of Lemma G.3. By a standard concentration result for well-specified regression (e.g., Lemma 39 in Jin et al. [39]), we have that with probability at least $1 - \delta$, for all $t \in [T]$,

$$\|\theta^t - \theta^*\|_{\Sigma^t}^2 = \sum_{i < t} \langle \theta^t - \theta^*, \phi(x^i, y_1^i) - \phi(x^i, y_2^i) \rangle^2 \lesssim O(d R_{\max}^2 \log(B R_{\max}^{-1} \delta^{-1} T)).$$

By [Assumption 2.1](#) and choice of λ , we have

$$\|\theta^t - \theta^*\|_{\Sigma^t + \lambda I_d}^2 = \|\theta^t - \theta^*\|_{\Sigma^t}^2 + \lambda \|\theta^t - \theta^*\|^2 \leq \|\theta^t - \theta^*\|_{\Sigma^t}^2 + 4\lambda B^2 \leq \|\theta^t - \theta^*\|_{\Sigma^t}^2 + 4R_{\max}^2,$$

and combining with the preceding bound completes the proof. \square

G.2.3. ELEMENTARY PROPERTIES OF KL-REGULARIZED REGRET

We now state some generic properties of the KL-regularized regret. Suppose the true reward is $f^*(x, y)$ for an arbitrary function f^* , and let

$$J_\beta(\pi; f^*, x) = \mathbb{E}_{y \sim \pi(x)}[f^*(x, y)] - \beta D_{\text{KL}}(\pi(x) \parallel \pi_{\text{ref}}(x)).$$

For a function f , let

$$\pi_f(y \mid x) \propto \pi_{\text{ref}}(y \mid x) \exp(\beta^{-1} f(x, y)),$$

and let $Z_f(x) := \mathbb{E}_{y \sim \pi_{\text{ref}}(\cdot \mid x)}[\exp(\beta^{-1} f(x, y))]$ denote the normalization constant. The following result follows from elementary manipulations.

Lemma G.4. *For all $f : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$ and $x \in \mathcal{X}$, it holds that*

$$\begin{aligned} & J_\beta(\pi_{f^*}; f^*, x) - J_\beta(\pi_f; f^*, x) \\ &= \beta \cdot D_{\text{KL}}(\pi_f(x) \parallel \pi_{f^*}(x)) \\ &= \beta \log(Z_{f^*}(x)/Z_f(x)) + \mathbb{E}_{y \sim \pi_f(x)}[f(x, y) - f^*(x, y)]. \\ &= \beta \log\left(\mathbb{E}_{y \sim \pi_f(x)} \exp(\beta^{-1}(f^*(x, y) - f(x, y)))\right) + \mathbb{E}_{y \sim \pi_f(x)}[f(x, y) - f^*(x, y)]. \end{aligned}$$

Proof of Lemma G.4. We can directly calculate that

$$\beta D_{\text{KL}}(\pi_f \parallel \pi_{f^*}) = \beta \mathbb{E}[\log(Z_{f^*}(x)/Z_f(x))] + \mathbb{E}_{y \sim \pi_f(x)}[f(x, y) - f^*(x, y)],$$

where $Z_f(x) := \mathbb{E}_{y \sim \pi_{\text{ref}}(x)}[\exp(\beta^{-1} f(x, y))]$. The first identity now follows by noting that for any f ,

$$J_\beta(\pi_f; f^*, x) = \mathbb{E}_{y \sim \pi_f(x)}[f^*(x, y)] - \mathbb{E}_{y \sim \pi_f(x)}[f(x, y)] + \beta \log(Z_f(x)).$$

We finally observe that

$$\log(Z_{f^*}(x)/Z_f(x)) = \log\left(\mathbb{E}_{y \sim \pi_f(x)}[\exp(\beta^{-1}(f^*(x, y) - f(x, y)))]\right)$$

which completes the proof. \square

G.3. KL-Regularized Regret Decomposition for Truncated Softmax Policies

This section gives tight bounds on the KL-regularized regret for truncated softmax policies of the type used in `SpannerSampling` and formally defined below. The main results, [Lemmas G.6](#) and [G.7](#), allow for fast $1/\varepsilon$ -type rates by exploiting regularization, as well as efficient rejection sampling.

Truncated softmax policies. Let $\Sigma \succ 0$ be a given matrix and $\nu > 0$ be a parameter. Recalling that $\varphi(x, y, y') := \phi(x, y) - \phi(x, y')$, define a *truncated* feature map by

$$\bar{\varphi}(x, y, y') = \varphi(x, y, y') \mathbb{I}\{\|\bar{\varphi}(x, y, y')\|_{\Sigma^{-1}} \leq \nu\}.$$

For a parameter $\theta \in \mathbb{R}^d$ we will define a *truncated softmax policy* $\bar{\pi}_\theta(y, y' | x)$ as follows. Fixing $x \in \mathcal{X}$, first define $\bar{\pi}_\theta(y' | x) = \pi_{\text{ref}}(y' | x)$. Next, define

$$\bar{\pi}_\theta(y | x, y') \propto \pi_{\text{ref}}(y | x) \exp(\beta^{-1} \langle \theta, \bar{\varphi}(x, y, y') \rangle).$$

We use $\bar{\pi}_\theta(y | x) = \mathbb{E}_{y' \sim \pi_{\text{ref}}(\cdot | x)}[\bar{\pi}_\theta(y | x, y')]$ to denote the marginal over y given x . We will overload notation slightly and use $J_\beta(\bar{\pi}_\theta)$ to denote the KL-regularized regret of the marginalized policy $\bar{\pi}_\theta(y | x)$. We also define

$$\bar{J}_\beta(\bar{\pi}_\theta) = \mathbb{E}_{x \sim \rho, y' \sim \pi_{\text{ref}}(\cdot | x), y \sim \bar{\pi}_\theta(\cdot | x, y')} [r^*(x, y) - \beta D_{\text{KL}}(\bar{\pi}_\theta(\cdot | x, y') \| \pi_{\text{ref}}(\cdot | x))]$$

and, for any $x \in \mathcal{X}$,

$$\bar{J}_\beta(\bar{\pi}_\theta; x) = \mathbb{E}_{y' \sim \pi_{\text{ref}}(\cdot | x), y \sim \bar{\pi}_\theta(\cdot | x, y')} [r^*(x, y) - \beta D_{\text{KL}}(\bar{\pi}_\theta(\cdot | x, y') \| \pi_{\text{ref}}(\cdot | x))].$$

We first give a bound on the regret of the truncated softmax policies that scales with (i) the squared estimation error (allowing for fast $1/\varepsilon$ -type rates), and (ii) truncation probability under responses drawn from π_{ref} and π_{θ^*} .

Lemma G.5 (Basic regret decomposition for truncated softmax policies). *Fix $x \in \mathcal{X}$ and define $\varepsilon_{\text{stat}}^2 := \|\theta - \theta^*\|_\Sigma^2$. Then under [Assumptions 1.1](#) and [2.1](#), if $\nu \leq \beta/\varepsilon_{\text{stat}}$, we have*

$$\begin{aligned} J_\beta(\pi_{\theta^*}; x) - J_\beta(\bar{\pi}_\theta; x) &\leq J_\beta(\pi_{\theta^*}; x) - \bar{J}_\beta(\bar{\pi}_\theta; x) \\ &\leq \beta^{-1} \mathbb{E}_{(y, y') \sim \bar{\pi}_\theta(x)} [\langle \theta^* - \theta, \bar{\varphi}(x, y, y') \rangle^2] \\ &\quad + R_{\max} \left(\mathbb{P}_{y \sim \pi_{\theta^*}(x), y' \sim \pi_{\text{ref}}(x)} [\|\varphi(x, y, y')\|_{\Sigma^{-1}} > \nu] \right. \\ &\quad \left. + \mathbb{P}_{(y, y') \sim \bar{\pi}_\theta(x)} [\|\varphi(x, y, y')\|_{\Sigma^{-1}} > \nu] \right). \end{aligned}$$

Next, we show that the density ratio between $\bar{\pi}_\theta$ and π_{ref} can be bounded by the optimal density ratio $C_{\text{cov}}(\pi_{\theta^*})$ for π_{θ^*} .

Lemma G.6 (Density ratio bound for truncated softmax policies). *Fix $x \in \mathcal{X}$ and $y' \in \mathcal{Y}$. Define*

$$\varepsilon_{\text{span}}(x, y') := \mathbb{P}_{y \sim \pi_{\theta^*}(\cdot | x)} [\|\varphi(x, y, y')\|_{\Sigma^{-1}} > \nu], \quad \text{and} \quad \varepsilon_{\text{stat}}^2 := \|\theta - \theta^*\|_\Sigma^2.$$

Suppose $\nu \leq \beta/\varepsilon_{\text{stat}}$ and $\varepsilon_{\text{span}}(x, y') \leq 1/2$. Then for all $y \in \mathcal{Y}$,

$$\bar{\pi}_\theta(y | x, y') \leq 2e^2 C_{\text{cov}}(\pi_{\theta^*}) \cdot \pi_{\text{ref}}(y | x).$$

Finally [Lemma G.5](#) and [Lemma G.6](#) gives our main regret decomposition for truncated softmax policies.

Lemma G.7 (Main regret decomposition for truncated softmax policies). *Define $\varepsilon_{\text{stat}}^2 := \|\theta - \theta^*\|_\Sigma^2$, and for any $\varepsilon > 0$, define*

$$\mathcal{X}_{\text{span}}(\varepsilon) := \{x \in \mathcal{X} \mid \mathbb{P}_{(y, y') \sim \pi_{\text{ref}}(x)} [\|\varphi(x, y, y')\|_{\Sigma^{-1}} > \nu] \leq \varepsilon\}.$$

Then under [Assumptions 1.1](#) and [2.1](#), if $\nu \leq \beta/\varepsilon_{\text{stat}}$,

$$\begin{aligned} J_\beta(\pi_{\theta^*}) - J_\beta(\bar{\pi}_\theta) &\leq J_\beta(\pi_{\theta^*}) - \bar{J}_\beta(\bar{\pi}_\theta) \\ &\leq \beta^{-1} \mathbb{E}_{(y, y') \sim \bar{\pi}_\theta(x)} [\langle \theta^* - \theta, \bar{\varphi}(x, y, y') \rangle^2] + 18R_{\max} C_{\text{cov}}(\pi_{\theta^*}) \cdot \varepsilon + 2R_{\max} \mathbb{P}[x \notin \mathcal{X}_{\text{span}}(\varepsilon)]. \end{aligned}$$

G.3.1. PROOF OF LEMMAS G.5 THROUGH G.7

Proof of Lemma G.5. To keep notation compact, in this proof we omit all dependence on the fixed $x \in \mathcal{X}$ (so that below, $J_\beta(\pi_{\theta^*})$ refers to $J_\beta(\pi_{\theta^*}; x)$, π_{θ^*} refers to $\pi_{\theta^*}(x)$, etc.). We have

$$\begin{aligned}
& J_\beta(\pi_{\theta^*}) - J_\beta(\bar{\pi}_\theta) \\
&= \mathbb{E}_{y \sim \pi_{\theta^*}}[r^*(y)] - \mathbb{E}_{y \sim \bar{\pi}_\theta}[r^*(y)] - \beta D_{\text{KL}}(\pi_{\theta^*} \parallel \pi_{\text{ref}}) + \beta D_{\text{KL}}(\bar{\pi}_\theta \parallel \pi_{\text{ref}}) \\
&\leq \mathbb{E}_{y' \sim \pi_{\text{ref}}}[\mathbb{E}_{y \sim \pi_{\theta^*}}[r^*(y)] - \mathbb{E}_{y \sim \bar{\pi}_\theta(\cdot | y')}[r^*(y)] - \beta D_{\text{KL}}(\pi_{\theta^*} \parallel \pi_{\text{ref}}) + \beta D_{\text{KL}}(\bar{\pi}_\theta(\cdot | y') \parallel \pi_{\text{ref}})], \\
&= J_\beta(\pi_{\theta^*}) - \bar{J}_\beta(\bar{\pi}_\theta)
\end{aligned}$$

since, by convexity of KL-divergence,

$$D_{\text{KL}}(\bar{\pi}_\theta \parallel \pi_{\text{ref}}) = D_{\text{KL}}(\mathbb{E}_{y' \sim \pi_{\text{ref}}}[\bar{\pi}_\theta(\cdot | y')] \parallel \pi_{\text{ref}}) \leq \mathbb{E}_{y' \sim \pi_{\text{ref}}}[D_{\text{KL}}(\bar{\pi}_\theta(\cdot | y') \parallel \pi_{\text{ref}})].$$

Under [Assumption 1.1](#), we can further write the quantity above as

$$\begin{aligned}
& \mathbb{E}_{y' \sim \pi_{\text{ref}}}[\mathbb{E}_{y \sim \pi_{\theta^*}}[r^*(y)] - \mathbb{E}_{y \sim \bar{\pi}_\theta(\cdot | y')}[r^*(y)] - \beta D_{\text{KL}}(\pi_{\theta^*} \parallel \pi_{\text{ref}}) + \beta D_{\text{KL}}(\bar{\pi}_\theta(\cdot | y') \parallel \pi_{\text{ref}})] \\
&= \mathbb{E}_{y' \sim \pi_{\text{ref}}}[\mathbb{E}_{y \sim \pi_{\theta^*}}[r^*(y) - r^*(y')] - \mathbb{E}_{y \sim \bar{\pi}_\theta(\cdot | y')}[r^*(y) - r^*(y')] - \beta D_{\text{KL}}(\pi_{\theta^*} \parallel \pi_{\text{ref}}) + \beta D_{\text{KL}}(\bar{\pi}_\theta(\cdot | y') \parallel \pi_{\text{ref}})] \\
&= \mathbb{E}_{y' \sim \pi_{\text{ref}}}[\mathbb{E}_{y \sim \pi_{\theta^*}}[\langle \theta^*, \varphi(y, y') \rangle] - \mathbb{E}_{y \sim \bar{\pi}_\theta(\cdot | y')}[\langle \theta^*, \varphi(y, y') \rangle] - \beta D_{\text{KL}}(\pi_{\theta^*} \parallel \pi_{\text{ref}}) + \beta D_{\text{KL}}(\bar{\pi}_\theta(\cdot | y') \parallel \pi_{\text{ref}})] \\
&\leq \mathbb{E}_{y' \sim \pi_{\text{ref}}}[\mathbb{E}_{y \sim \pi_{\theta^*}}[\langle \theta^*, \bar{\varphi}(y, y') \rangle] - \mathbb{E}_{y \sim \bar{\pi}_\theta(\cdot | y')}[\langle \theta^*, \bar{\varphi}(y, y') \rangle] - \beta D_{\text{KL}}(\pi_{\theta^*} \parallel \pi_{\text{ref}}) + \beta D_{\text{KL}}(\bar{\pi}_\theta(\cdot | y') \parallel \pi_{\text{ref}})] \\
&\quad + R_{\max}(\mathbb{P}_{y \sim \pi_{\theta^*}, y' \sim \pi_{\text{ref}}}[\|\varphi(y, y')\|_{\Sigma^{-1}} > \nu] + \mathbb{P}_{(y, y') \sim \bar{\pi}_\theta}[\|\varphi(y, y')\|_{\Sigma^{-1}} > \nu]) \\
&\leq \mathbb{E}_{y' \sim \pi_{\text{ref}}}[\mathbb{E}_{y \sim \bar{\pi}_{\theta^*}(\cdot | y')}[\langle \theta^*, \bar{\varphi}(y, y') \rangle] - \mathbb{E}_{y \sim \bar{\pi}_\theta(\cdot | y')}[\langle \theta^*, \bar{\varphi}(y, y') \rangle] \\
&\quad - \beta D_{\text{KL}}(\bar{\pi}_{\theta^*}(\cdot | y') \parallel \pi_{\text{ref}}) + \beta D_{\text{KL}}(\bar{\pi}_\theta(\cdot | y') \parallel \pi_{\text{ref}})] \\
&\quad + R_{\max}(\mathbb{P}_{y \sim \pi_{\theta^*}, y' \sim \pi_{\text{ref}}}[\|\varphi(y, y')\|_{\Sigma^{-1}} > \nu] + \mathbb{P}_{(y, y') \sim \bar{\pi}_\theta}[\|\varphi(y, y')\|_{\Sigma^{-1}} > \nu]), \tag{54}
\end{aligned}$$

where the first inequality is by definition of $\bar{\varphi}$, and the second is because for any fixed y' , we have

$$\begin{aligned}
& \mathbb{E}_{y \sim \pi_{\theta^*}}[\langle \theta^*, \bar{\varphi}(y, y') \rangle] - \beta D_{\text{KL}}(\pi_{\theta^*} \parallel \pi_{\text{ref}}) \\
&\leq \max_{\pi: \mathcal{X} \rightarrow \Delta(\mathcal{Y})} \{ \mathbb{E}_{y \sim \pi}[\langle \theta^*, \bar{\varphi}(y, y') \rangle] - \beta D_{\text{KL}}(\pi \parallel \pi_{\text{ref}}) \} \\
&= \mathbb{E}_{y \sim \bar{\pi}_{\theta^*}(\cdot | y')}[\langle \theta^*, \bar{\varphi}(y, y') \rangle] - \beta D_{\text{KL}}(\bar{\pi}_{\theta^*}(\cdot | y') \parallel \pi_{\text{ref}}).
\end{aligned}$$

With this upper bound, for any fixed y' , we can interpret the quantity

$$\mathbb{E}_{y \sim \bar{\pi}_{\theta^*}(\cdot | y')}[\langle \theta^*, \bar{\varphi}(y, y') \rangle] - \mathbb{E}_{y \sim \bar{\pi}_\theta(\cdot | y')}[\langle \theta^*, \bar{\varphi}(y, y') \rangle] - \beta D_{\text{KL}}(\bar{\pi}_{\theta^*}(\cdot | y') \parallel \pi_{\text{ref}}) + \beta D_{\text{KL}}(\bar{\pi}_\theta(\cdot | y') \parallel \pi_{\text{ref}})$$

in [Eq. \(54\)](#) as the KL-regularized regret of $\bar{\pi}_\theta(\cdot | y')$ to $\bar{\pi}_{\theta^*}(\cdot | y')$ under the reward $\langle \theta^*, \varphi(y, y') \rangle$. Consequently, [Lemma G.4](#) allows us to bound this regret by

$$\beta \log(\mathbb{E}_{y \sim \bar{\pi}_\theta(\cdot | y')} \exp(\beta^{-1} \langle \theta^* - \theta, \bar{\varphi}(y, y') \rangle)) + \mathbb{E}_{y \sim \bar{\pi}_\theta(\cdot | y')}[\langle \theta - \theta^*, \bar{\varphi}(y, y') \rangle]. \tag{55}$$

Note that for all $y, y' \in \mathcal{Y}$, under the condition on ν in the lemma statement,

$$|\langle \theta - \theta^*, \bar{\varphi}(y, y') \rangle| \leq \|\bar{\varphi}(y, y')\|_{\Sigma^{-1}} \|\theta - \theta^*\|_{\Sigma} \leq \nu \varepsilon_{\text{stat}} \leq \beta, \tag{56}$$

since $\bar{\varphi}(y, y') = \mathbf{0}$ if $\|\varphi(y, y')\|_{\Sigma^{-1}} \leq \nu$ does not hold. Hence, using that $e^z \leq 1 + z + z^2$ for all $z \leq 1$, we have

$$\begin{aligned} & \beta \log(\mathbb{E}_{y \sim \bar{\pi}_\theta(\cdot | y')} [\exp(\beta^{-1} \langle \theta^* - \theta, \bar{\varphi}(y, y') \rangle)]) \\ & \leq \beta \log\left(1 + \beta^{-1} \mathbb{E}_{y \sim \bar{\pi}_\theta(\cdot | y')} [\langle \theta^* - \theta, \bar{\varphi}(y, y') \rangle] + \beta^{-2} \mathbb{E}_{y \sim \bar{\pi}_\theta(\cdot | y')} [\langle \theta^* - \theta, \bar{\varphi}(y, y') \rangle^2]\right) \\ & \leq \mathbb{E}_{y \sim \bar{\pi}_\theta(\cdot | y')} [\langle \theta^* - \theta, \bar{\varphi}(y, y') \rangle] + \beta^{-1} \mathbb{E}_{y \sim \bar{\pi}_\theta(\cdot | y')} [\langle \theta^* - \theta, \bar{\varphi}(y, y') \rangle^2], \end{aligned}$$

which we can substitute into the bound from [Eq. \(55\)](#) (cancelling out the linear term) to get the bound

$$\begin{aligned} & \mathbb{E}_{y \sim \bar{\pi}_{\theta^*}(\cdot | y')} [\langle \theta^*, \bar{\varphi}(y, y') \rangle] - \mathbb{E}_{y \sim \bar{\pi}_\theta(\cdot | y')} [\langle \theta^*, \bar{\varphi}(y, y') \rangle] - \beta D_{\text{KL}}(\bar{\pi}_{\theta^*}(\cdot | y') \| \pi_{\text{ref}}) + \beta D_{\text{KL}}(\bar{\pi}_\theta(\cdot | y') \| \pi_{\text{ref}}) \\ & \leq \beta^{-1} \mathbb{E}_{y \sim \bar{\pi}_\theta(\cdot | y')} [\langle \theta^* - \theta, \bar{\varphi}(y, y') \rangle^2]. \end{aligned}$$

Since this holds uniformly for all $y' \in \mathcal{Y}$, returning to [Eq. \(54\)](#), we conclude that

$$\begin{aligned} J_\beta(\pi_{\theta^*}) - J_\beta(\bar{\pi}_\theta) & \leq \beta^{-1} \mathbb{E}_{(y, y') \sim \bar{\pi}_\theta} [\langle \theta^* - \theta, \bar{\varphi}(y, y') \rangle^2] \\ & \quad + R_{\max}(\mathbb{P}_{y \sim \pi_{\theta^*}, y' \sim \pi_{\text{ref}}} [\|\varphi(y, y')\|_{\Sigma^{-1}} > \nu] + \mathbb{P}_{(y, y') \sim \bar{\pi}_\theta} [\|\varphi(y, y')\|_{\Sigma^{-1}} > \nu]) \end{aligned}$$

as claimed. \square

Proof of [Lemma G.6](#). To keep notation compact, we again omit all dependence on x . Fix $y' \in \mathcal{Y}$. Then we have

$$\frac{\bar{\pi}_\theta(y | y')}{\pi_{\text{ref}}(y)} = \frac{\exp(\beta^{-1} \langle \theta, \bar{\varphi}(y, y') \rangle)}{Z_{\bar{\pi}_\theta}(y')},$$

where $Z_{\bar{\pi}_\theta}(y') := \mathbb{E}_{y \sim \pi_{\text{ref}}} [\exp(\beta^{-1} \langle \theta, \bar{\varphi}(y, y') \rangle)]$. Note that for all $y, y' \in \mathcal{Y}$, under the conditions in the lemma statement, we have

$$|\langle \theta - \theta^*, \bar{\varphi}(y, y') \rangle| \leq \|\bar{\varphi}(y, y')\|_{\Sigma^{-1}} \|\theta - \theta^*\|_\Sigma \leq \nu \varepsilon_{\text{stat}} \leq \beta. \quad (57)$$

We begin by giving a lower bound on the normalization constant $Z_{\bar{\pi}_\theta}(y')$. Observe that

$$\begin{aligned} Z_{\bar{\pi}_\theta}(y') & := \mathbb{E}_{y \sim \pi_{\text{ref}}} [\exp(\beta^{-1} \langle \theta, \bar{\varphi}(y, y') \rangle)] \\ & \geq \mathbb{E}_{y \sim \pi_{\text{ref}}} [\exp(\beta^{-1} \langle \theta, \bar{\varphi}(y, y') \rangle) \mathbb{I}\{\|\varphi(y, y')\|_{\Sigma^{-1}} \leq \nu\}] \\ & \geq e^{-1} \mathbb{E}_{y \sim \pi_{\text{ref}}} [\exp(\beta^{-1} \langle \theta^*, \bar{\varphi}(y, y') \rangle) \mathbb{I}\{\|\varphi(y, y')\|_{\Sigma^{-1}} \leq \nu\}] \\ & = e^{-1} \mathbb{E}_{y \sim \pi_{\text{ref}}} [\exp(\beta^{-1} \langle \theta^*, \varphi(y, y') \rangle) \mathbb{I}\{\|\varphi(y, y')\|_{\Sigma^{-1}} \leq \nu\}], \\ & = e^{-1} \mathbb{E}_{y \sim \pi_{\text{ref}}} [\exp(\beta^{-1} \langle \theta^*, \phi(y) \rangle) \mathbb{I}\{\|\varphi(y, y')\|_{\Sigma^{-1}} \leq \nu\}] \cdot \exp(-\beta^{-1} \langle \theta^*, \phi(y') \rangle), \end{aligned}$$

where the second inequality uses [Eq. \(57\)](#) and the second-to-last equality uses the definition of the indicator. Now, define

$$Z_{\pi_{\theta^*}} = \mathbb{E}_{y \sim \pi_{\text{ref}}} [\exp(\beta^{-1} \langle \theta^*, \phi(y) \rangle)]$$

as the normalization constant for π_{θ^*} . We can write

$$\mathbb{E}_{y \sim \pi_{\text{ref}}} [\exp(\beta^{-1} \langle \theta^*, \phi(y) \rangle) \mathbb{I}\{\|\varphi(y, y')\|_{\Sigma^{-1}} \leq \nu\}]$$

$$= Z_{\pi_{\theta^*}} - \mathbb{E}_{y \sim \pi_{\text{ref}}} [\exp(\beta^{-1} \langle \theta^*, \phi(y) \rangle) \mathbb{I}\{\|\varphi(y, y')\|_{\Sigma^{-1}} > \nu\}].$$

We can further bound

$$\begin{aligned} & \mathbb{E}_{y \sim \pi_{\text{ref}}} [\exp(\beta^{-1} \langle \theta^*, \phi(y) \rangle) \mathbb{I}\{\|\varphi(y, y')\|_{\Sigma^{-1}} > \nu\}] \\ &= \mathbb{E}_{y \sim \pi_{\text{ref}}} \left[\frac{\exp(\beta^{-1} \langle \theta^*, \phi(y) \rangle)}{Z_{\pi_{\theta^*}}} \mathbb{I}\{\|\varphi(y, y')\|_{\Sigma^{-1}} > \nu\} \right] \cdot Z_{\pi_{\theta^*}} \\ &= \mathbb{E}_{y \sim \pi_{\theta^*}} [\mathbb{I}\{\|\varphi(y, y')\|_{\Sigma^{-1}} > \nu\}] \cdot Z_{\pi_{\theta^*}}. \end{aligned}$$

It follows that as long as $\varepsilon_{\text{span}}(x, y') := \mathbb{E}_{y \sim \pi_{\theta^*}} [\mathbb{I}\{\|\varphi(y, y')\|_{\Sigma^{-1}} > \nu\}] \leq 1/2$, we have

$$\mathbb{E}_{y \sim \pi_{\text{ref}}} [\exp(\beta^{-1} \langle \theta^*, \phi(y) \rangle) \mathbb{I}\{\|\varphi(y, y')\|_{\Sigma^{-1}} \leq \nu\}] \geq \frac{1}{2} Z_{\pi_{\theta^*}}.$$

Combining this with the preceding steps gives

$$\begin{aligned} \frac{\bar{\pi}_{\theta}(y | y')}{\pi_{\text{ref}}(y)} &\leq 2e \cdot \frac{\exp(\beta^{-1} (\langle \theta, \bar{\varphi}(y, y') \rangle + \langle \theta^*, \phi(y') \rangle))}{Z_{\pi_{\theta^*}}} \\ &\leq 2e^2 \cdot \frac{\exp(\beta^{-1} (\langle \theta^*, \bar{\varphi}(y, y') \rangle + \langle \theta^*, \phi(y') \rangle))}{Z_{\pi_{\theta^*}}} \end{aligned}$$

where the second inequality is by [Eq. \(57\)](#). To proceed, we consider two cases. First, if $\langle \theta^*, \varphi(y, y') \rangle \geq 0$, then

$$\langle \theta^*, \bar{\varphi}(y, y') \rangle + \langle \theta^*, \phi(y') \rangle \leq \langle \theta^*, \varphi(y, y') \rangle + \langle \theta^*, \phi(y') \rangle = \langle \theta^*, \phi(y) \rangle.$$

Otherwise,

$$\langle \theta^*, \bar{\varphi}(y, y') \rangle + \langle \theta^*, \phi(y') \rangle \leq \langle \theta^*, \phi(y') \rangle.$$

Combining these cases gives

$$\begin{aligned} \frac{\exp(\beta^{-1} (\langle \theta^*, \bar{\varphi}(y, y') \rangle + \langle \theta^*, \phi(y') \rangle))}{Z_{\pi_{\theta^*}}} &\leq \max \left\{ \frac{\exp(\beta^{-1} \langle \theta^*, \phi(y) \rangle)}{Z_{\pi_{\theta^*}}}, \frac{\exp(\beta^{-1} \langle \theta^*, \phi(y') \rangle)}{Z_{\pi_{\theta^*}}} \right\} \\ &= \max \left\{ \frac{\pi_{\theta^*}(y)}{\pi_{\text{ref}}(y)}, \frac{\pi_{\theta^*}(y')}{\pi_{\text{ref}}(y')} \right\} \\ &\leq C_{\text{cov}}(\pi_{\theta^*}) \end{aligned}$$

which completes the proof. \square

Proof of [Lemma G.7](#). By [Lemma G.5](#), taking expectation over $x \sim \rho$, we have

$$\begin{aligned} J_{\beta}(\pi_{\theta^*}) - J_{\beta}(\bar{\pi}_{\theta}) &\leq J_{\beta}(\pi_{\theta^*}) - \bar{J}_{\beta}(\bar{\pi}_{\theta}) \\ &\leq \beta^{-1} \mathbb{E}_{x \sim \rho, (y, y') \sim \bar{\pi}_{\theta}(x)} \left[\langle \theta^* - \theta, \bar{\varphi}(x, y, y') \rangle^2 \right] \\ &\quad + R_{\max} \mathbb{P}_{x \sim \rho, y \sim \pi_{\theta^*}(x), y' \sim \pi_{\text{ref}}(x)} [\|\varphi(x, y, y')\|_{\Sigma^{-1}} > \nu] \\ &\quad + R_{\max} \mathbb{P}_{x \sim \rho, (y, y') \sim \bar{\pi}_{\theta}(x)} [\|\varphi(x, y, y')\|_{\Sigma^{-1}} > \nu]. \end{aligned}$$

We need to bound the second and third terms. Let $\varepsilon > 0$ be fixed, and let us abbreviate

$$\mathcal{X}_{\text{span}} \equiv \mathcal{X}_{\text{span}}(\varepsilon) = \{x \in \mathcal{X} \mid \mathbb{P}_{(y,y') \sim \pi_{\text{ref}}(x)} [\|\varphi(x, y, y')\|_{\Sigma^{-1}} > \nu] \leq \varepsilon\}.$$

For the second term, it is immediate that

$$\begin{aligned} & \mathbb{P}_{x \sim \rho, y \sim \pi_{\theta^*}(x), y' \sim \pi_{\text{ref}}(x)} [\|\varphi(x, y, y')\|_{\Sigma^{-1}} > \nu] \\ & \leq \mathbb{E}_{x \sim \rho} [\mathbb{P}_{y \sim \pi_{\theta^*}(x), y' \sim \pi_{\text{ref}}(x)} [\|\varphi(x, y, y')\|_{\Sigma^{-1}} > \nu] \mathbb{I}\{x \in \mathcal{X}_{\text{span}}\}] + \mathbb{P}_{x \sim \rho}[x \notin \mathcal{X}_{\text{span}}] \\ & \leq C_{\text{cov}}(\pi_{\theta^*}) \cdot \mathbb{E}_{x \sim \rho} [\mathbb{P}_{(y,y') \sim \pi_{\text{ref}}(x)} [\|\varphi(x, y, y')\|_{\Sigma^{-1}} > \nu] \mathbb{I}\{x \in \mathcal{X}_{\text{span}}\}] + \mathbb{P}_{x \sim \rho}[x \notin \mathcal{X}_{\text{span}}] \\ & \leq C_{\text{cov}}(\pi_{\theta^*}) \cdot \varepsilon + \mathbb{P}_{x \sim \rho}[x \notin \mathcal{X}_{\text{span}}]. \end{aligned}$$

To handle the third term, define

$$\mathcal{Z}_{\text{good}} := \left\{ (x, y') \in \mathcal{X} \times \mathcal{Y} \mid \mathbb{P}_{y \sim \pi_{\theta^*}(\cdot|x)} [\|\varphi(x, y, y')\|_{\Sigma^{-1}} > \nu] \leq \frac{1}{2} \right\}.$$

We can bound

$$\begin{aligned} & \mathbb{P}_{x \sim \rho, (y,y') \sim \bar{\pi}_{\theta}(x)} [\|\varphi(x, y, y')\|_{\Sigma^{-1}} > \nu] \\ & = \mathbb{P}_{x \sim \rho, y' \sim \pi_{\text{ref}}(\cdot|x), y' \sim \bar{\pi}_{\theta}(\cdot|x, y')} [\|\varphi(x, y, y')\|_{\Sigma^{-1}} > \nu] \\ & \leq \mathbb{E}_{x \sim \rho, y' \sim \pi_{\text{ref}}(\cdot|x)} [\mathbb{P}_{y' \sim \bar{\pi}_{\theta}(\cdot|x, y')} [\|\varphi(x, y, y')\|_{\Sigma^{-1}} > \nu] \mathbb{I}\{(x, y') \in \mathcal{Z}_{\text{good}}, x \in \mathcal{X}_{\text{span}}\}] \\ & \quad + \mathbb{E}_{x \sim \rho, y' \sim \pi_{\text{ref}}(\cdot|x)} [\mathbb{I}\{(x, y') \notin \mathcal{Z}_{\text{good}}, x \in \mathcal{X}_{\text{span}}\}] + \mathbb{P}_{x \sim \rho}[x \notin \mathcal{X}_{\text{span}}]. \end{aligned}$$

For the first term, [Lemma G.6](#) implies that when $(x, y') \in \mathcal{Z}_{\text{good}}$, $\bar{\pi}_{\theta}(y \mid x, y') \leq 2e^2 C_{\text{cov}}(\pi_{\theta^*}) \cdot \pi_{\text{ref}}(y \mid x)$ for all $y \in \mathcal{Y}$, so we can bound

$$\begin{aligned} & \mathbb{E}_{x \sim \rho, y' \sim \pi_{\text{ref}}(\cdot|x)} [\mathbb{P}_{y' \sim \bar{\pi}_{\theta}(\cdot|x, y')} [\|\varphi(x, y, y')\|_{\Sigma^{-1}} > \nu] \mathbb{I}\{(x, y') \in \mathcal{Z}_{\text{good}}, x \in \mathcal{X}_{\text{span}}\}] \\ & \leq 2e^2 C_{\text{cov}}(\pi_{\theta^*}) \cdot \mathbb{E}_{x \sim \rho} [\mathbb{P}_{(y,y') \sim \pi_{\text{ref}}(x)} [\|\varphi(x, y, y')\|_{\Sigma^{-1}} > \nu] \mathbb{I}\{x \in \mathcal{X}_{\text{span}}\}] \\ & \leq 2e^2 C_{\text{cov}}(\pi_{\theta^*}) \cdot \varepsilon. \end{aligned}$$

For the second term, we can use Markov's inequality to bound

$$\begin{aligned} \mathbb{E}_{x \sim \rho, y' \sim \pi_{\text{ref}}(\cdot|x)} [\mathbb{I}\{(x, y') \notin \mathcal{Z}_{\text{good}}, x \in \mathcal{X}_{\text{span}}\}] & \leq 2 \mathbb{E}_{x \sim \rho} [\mathbb{P}_{y \sim \pi_{\theta^*}(\cdot|x), y' \sim \pi_{\text{ref}}(\cdot|x)} [\|\varphi(x, y, y')\|_{\Sigma^{-1}} > \nu] \mathbb{I}\{x \in \mathcal{X}_{\text{span}}\}] \\ & \leq 2C_{\text{cov}}(\pi_{\theta^*}) \cdot \mathbb{E}_{x \sim \rho} [\mathbb{P}_{(y,y') \sim \pi_{\text{ref}}(x)} [\|\varphi(x, y, y')\|_{\Sigma^{-1}} > \nu] \mathbb{I}\{x \in \mathcal{X}_{\text{span}}\}] \\ & \leq 2C_{\text{cov}}(\pi_{\theta^*}) \cdot \varepsilon. \end{aligned}$$

Combining the preceding bounds completes the proof. \square

G.4. Proof of Theorem 3.1 (Guarantee for SpannerSampling)

In this section we prove [Theorem 3.1](#), restated below.

Theorem 3.1 (Guarantee for SpannerSampling). *For any $\varepsilon > 0$ and $\delta \in (0, 1)$, by choosing T_{prompt} , T_{span} , T_{exp} appropriately, [Algorithm 3](#) learns a policy with $\mathbb{E}_{\hat{\pi} \sim \text{unif}(\hat{\pi}^1, \dots, \hat{\pi}^{T_{\text{exp}}})} [J_{\beta}(\pi_{\beta}^*) - J_{\beta}(\hat{\pi})] \leq \varepsilon$ with probability at least $1 - \delta$, and achieves the following data efficiency and oracle efficiency bounds:*

$$T_{\text{data}}(\varepsilon, \delta) = \tilde{O}\left(\frac{R_{\text{max}}^2}{\beta}\right) \cdot \frac{d^2 \log^2(\delta^{-1})}{\min\{\varepsilon, \beta\}}, \quad \text{and} \quad T_{\text{comp}}(\varepsilon, \delta) = \tilde{O}\left(C_{\text{cov}}(\pi_{\beta}^*) \cdot \frac{R_{\text{max}}^2}{\beta^2}\right) \cdot T_{\text{data}}(\varepsilon, \delta).$$

The algorithm uses at most $\tilde{O}\left(\frac{R_{\max}^4}{\beta^3}\right) \cdot \frac{d^2 \log^2(\delta^{-1})}{\varepsilon}$ prompts. For any $x \in \mathcal{X}$, one can generate a sample $y \sim \hat{\pi}(\cdot | x)$ from the returned policy using $T_{\text{comp}} = \tilde{O}(C_{\text{cov}}(\pi_\beta^*))$ weak sampling oracle queries.

Note that the high-probability bound is over the randomness of the policies $\hat{\pi}^1, \dots, \hat{\pi}^{T_{\text{exp}}}$, but $\hat{\pi}$ is chosen uniformly from these; a true high-probability bound on $J_\beta(\pi_\beta^*) - J_\beta(\hat{\pi})$ could be obtained by estimating each $J_\beta(\hat{\pi}^t)$ and choosing $\hat{\pi}$ as the minimizer over $t \in [T_{\text{exp}}]$ (as we do in MTSS), but we omit this extra complication here. We begin by proving a number of intermediate results. We then use these results to prove [Theorem 3.1](#) in [Section G.4.2](#).

G.4.1. INTERMEDIATE GUARANTEE FOR SPANNER CONSTRUCTION

In this section we give self-contained guarantees for [Line 5](#) of [Algorithm 3](#), which aims to construct a *spanner*: a collection Ψ_{span} of tuples (x, y, y') for which $\sum_{(x,y,y') \in \Psi_{\text{span}}} \varphi(x, y, y') \varphi(x, y, y')$ covers the feature space as least as well as $(y, y') \sim \pi_{\text{ref}}(\cdot | x)$.

Concretely, let Ψ_{span} denote the collection of all tuples $(x^i, y_1^{t,i}, y_2^{t,i})$ for which the if statement in [Line 9](#) is triggered, so that $\Sigma_{\text{span}} = \lambda I_d + \sum_{(x,y,y') \in \Psi_{\text{span}}} \varphi(x, y, y') \varphi(x, y, y')^\top$ when the outer for loop completes. Our first lemma gives a bound on the size of Ψ_{span} .

Lemma G.8. *Suppose that $\nu, \lambda \leq 1$. With probability 1, we have*

$$|\Psi_{\text{span}}| \lesssim \frac{d \log(1 + \nu^{-1} \lambda^{-1})}{\nu^2}.$$

Proof of Lemma G.8. Order Ψ_{span} as $\Psi_{\text{span}} = \{(x^1, y_1^1, y_2^1) \dots, (x^k, y_1^k, y_2^k)\}$ and let

$$\Gamma_j = \lambda I_d + \sum_{i=1}^j \varphi(x^i, y_1^i, y_2^i) \varphi(x^i, y_1^i, y_2^i)^\top.$$

We will bound $k \in \mathbb{N}$. From the standard elliptic potential lemma argument, we have

$$\log \det \Gamma_k - \log \det \Gamma_0 \geq \sum_{j=1}^k \log \left(1 + \varphi(x^j, y_1^j, y_2^j)^\top (\Gamma_{j-1})^{-1} \varphi(x^j, y_1^j, y_2^j) \right) \geq k \log(1 + \nu^2) \geq \frac{k \nu^2}{2}.$$

Moreover, $\|\varphi\| \leq 2$, $\log \det \Gamma_k \leq d \log(\lambda + 4k/d)$ (e.g., Lemma 10 of Abbasi-Yadkori et al. [1]), whereas $\log \det \Gamma_0 = d \log \lambda$. Hence, we have

$$k \leq \frac{2d \log(1 + 4k/(d\lambda))}{\nu^2},$$

and [Lemma D.6](#) further implies that $k \lesssim \frac{d \log(1 + \nu^{-1} \lambda^{-1})}{\nu^2}$ as claimed. \square

Our second lemma gives a guarantee on the quality of the spanner.

Lemma G.9. *Let $\delta \in (0, 1)$ be fixed, and define*

$$\varepsilon_{\text{span}} := \frac{8 \log(4T_{\text{prompt}} \delta^{-1})}{T_{\text{span}}}.$$

With probability at least $1 - \delta$, [Algorithm 3](#) satisfies

$$\mathbb{P}_{x \sim \rho} \left[\mathbb{P}_{(y, y') \sim \pi_{\text{ref}}(\cdot | x)} \left[\left\| \varphi(x, y, y') \right\|_{\Sigma_{\text{span}}^{-1}} > \nu \right] > \varepsilon_{\text{span}} \right] \lesssim \frac{d \log(1 + \nu^{-1} \lambda^{-1})}{T_{\text{prompt}} \nu^2} + \frac{\log(\delta^{-1})}{T_{\text{prompt}}}.$$

Proof of Lemma G.9. Let Σ_{span}^t and Ψ_{span}^t denote the value of Σ_{span} and Ψ_{span} at the beginning of the iteration t of the for loop in [Line 5](#). For each $t \in [T_{\text{prompt}}]$, let i_t denote the first index i such that the if statement in [Line 9](#) is triggered, and let $i_t = T_{\text{span}}$ otherwise. Using [Lemma D.3](#) and a union bound, we have that with probability at least $1 - \delta/2$, for all $t \in [T_{\text{prompt}}]$,

$$\begin{aligned} \mathbb{P}_{(y, y') \sim \pi_{\text{ref}}(\cdot | x^t)} \left[\left\| \varphi(x, y, y') \right\|_{(\Sigma_{\text{span}}^t)^{-1}} > \nu \right] &\leq \frac{2}{i_t} \sum_{i=1}^{i_t} \mathbb{I} \left\{ \left\| \varphi(x^t, y^{t,i}, y^{t,i}) \right\|_{(\Sigma_{\text{span}}^t)^{-1}} > \nu \right\} + \frac{8 \log(4T_{\text{prompt}} \delta^{-1})}{i_t} \\ &\leq 2 \mathbb{I} \left\{ \exists i : \left\| \varphi(x^t, y^{t,i}, y^{t,i}) \right\|_{(\Sigma_{\text{span}}^t)^{-1}} > \nu \right\} + \frac{8 \log(4T_{\text{prompt}} \delta^{-1})}{i_t}. \end{aligned}$$

If $\left\| \varphi(x^t, y^{t,i}, y^{t,i}) \right\|_{(\Sigma_{\text{span}}^t)^{-1}} \leq \nu$ for all i , then $i_t = T_{\text{span}}$, and consequently the right-hand side above is bounded by $\varepsilon_{\text{span}} := \frac{8 \log(4T_{\text{prompt}} \delta^{-1})}{T_{\text{span}}}$. It follows that under the concentration event above, we have that for all $t \in [T_{\text{prompt}}]$,

$$\mathbb{I} \left\{ \mathbb{P}_{(y, y') \sim \pi_{\text{ref}}(\cdot | x^t)} \left[\left\| \varphi(x, y, y') \right\|_{(\Sigma_{\text{span}}^t)^{-1}} > \nu \right] > \varepsilon_{\text{span}} \right\} \leq \mathbb{I} \left\{ \exists i : \left\| \varphi(x^t, y^{t,i}, y^{t,i}) \right\|_{(\Sigma_{\text{span}}^t)^{-1}} > \nu \right\}.$$

Now, define

$$p^t = \mathbb{P}_{x \sim \rho} \left[\mathbb{P}_{(y, y') \sim \pi_{\text{ref}}(\cdot | x)} \left[\left\| \varphi(x, y, y') \right\|_{(\Sigma_{\text{span}}^t)^{-1}} > \nu \right] > \varepsilon_{\text{span}} \right].$$

Then $p^{T_{\text{prompt}}+1} \leq p^{T_{\text{prompt}}} \leq \dots p^1$, so

$$\mathbb{P}_{x \sim \rho} \left[\mathbb{P}_{(y, y') \sim \pi_{\text{ref}}(\cdot | x)} \left[\left\| \varphi(x, y, y') \right\|_{\Sigma_{\text{span}}^{-1}} > \nu \right] > \varepsilon_{\text{span}} \right] = p^{T_{\text{prompt}}+1} \leq \frac{1}{T_{\text{prompt}}} \sum_{t=1}^{T_{\text{prompt}}} p^t.$$

Since Σ_{span}^t does not depend on x^t , [Lemma D.3](#) implies that with probability at least $1 - \delta/2$,

$$\begin{aligned} \sum_{t=1}^{T_{\text{prompt}}} p^t &\leq 2 \sum_{t=1}^{T_{\text{prompt}}} \mathbb{I} \left\{ \mathbb{P}_{(y, y') \sim \pi_{\text{ref}}(\cdot | x^t)} \left[\left\| \varphi(x, y, y') \right\|_{(\Sigma_{\text{span}}^t)^{-1}} > \nu \right] > \varepsilon_{\text{span}} \right\} + 8 \log(4\delta^{-1}) \\ &\leq 2 \sum_{t=1}^{T_{\text{prompt}}} \mathbb{I} \left\{ \exists i : \left\| \varphi(x^t, y^{t,i}, y^{t,i}) \right\|_{(\Sigma_{\text{span}}^t)^{-1}} > \nu \right\} + 8 \log(4\delta^{-1}) \\ &\leq 2 \left| \Psi_{\text{span}}^{T_{\text{prompt}}+1} \right| + 8 \log(4\delta^{-1}). \end{aligned}$$

From here, the result follows from [Lemma G.8](#). □

G.4.2. PROOF OF THEOREM 3.1

Proof of Theorem 3.1. Recall that we define $\varepsilon_{\text{stat}} := c \cdot \sqrt{dR_{\max}^2 \log(BR_{\max}^{-1}\delta^{-1}T_{\text{exp}})}$ for a sufficiently large absolute constant $c > 0$, and use the parameter settings $\lambda \leftarrow (R_{\max}/B)^2$, $\nu := \beta/\varepsilon_{\text{stat}}$, $M_{\text{rej}} := 8e^2 \cdot C_{\text{cov}}(\pi_{\theta^*})$, and $\delta_{\text{rej}} := T_{\text{exp}}^{-1}$. We will show that under these settings, for any choice of T_{prompt} , T_{span} , and T_{exp} , we have that with probability at least $1 - \delta$,

$$\mathbb{E}_{t \sim \text{unif}([T_{\text{exp}}])} [J_{\beta}(\pi_{\theta^*}) - J_{\beta}(\hat{\pi}^t)] \lesssim \frac{R_{\max}^2}{\beta} \cdot \tilde{O} \left(\frac{d^2 \log^2(\delta^{-1})}{T_{\text{exp}}} + \frac{d^2 R_{\max}^2 \log^2(\delta^{-1})}{\beta^2 T_{\text{prompt}}} + \frac{C_{\text{cov}}(\pi_{\theta^*}) \cdot \log(\delta^{-1})}{T_{\text{span}}} \right).$$

We will use this to give bounds on $T_{\text{data}}(\varepsilon, \delta)$ and $T_{\text{comp}}(\varepsilon, \delta)$ at the end of the proof.

Preliminaries: Least squares. We begin with some preliminary observations. First, for each $t \in [T_{\text{exp}}]$, define

$$\Sigma_{\text{full}}^t = \lambda I_d + \sum_{(x, y_1, y_2) \in \Psi_{\text{span}}} \varphi(x^t, y_1, y_2) \varphi(x^t, y_1, y_2)^\top + \sum_{i < t} \varphi(x^t, y_1^i, y_2^i) \varphi(x^t, y_1^i, y_2^i)^\top$$

and $\Sigma_{\text{exp}}^t = \lambda I_d + \sum_{i < t} \varphi(x^t, y_1^i, y_2^i) \varphi(x^t, y_1^i, y_2^i)^\top$.

We invoke [Lemma G.3](#), which implies that for the choice of λ in [Line 2](#), we are guaranteed that with probability at least $1 - \delta/3$, for all $t \in [T_{\text{exp}}]$,

$$\|\theta^t - \theta^*\|_{\Sigma_{\text{full}}^t}^2 \leq \underbrace{c \cdot dR_{\max}^2 \log(BR_{\max}^{-1}\delta^{-1}T_{\text{exp}})}_{=:\varepsilon_{\text{stat}}^2}. \quad (58)$$

for an absolute constant $c > 0$. We denote this event by $\mathcal{E}_{\text{conc}}$ and condition on it going forward. In particular, under this event, we have

$$\|\theta^t - \theta^*\|_{\Sigma_{\text{span}}}^2 \leq \varepsilon_{\text{stat}}^2, \quad \text{and} \quad \|\theta^t - \theta^*\|_{\Sigma_{\text{exp}}^t}^2 \leq \varepsilon_{\text{stat}}^2. \quad (59)$$

Preliminaries: Truncated policies. Next, recall that we define

$$r^t(x, y, y') := \langle \theta^t, \varphi(x, y, y') \rangle \mathbb{I} \{ \|\varphi(x, y, y')\|_{\Sigma_{\text{span}}^{-1}} \leq \nu \} = \langle \theta^t, \bar{\varphi}(x, y, y') \rangle$$

in [Line 17](#), where $\bar{\varphi}(x, y, y') := \varphi(x, y, y') \mathbb{I} \{ \|\varphi(x, y, y')\|_{\Sigma_{\text{span}}^{-1}} \leq \nu \}$. It will be helpful to define some intermediate policies. First, define

$$\pi^t(y \mid x, y') \propto \pi_{\text{ref}}(y \mid x) \exp(\beta^{-1} r^t(x, y, y')) = \pi_{\text{ref}}(y \mid x) \exp(\beta^{-1} \langle \theta^t, \bar{\varphi}(x, y, y') \rangle)$$

be the softmax policy induced by $r^t(\cdot, \cdot, y')$. Clearly, we have

$$\pi^t(y \mid x, y') = \bar{\pi}_{\theta^t}(y \mid x, y'),$$

where $\bar{\pi}_{\theta}(y \mid x, y')$ is the truncated softmax policy defined in [Appendix G.3](#) for parameters Σ_{span} and ν . We further define

$$\pi^t(y, y' \mid x) := \bar{\pi}_{\theta^t}(y, y' \mid x) = \bar{\pi}_{\theta^t}(y \mid x, y') \cdot \pi_{\text{ref}}(y' \mid x)$$

as the joint distribution over (y, y') induced by sampling $y' \sim \pi_{\text{ref}}(\cdot \mid x)$ and $y \sim \pi^t(\cdot \mid x, y')$, and define $\pi^t(y \mid x) = \bar{\pi}_{\theta^t}(y \mid x) := \mathbb{E}_{y' \sim \pi_{\text{ref}}(\cdot \mid x)} [\pi^t(y \mid x, y')]$ as the induced “marginal” policy over y .

Note that by definition of ν , whenever $\mathcal{E}_{\text{conc}}$ holds, we have

$$\begin{aligned} |r^t(x, y, y')| &\leq \|\theta^t - \theta^*\|_{\Sigma_{\text{span}}} \|\bar{\varphi}(x, y, y')\|_{\Sigma_{\text{span}}^{-1}} + |\langle \theta^*, \varphi(x, y, y') \rangle| \\ &\leq \nu \varepsilon_{\text{stat}} + R_{\max} \leq \beta + R_{\max} \leq 2R_{\max}. \end{aligned} \quad (60)$$

Preliminaries: Spanner construction. Define

$$\varepsilon_{\text{span}} := \frac{8 \log(12T_{\text{prompt}}\delta^{-1})}{T_{\text{span}}}.$$

[Lemma G.9](#) implies that with probability at least $1 - \delta/3$,

$$\mathbb{P}_{x \sim \rho} \left[\mathbb{P}_{(y, y') \sim \pi_{\text{ref}}(\cdot | x)} \left[\left\| \varphi(x, y, y') \right\|_{\Sigma_{\text{span}}^{-1}} > \nu \right] > \varepsilon_{\text{span}} \right] \leq \varepsilon_{\text{prompt}} \quad (61)$$

for

$$\varepsilon_{\text{prompt}} \lesssim \frac{d \log(1 + \nu^{-1} \lambda^{-1})}{T_{\text{prompt}} \nu^2} + \frac{\log(\delta^{-1})}{T_{\text{prompt}}}.$$

We denote this event by $\mathcal{E}_{\text{span}}$ and condition on it going forward. It will be convenient to define

$$\mathcal{X}_{\text{span}} := \left\{ x \in \mathcal{X} \mid \mathbb{P}_{(y, y') \sim \pi_{\text{ref}}(\cdot | x)} \left[\left\| \varphi(x, y, y') \right\|_{\Sigma_{\text{span}}^{-1}} > \nu \right] \leq \varepsilon_{\text{span}} \right\}$$

so that [Eq. \(61\)](#) can be equivalently written as $\mathbb{P}_{x \sim \rho} [x \notin \mathcal{X}_{\text{span}}] \leq \varepsilon_{\text{prompt}}$ under this event.

Preliminaries: Rejection sampling. We define

$$\hat{\pi}^t(\cdot \mid x, y') := \text{SoftmaxSampler}_{\beta, M_{\text{rej}}, \delta_{\text{rej}}}(r^t(\cdot, \cdot, y'); x, \pi_{\text{ref}})$$

denote the distribution over y'_1 in [Line 19](#) (given $x^t = x$ and $y_2^t = y'$), which aims to approximate $\bar{\pi}_{\theta^t}(\cdot \mid x, y')$, and define

$$\hat{\pi}^t(y, y' \mid x)$$

as the law of $y' \sim \pi_{\text{ref}}(\cdot \mid x)$ and $y \sim \text{SoftmaxSampler}_{\beta, M_{\text{rej}}, \delta_{\text{rej}}}(\bar{r}^t(\cdot, \cdot, y'); x, \pi_{\text{ref}})$, using $\hat{\pi}^t(y \mid x) = \mathbb{E}_{y' \sim \pi_{\text{ref}}(\cdot | x)} [\hat{\pi}^t(y \mid x, y')]$ to denote the marginal.

Define

$$\mathcal{Z}_{\text{good}} := \left\{ (x, y') \in \mathcal{X} \times \mathcal{Y} \mid \mathbb{P}_{y \sim \pi_{\theta^*}(\cdot | x)} \left[\left\| \varphi(x, y, y') \right\|_{\Sigma_{\text{span}}^{-1}} > \nu \right] \leq \frac{1}{2} \right\}.$$

By [Lemma G.6](#), we have that under $\mathcal{E}_{\text{conc}}$,

$$C_{\text{cov}}(\bar{\pi}_{\theta^t}(\cdot \mid x, y')) \leq 2e^2 C_{\text{cov}}(\pi_{\theta^*}) \quad \text{for all } (x, y') \in \mathcal{Z}_{\text{good}}, \quad (62)$$

so [Theorem F.2](#) implies that for the choice for M_{rej} in [Line 3](#), we have

$$D_{\text{TV}}(\hat{\pi}^t(\cdot \mid x, y'), \pi^t(\cdot \mid x, y')) \leq \delta_{\text{rej}} \quad (63)$$

for all $(x, y') \in \mathcal{Z}_{\text{good}}$. We can further derive the following consequence.

Lemma G.10. *Under the event $\mathcal{E}_{\text{conc}}$, for any function $f(x, y, y') \in [0, 1]$,*

$$\begin{aligned} & \left| \mathbb{E}_{x \sim \rho, (y, y') \sim \pi^t} [f(x, y, y')] - \mathbb{E}_{x \sim \rho, (y, y') \sim \hat{\pi}^t} [f(x, y, y')] \right| \\ & \leq \delta_{\text{rej}} + 2C_{\text{cov}}(\pi_{\theta^*}) \cdot \varepsilon_{\text{span}} + \varepsilon_{\text{prompt}}. \end{aligned}$$

Proof of Lemma G.10. By Eq. (63), we can bound

$$\begin{aligned}
& \left| \mathbb{E}_{x \sim \rho, (y, y') \sim \pi^t} [f(x, y, y')] - \mathbb{E}_{x \sim \rho, (y, y') \sim \hat{\pi}^t} [f(x, y, y')] \right| \\
& \leq \mathbb{E}_{x \sim \rho, y' \sim \pi_{\text{ref}}(\cdot | x)} \left[\left| \mathbb{E}_{y \sim \pi^t(\cdot | x, y')} [f(x, y, y')] - \mathbb{E}_{y \sim \hat{\pi}^t(\cdot | x, y')} [f(x, y, y')] \right| \mathbb{I}\{(x, y') \in \mathcal{Z}_{\text{good}}, x \in \mathcal{X}_{\text{span}}\} \right] \\
& \quad + \mathbb{P}_{x \sim \rho, y' \sim \pi_{\text{ref}}(\cdot | x)} [\mathbb{I}\{(x, y') \notin \mathcal{Z}_{\text{good}}\}] + \mathbb{P}_{x \sim \rho} [x \notin \mathcal{X}_{\text{span}}] \\
& \leq \delta_{\text{rej}} + \mathbb{E}_{x \sim \rho, y' \sim \pi_{\text{ref}}(\cdot | x)} [\mathbb{I}\{(x, y') \notin \mathcal{Z}_{\text{good}}, x \in \mathcal{X}_{\text{span}}\}] + \varepsilon_{\text{prompt}}.
\end{aligned}$$

Using Markov's inequality, we can further bound

$$\begin{aligned}
& \mathbb{E}_{x \sim \rho, y' \sim \pi_{\text{ref}}(\cdot | x)} [\mathbb{I}\{(x, y') \notin \mathcal{Z}_{\text{good}}, x \in \mathcal{X}_{\text{span}}\}] \\
& \leq 2 \mathbb{E}_{x \sim \rho} \left[\mathbb{P}_{y' \sim \pi_{\text{ref}}(\cdot | x), y \sim \pi_{\theta^*}(x)} \left[\left\| \varphi(x, y, y') \right\|_{\Sigma_{\text{span}}^{-1}} > \nu \right] \mathbb{I}\{x \in \mathcal{X}_{\text{span}}\} \right] \\
& \leq 2C_{\text{cov}}(\pi_{\theta^*}) \mathbb{E}_{x \sim \rho} \left[\mathbb{P}_{(y, y') \sim \pi_{\text{ref}}(\cdot | x)} \left[\left\| \varphi(x, y, y') \right\|_{\Sigma_{\text{span}}^{-1}} > \nu \right] \mathbb{I}\{x \in \mathcal{X}_{\text{span}}\} \right] \\
& \leq 2C_{\text{cov}}(\pi_{\theta^*}) \cdot \varepsilon_{\text{span}},
\end{aligned}$$

where the final inequality follows from the definition of $\mathcal{X}_{\text{span}}$. \square

Moving to idealized softmax policies. Our aim is to bound the regret

$$\mathbb{E}_{t \sim \text{unif}([T_{\text{exp}}])} [J_{\beta}(\pi_{\theta^*}) - J_{\beta}(\hat{\pi}^t)] = \frac{1}{T_{\text{exp}}} \sum_{t=1}^{T_{\text{exp}}} J_{\beta}(\pi_{\theta^*}) - J_{\beta}(\hat{\pi}^t).$$

Define

$$\bar{J}_{\beta}(\bar{\pi}_{\theta}) = \mathbb{E}_{x \sim \rho, y' \sim \pi_{\text{ref}}(\cdot | x), y \sim \bar{\pi}_{\theta}(\cdot | x, y')} [r^*(x, y) - \beta D_{\text{KL}}(\bar{\pi}_{\theta}(\cdot | x, y') \parallel \pi_{\text{ref}}(\cdot | x))].$$

We invoke Lemma G.11 below (proven in the sequel) to bound

$$\frac{1}{T_{\text{exp}}} \sum_{t=1}^{T_{\text{exp}}} J_{\beta}(\pi_{\theta^*}) - J_{\beta}(\hat{\pi}^t) \lesssim \frac{1}{T_{\text{exp}}} \sum_{t=1}^{T_{\text{exp}}} J_{\beta}(\pi_{\theta^*}) - \bar{J}_{\beta}(\bar{\pi}_{\theta^t}) + R_{\text{max}} \log \log(T_{\text{exp}}) \cdot (\delta_{\text{rej}} + C_{\text{cov}}(\pi_{\theta^*}) \cdot \varepsilon_{\text{span}} + \varepsilon_{\text{prompt}}). \quad (64)$$

Lemma G.11. Under the event $\mathcal{E}_{\text{conc}}$, for any $\delta_{\text{rej}} \in (0, 1)$, we have

$$\bar{J}_{\beta}(\bar{\pi}_{\theta^t}) - J_{\beta}(\hat{\pi}^t) \lesssim O(R_{\text{max}} \log \log(\delta_{\text{rej}}^{-1}) \cdot (\delta_{\text{rej}} + C_{\text{cov}}(\pi_{\theta^*}) \cdot \varepsilon_{\text{span}} + \varepsilon_{\text{prompt}})).$$

Regret bound for truncated softmax policy. For the next step, we note that for the choice of $\nu = \beta/\varepsilon_{\text{stat}}$, under $\mathcal{E}_{\text{conc}}$, our central regret decomposition for truncated softmax policies (Lemma G.7) implies that

$$\begin{aligned}
& \frac{1}{T_{\text{exp}}} \sum_{t=1}^{T_{\text{exp}}} J_{\beta}(\pi_{\theta^*}) - \bar{J}_{\beta}(\bar{\pi}_{\theta^t}) \\
& \leq \frac{1}{\beta T_{\text{exp}}} \sum_{t=1}^{T_{\text{exp}}} \mathbb{E}_{x \sim \rho, (y, y') \sim \bar{\pi}_{\theta^t}(x)} \left[\langle \theta^t - \theta^*, \bar{\varphi}(x, y, y') \rangle^2 \right] + O(R_{\text{max}}) \cdot (C_{\text{cov}}(\pi_{\theta^*}) \cdot \varepsilon_{\text{span}} + \varepsilon_{\text{prompt}}).
\end{aligned}$$

Using Eq. (59) and Eq. (60), we can bound

$$\begin{aligned}
 & \sum_{t=1}^{T_{\text{exp}}} \mathbb{E}_{x \sim \rho, (y, y') \sim \bar{\pi}_{\theta^t}(x)} \left[\langle \theta^t - \theta^*, \bar{\varphi}(x, y, y') \rangle^2 \right] \\
 & \leq 4 \sum_{t=1}^{T_{\text{exp}}} \mathbb{E}_{x \sim \rho, (y, y') \sim \bar{\pi}_{\theta^t}(x)} \left[\langle \theta^t - \theta^*, \bar{\varphi}(x, y, y') \rangle^2 \wedge R_{\max}^2 \right] \\
 & \leq 4 \sum_{t=1}^{T_{\text{exp}}} \mathbb{E}_{x \sim \rho, (y, y') \sim \bar{\pi}_{\theta^t}(x)} \left[\varepsilon_{\text{stat}}^2 \|\varphi(x, y, y')\|_{(\Sigma_{\text{exp}}^t)^{-1}}^2 \wedge R_{\max}^2 \right].
 \end{aligned}$$

We can further use Lemma G.10 to bound

$$\begin{aligned}
 & \sum_{t=1}^{T_{\text{exp}}} \mathbb{E}_{x \sim \rho, (y, y') \sim \bar{\pi}_{\theta^t}(x)} \left[\varepsilon_{\text{stat}}^2 \|\varphi(x, y, y')\|_{(\Sigma_{\text{exp}}^t)^{-1}}^2 \wedge R_{\max}^2 \right] \\
 & \leq \sum_{t=1}^{T_{\text{exp}}} \mathbb{E}_{x \sim \rho, (y, y') \sim \hat{\pi}^t(x)} \left[\varepsilon_{\text{stat}}^2 \|\varphi(x, y, y')\|_{(\Sigma_{\text{exp}}^t)^{-1}}^2 \wedge R_{\max}^2 \right] + O(R_{\max}^2 T_{\text{exp}} (\delta_{\text{rej}} + C_{\text{cov}}(\pi_{\theta^*}) \cdot \varepsilon_{\text{span}} + \varepsilon_{\text{prompt}})) \\
 & \leq \varepsilon_{\text{stat}}^2 \sum_{t=1}^{T_{\text{exp}}} \mathbb{E}_{x \sim \rho, (y, y') \sim \hat{\pi}^t(x)} \left[\|\varphi(x, y, y')\|_{(\Sigma_{\text{exp}}^t)^{-1}}^2 \wedge 1 \right] + O(R_{\max}^2 T_{\text{exp}} (\delta_{\text{rej}} + C_{\text{cov}}(\pi_{\theta^*}) \cdot \varepsilon_{\text{span}} + \varepsilon_{\text{prompt}})),
 \end{aligned}$$

where the last inequality uses that $\varepsilon_{\text{stat}} \geq R_{\max}$. Now, by Lemma D.3, we are guaranteed that with probability at least $1 - \delta/3$,

$$\sum_{t=1}^{T_{\text{exp}}} \mathbb{E}_{x \sim \rho, (y, y') \sim \hat{\pi}^t(x)} \left[\min \left\{ \|\varphi(x, y, y')\|_{(\Sigma_{\text{exp}}^t)^{-1}}^2, 1 \right\} \right] \leq \frac{3}{2} \sum_{t=1}^{T_{\text{exp}}} \min \left\{ \|\varphi(x^t, y_1^t, y_2^t)\|_{(\Sigma_{\text{exp}}^t)^{-1}}^2, 1 \right\} + 4 \log(6\delta^{-1}).$$

Finally, since $\Sigma_{\text{exp}}^t = \lambda I_d + \sum_{i < t} \varphi(x^i, y_1^i, y_2^i) \varphi(x^i, y_1^i, y_2^i)^\top$, Lemma D.4 implies that

$$\sum_{t=1}^{T_{\text{exp}}} \min \left\{ \|\varphi(x^t, y_1^t, y_2^t)\|_{(\Sigma_{\text{exp}}^t)^{-1}}^2, 1 \right\} \leq 2d \log(1 + \lambda^{-1} T_{\text{exp}}/d).$$

Putting everything together: Final bounds on T_{data} and T_{comp} . Combining all of the preceding inequalities and simplifying (using that $\varepsilon_{\text{stat}} \geq R_{\max} \geq \beta$ and $\delta_{\text{rej}} = T_{\text{exp}}^{-1}$), we conclude that with probability at least $1 - \delta$,

$$\begin{aligned}
 & \mathbb{E}_{t \sim \text{unif}([T_{\text{exp}}])} [J_{\beta}(\pi_{\theta^*}) - J_{\beta}(\hat{\pi}^t)] \\
 & \lesssim \frac{\varepsilon_{\text{stat}}^2 \cdot d \log(\lambda^{-1} T_{\text{exp}} \delta^{-1})}{\beta T_{\text{exp}}} + \frac{R_{\max}^2 \log \log(T_{\text{exp}})}{\beta} \cdot \left(\frac{1}{T_{\text{exp}}} + C_{\text{cov}}(\pi_{\theta^*}) \cdot \varepsilon_{\text{span}} + \varepsilon_{\text{prompt}} \right) \\
 & \lesssim \frac{\varepsilon_{\text{stat}}^2 \cdot d \log(B R_{\max}^{-1} T_{\text{exp}} \delta^{-1})}{\beta T_{\text{exp}}} + \frac{d R_{\max}^2 \log(B R_{\max}^{-1} \nu^{-1} \delta^{-1}) \log \log(T_{\text{exp}})}{\beta \nu^2 T_{\text{prompt}}} \\
 & \quad + \frac{R_{\max}^2 \log(\delta^{-1}) \log \log(T_{\text{exp}}) \cdot C_{\text{cov}}(\pi_{\theta^*})}{\beta T_{\text{span}}}
 \end{aligned}$$

$$\begin{aligned} &\lesssim \frac{\varepsilon_{\text{stat}}^2 \cdot d \log(BR_{\max}^{-1} T_{\text{exp}} \delta^{-1})}{\beta T_{\text{exp}}} + \frac{\varepsilon_{\text{stat}}^2 \cdot d R_{\max}^2 \log(BR_{\max}^{-1} \nu^{-1} \delta^{-1}) \log \log(T_{\text{exp}})}{\beta^3 T_{\text{prompt}}} \\ &\quad + \frac{R_{\max}^2 \log(\delta^{-1}) \log \log(T_{\text{exp}}) \cdot C_{\text{cov}}(\pi_{\theta^*})}{\beta T_{\text{span}}}, \end{aligned}$$

where the second inequality uses [Eq. \(61\)](#) and the third inequality uses that $\nu := \beta/\varepsilon_{\text{stat}}$. Choosing

$$T_{\text{prompt}} = \tilde{\Theta}\left(\frac{R_{\max}^2}{\beta^2} \cdot T_{\text{exp}}\right), \quad \text{and} \quad T_{\text{span}} = \tilde{\Theta}(C_{\text{cov}}(\pi_{\theta^*}) \cdot T_{\text{exp}})$$

suffices to give

$$\mathbb{E}_{t \sim \text{unif}([T_{\text{exp}}])} [J_{\beta}(\pi_{\theta^*}) - J_{\beta}(\hat{\pi}^t)] \leq \tilde{O}\left(\frac{\varepsilon_{\text{stat}}^2 \cdot d \log(\delta^{-1})}{\beta T_{\text{exp}}}\right) = \tilde{O}\left(\frac{d^2 R_{\max}^2 \log^2(\delta^{-1})}{\beta T_{\text{exp}}}\right).$$

so that setting suffices to achieve $\mathbb{E}_{t \sim \text{unif}([T_{\text{exp}}])} [J_{\beta}(\pi_{\theta^*}) - J_{\beta}(\hat{\pi}^t)] \leq \varepsilon$. We now bound the number of reward/prompt queries and sampling oracle queries. First, note that during the spanner construction phase, the algorithm queries the reward oracle twice whenever it expands Ψ_{span} , and does not query it otherwise. Meanwhile, it queries the reward oracle twice at each round of the exploration phase. Consequently, by [Lemma G.8](#), we have

$$T_{\text{data}}(\varepsilon, \delta) \leq 2(|\Psi_{\text{span}}| + T_{\text{exp}}) \leq \tilde{O}\left(\frac{d}{\nu^2} + T_{\text{exp}}\right) \leq \tilde{O}\left(\frac{d^2 R_{\max}^2 \log(\delta^{-1})}{\beta^2} + \frac{d^2 R_{\max}^2 \log^2(\delta^{-1})}{\beta \varepsilon}\right).$$

where we have used that $\nu = \beta/\varepsilon_{\text{stat}}$. We also observe that the number of prompts used by the algorithm is

$$T_{\text{prompt}} + T_{\text{exp}} = \tilde{O}\left(\frac{R_{\max}^2}{\beta^2} \cdot T_{\text{exp}} + T_{\text{exp}}\right) = \tilde{O}\left(\frac{d^2 R_{\max}^4 \log^2(\delta^{-1})}{\beta^3 \varepsilon}\right).$$

To bound the number of sampling oracle queries, we note that the algorithm queries the sampling oracle twice during each inner loop iteration of the spanner construction phase, and calls it $O(M_{\text{rej}} \log(\delta_{\text{rej}}^{-1})) = \tilde{O}(C_{\text{cov}}(\pi_{\theta^*}))$ times during each round of the exploration phase (through the invocation of `SoftmaxSampler` ([Algorithm 1](#))). We can thus bound

$$T_{\text{comp}}(\varepsilon, \delta) \leq \tilde{O}(T_{\text{prompt}} T_{\text{span}} + C_{\text{cov}}(\pi_{\theta^*}) \cdot T_{\text{exp}}) \leq \tilde{O}\left(C_{\text{cov}}(\pi_{\theta^*}) \cdot \frac{R_{\max}^2}{\beta^2} \cdot T_{\text{exp}}^2\right).$$

□

G.4.3. PROOFS FOR SUPPORTING LEMMAS

Proof of [Lemma G.11](#). We begin by writing

$$\begin{aligned} \bar{J}_{\beta}(\bar{\pi}_{\theta^t}) - J_{\beta}(\hat{\pi}^t) &= \mathbb{E}_{x \sim \rho, y' \sim \pi_{\text{ref}}(\cdot | x), y \sim \bar{\pi}_{\theta^t}(\cdot | x, y')} [r^*(x, y) - \beta D_{\text{KL}}(\bar{\pi}_{\theta^t}(\cdot | x, y') \| \pi_{\text{ref}}(\cdot | x))] \\ &\quad - \mathbb{E}_{x \sim \rho, y' \sim \pi_{\text{ref}}(\cdot | x), y \sim \hat{\pi}^t(\cdot | x, y')} [r^*(x, y)] + \beta \mathbb{E}_{x \sim \rho} [D_{\text{KL}}(\hat{\pi}^t(\cdot | x) \| \pi_{\text{ref}}(\cdot | x))] \end{aligned}$$

$$\begin{aligned} &\leq \mathbb{E}_{x \sim \rho, y' \sim \pi_{\text{ref}}(\cdot | x), y \sim \bar{\pi}_{\theta^t}(\cdot | x, y')} [r^*(x, y) - \beta D_{\text{KL}}(\bar{\pi}_{\theta^t}(\cdot | x, y') \| \pi_{\text{ref}}(\cdot | x))] \\ &\quad - \mathbb{E}_{x \sim \rho, y' \sim \pi_{\text{ref}}(\cdot | x), y \sim \hat{\pi}^t(\cdot | x, y')} [r^*(x, y) - \beta D_{\text{KL}}(\hat{\pi}^t(\cdot | x, y') \| \pi_{\text{ref}}(\cdot | x))], \end{aligned}$$

where the inequality uses convexity of KL-divergence. We can further bound this by

$$\begin{aligned} &\underbrace{\mathbb{E}_{x \sim \rho, y' \sim \pi_{\text{ref}}(\cdot | x)} \left[\left(\mathbb{E}_{y \sim \bar{\pi}_{\theta^t}(\cdot | x, y')} [r^*(x, y)] - \mathbb{E}_{y \sim \hat{\pi}^t(\cdot | x, y')} [r^*(x, y)] \right) \mathbb{I}\{(x, y') \in \mathcal{Z}_{\text{good}}\} \right]}_{\text{I}} \\ &+ \underbrace{\beta \mathbb{E}_{x \sim \rho, y' \sim \pi_{\text{ref}}(\cdot | x)} \left[\left(D_{\text{KL}}(\hat{\pi}^t(\cdot | x, y') \| \pi_{\text{ref}}(\cdot | x)) - D_{\text{KL}}(\bar{\pi}_{\theta^t}(\cdot | x, y') \| \pi_{\text{ref}}(\cdot | x)) \right) \mathbb{I}\{(x, y') \in \mathcal{Z}_{\text{good}}\} \right]}_{\text{II}} \\ &+ \underbrace{\mathbb{E}_{x \sim \rho, y' \sim \pi_{\text{ref}}(\cdot | x)} \left[\left(R_{\text{max}} + \beta D_{\text{KL}}(\hat{\pi}^t(\cdot | x, y') \| \pi_{\text{ref}}(\cdot | x)) \right) \mathbb{I}\{(x, y') \notin \mathcal{Z}_{\text{good}}\} \right]}_{\text{III}}. \end{aligned}$$

For the first two terms above, our choice for M_{rej} and [Eq. \(62\)](#) imply that whenever $(x, y') \in \mathcal{Z}_{\text{good}}$, the conditions of [Lemma F.1](#) apply, so we have

$$\text{I} \leq 2R_{\text{max}}\delta_{\text{rej}}, \quad \text{and} \quad \text{II} \leq \beta \cdot O\left(\frac{R_{\text{max}}}{\beta} + \log(C_{\text{cov}}(\pi_{\theta^*}) \log(\delta_{\text{rej}}^{-1}))\right) \cdot \delta_{\text{rej}} \leq O(R_{\text{max}}\delta_{\text{rej}} \log \log(\delta_{\text{rej}}^{-1}))$$

as long as $\beta \leq R_{\text{max}}$. Meanwhile, [Lemma F.1](#) also implies that for all (x, y') ,

$$\beta D_{\text{KL}}(\hat{\pi}^t(\cdot | x, y') \| \pi_{\text{ref}}(\cdot | x)) \leq O(R_{\text{max}} + \beta \log(C_{\text{cov}}(\pi_{\theta^*}) \log(\delta_{\text{rej}}^{-1}))) \leq O(R_{\text{max}} \log \log(\delta_{\text{rej}}^{-1})),$$

and hence

$$\text{III} \leq O(R_{\text{max}} \log \log(\delta_{\text{rej}}^{-1})) \cdot \mathbb{E}_{x \sim \rho, y' \sim \pi_{\text{ref}}(\cdot | x)} [\mathbb{I}\{(x, y') \notin \mathcal{Z}_{\text{good}}\}].$$

To conclude, we use the definition of $\mathcal{X}_{\text{span}}$ to bound

$$\begin{aligned} &\mathbb{E}_{x \sim \rho, y' \sim \pi_{\text{ref}}(\cdot | x)} [\mathbb{I}\{(x, y') \notin \mathcal{Z}_{\text{good}}\}] \\ &\leq \mathbb{E}_{x \sim \rho, y' \sim \pi_{\text{ref}}(\cdot | x)} [\mathbb{I}\{(x, y') \notin \mathcal{Z}_{\text{good}}, x \in \mathcal{X}_{\text{span}}\}] + \varepsilon_{\text{prompt}} \\ &\leq 2 \mathbb{E}_{x \sim \rho} \left[\mathbb{P}_{y' \sim \pi_{\text{ref}}(\cdot | x), y \sim \pi_{\theta^*}(\cdot | x)} \left[\|\varphi(x, y, y')\|_{\Sigma_{\text{span}}^{-1}} > \nu \right] \mathbb{I}\{x \in \mathcal{X}_{\text{span}}\} \right] + \varepsilon_{\text{prompt}} \\ &\leq 2C_{\text{cov}}(\pi_{\theta^*}) \mathbb{E}_{x \sim \rho} \left[\mathbb{P}_{(y, y') \sim \pi_{\text{ref}}(\cdot | x)} \left[\|\varphi(x, y, y')\|_{\Sigma_{\text{span}}^{-1}} > \nu \right] \mathbb{I}\{x \in \mathcal{X}_{\text{span}}\} \right] + \varepsilon_{\text{prompt}} \\ &\leq 2C_{\text{cov}}(\pi_{\theta^*}) \cdot \varepsilon_{\text{span}} + \varepsilon_{\text{prompt}}. \end{aligned}$$

where the second inequality above is Markov's inequality. □

Appendix H. Proofs from Section 4

In this section we prove [Theorem 4.1](#); we formally state the Randomized Exponential Time Hypothesis and restate the theorem below.

Conjecture H.1 (Randomized Exponential Time Hypothesis [11]). *There is no randomized algorithm with time complexity $2^{o(n)}$ that, given a 3-SAT formula φ with n clauses, has the following guarantee:*

- *If φ is satisfiable, then the output is YES with probability at least $1/2$.*
- *If φ is unsatisfiable, then the output is NO.*

Theorem 4.1 (Proper algorithms cannot be computationally efficient). *Under the Randomized ETH ([Conjecture H.1](#)), there is no proper alignment algorithm, even with a strong oracle, that (i) has $T_{\text{data}}(\varepsilon, \delta) \leq \text{poly}(d, \beta^{-1}, \varepsilon^{-1}, \delta^{-1})$ and $T_{\text{comp}}(\varepsilon, \delta) \leq \text{poly}(d, \exp(\beta^{-1}), \varepsilon^{-1}, \delta^{-1})$ under [Assumption 2.1](#) (with $R_{\max} = 1$, $B = \sqrt{d}$),¹⁶ and (ii) has runtime $\text{poly}(d, \exp(\beta^{-1}), \varepsilon^{-1}, \delta^{-1})$.*

Recall the definition of a proper alignment algorithm from [Definition 4.1](#). We note in passing that our proof shows that [Theorem 4.1](#) holds in a stronger sampling oracle model where the algorithm directly observes the log-probability $\log \pi_{\theta}(y \mid x)$ for each response sampled from the oracle.

Preliminaries. We say that Alg is an online alignment algorithm for linear softmax policies with parameter set Θ if, for any given $d \in \mathbb{N}$, $\beta > 0$, and $\varepsilon, \delta > 0$, Alg solves any d -dimensional instance with regularization parameter β , regret ε , and failure probability δ . In order to be explicit, we write $T_{\text{data}}(d, \beta, \varepsilon, \delta)$ to denote the number of reward oracle queries used by Alg, and $T_{\text{comp}}(d, \beta, \varepsilon, \delta)$ to denote the number of strong sampling oracle queries.

H.1. Overview of Proof

Note that [Theorem 4.1](#) does not require the *output* of the alignment algorithm to itself be proper, i.e. lie in the policy class Π ; per the definition of a proper alignment algorithm ([Definition 4.1](#)), it only requires the exploratory policies to be proper. To prove [Theorem 4.1](#), the primary building block is the following weaker result, [Theorem H.1](#), which gives hardness under the additional assumption that the output policy is required to lie in Π . We deduce [Theorem 4.1](#) from this result by showing that one can use imitation learning to efficiently convert any improper output policy into a proper one ([Lemma H.4](#)); this leverages the fact that behavior cloning with the log-loss is computationally efficient for linearly parametrized softmax policies [72].

Theorem H.1. *Under the Randomized Exponential Time Hypothesis ([Conjecture H.1](#)), there is no proper alignment algorithm, even with a strong oracle and a Euclidean projection oracle for Θ , that (i) has $T_{\text{data}}(\varepsilon, \delta) \leq \text{poly}\left(d, \beta^{-1}, \varepsilon^{-1}, \delta^{-1}, \log \frac{1}{\min_{x,y} \pi_{\text{ref}}(y|x)}\right)$ and*

$$T_{\text{comp}}(\varepsilon, \delta) \leq \text{poly}\left(d, \exp(\beta^{-1}), \varepsilon^{-1}, \delta^{-1}, |\mathcal{Y}|, \log \frac{1}{\min_{x,y} \pi_{\text{ref}}(y|x)}\right)$$

under [Assumption 2.1](#) (with $R_{\max} = 1$, $B = \sqrt{d}$), (ii) has runtime $\text{poly}\left(d, \exp(\beta^{-1}), \varepsilon^{-1}, \delta^{-1}, |\mathcal{Y}|, \log \frac{1}{\min_{x,y} \pi_{\text{ref}}(y|x)}\right)$, and (iii) has output $\hat{\pi} \in \Pi$.

16. Concretely, we use the parameter set $\Theta = \{\theta \in \mathbb{R}^d \mid \|\theta\|_{\infty} \leq 1\}$.

We prove this hardness result in the simpler fixed-prompt setting (i.e. $\mathcal{X} = \{\perp\}$). For notational convenience, we henceforth omit all dependencies on \perp , i.e. we write $\pi(y) := \pi(y \mid \perp)$ and $r^*(y) := r^*(\perp, y)$ for any response $y \in \mathcal{Y}$. We prove the result for parameter set $\Theta := \{\theta \in \mathbb{R}^d : \|\theta\|_\infty \leq 1\}$, which is indeed contained in the Euclidean ball of radius $B = \sqrt{d}$, and admits an efficient Euclidean projection oracle. The proof is based on a reduction from the NP-hard *Max-k-DNF* problem.

Definition H.1 (Max- k -DNF formula). *Fix $n, m, k \in \mathbb{N}$. A Max- k -DNF formula with n variables and m clauses is a tuple $\varphi = (\mathcal{C}_1, \dots, \mathcal{C}_m)$, where each clause \mathcal{C}_i consists of a subset $S_i \subseteq [n]$ of size $|S_i| \leq k$, and a partial assignment $f_i : S_i \rightarrow \{-1, 1\}$. The value of φ is*

$$\text{val}_{\text{DNF}}(\varphi) := \max_{x \in \{-1, 1\}^n} \text{val}_{\text{DNF}}(\varphi; x),$$

where

$$\text{val}_{\text{DNF}}(\varphi; x) := \sum_{i=1}^m \mathbb{1}[\forall j \in S_i : x_j = f_i(j)].$$

The Max- k -DNF problem is to compute $\text{val}_{\text{DNF}}(\varphi)$ for a given formula φ . Under the randomized Exponential Time Hypothesis ([Conjecture H.1](#)), even *approximating* this value is computationally hard – see [Theorem H.2](#) in [Appendix H.5](#) for the precise statement that we will need. This motivates the following reduction, which shows that any proper online alignment algorithm for the linear softmax policy class gives an approximation algorithm for Max- k -DNF.

Lemma H.1. *Let Alg be a proper ([Definition 4.1](#)) online alignment algorithm for linear softmax policies, in the strong oracle setting, with parameter set Θ , which uses $T_{\text{data}}(\cdot)$ reward oracle queries and has time complexity bounded by $T_{\text{comp}}(\cdot)$. Suppose also that the output of Alg lies in Π . Define*

$$\beta(k, \delta) := \frac{1}{k^2 \log(16/\delta)}, \quad \text{and} \quad \varepsilon(k, \delta) := \frac{\delta^2}{16k^2 \log(16/\delta)}.$$

Then there is an algorithm Alg' for Max- k -DNF with the following guarantee: given any parameter $\delta > 0$ and Max- k -DNF formula φ with d variables and m clauses,

- *If $\text{val}_{\text{DNF}}(\varphi) \geq \delta m$ and $T_{\text{data}}(d, \beta(k, \delta), \varepsilon(k, \delta), 1/4) \leq 2^k$, then Alg' outputs YES with probability at least $1/4$.*
- *If $\text{val}_{\text{DNF}}(\varphi) \leq \frac{\delta m}{16 \cdot T_{\text{data}}(d, \beta(k, \delta), \varepsilon(k, \delta), 1/4)}$, then Alg' outputs NO.*

Moreover, the time complexity of Alg' is $\text{poly}(d, m) \cdot T_{\text{comp}}(d, \beta(k, \delta), \varepsilon(k, \delta), 1/4)$.

Organization of appendix. In [Appendix H.2](#), we prove [Lemma H.1](#). In [Appendix H.3](#), we use this result to prove [Theorem H.1](#): in particular, if the proper alignment algorithm hypothesized in [Theorem H.1](#) exists, then [Lemma H.1](#) gives an algorithm for approximating Max- k -DNF that, by [Theorem H.2](#), violates the randomized Exponential Time Hypothesis ([Conjecture H.1](#)). Note that in [Lemma H.1](#), the approximation factor for Alg' depends on the algorithm's sample complexity $T_{\text{data}}(d, \beta(k, \delta), \varepsilon(k, \delta), 1/4)$, and consequently the assumption that the algorithm is data-efficient (i.e., T_{data} does not scale with $\exp(\beta^{-1})$ or $C_{\text{cov}}(\pi_\beta^*)$) is essential for the argument to hold. Finally, in [Appendix H.4](#) we complete the proof of [Theorem 4.1](#) by showing that properness of the output policy is essentially without loss of generality from a computational perspective. The hardness of approximation result for Max- k -DNF is deferred to [Appendix H.5](#).

H.2. Proof of Lemma H.1

Before proceeding to the proof of [Theorem 4.1](#), we prove [Lemma H.1](#) by introducing a method for embedding a Max- k -DNF formula into an instance of the online alignment problem satisfying $r^*(x, y) = \langle \theta^*, \phi(x, y) \rangle$ (and thus [Assumption 1.1](#)).

Embedding a DNF formula. Given a Max- k -DNF formula $\varphi = (\mathcal{C}_1, \dots, \mathcal{C}_m)$ with d variables and m clauses, and some $\beta > 0$, we define an instance $\mathcal{I}(\varphi)$ of the online alignment problem ([Section 1.1](#)) as follows. As discussed above, we set prompt space $\mathcal{X} := \{\perp\}$ and omit dependences on \perp henceforth. We set response space $\mathcal{Y} := \{0\} \cup [m]$. We set the regularization parameter to be β . The reference policy $\pi_{\text{ref}} \in \Delta(\mathcal{Y})$ is defined by $\pi_{\text{ref}}(0) := 1 - \varepsilon_{\text{ref}}$ and $\pi_{\text{ref}}(i) := \varepsilon_{\text{ref}}/m$ for all $i \in [m]$, where we write $\varepsilon_{\text{ref}} := e^{-1/\beta}$. We consider the linear softmax policy class $\Pi = \{\pi_\theta : \theta \in \Theta\}$ with $\Theta := \{\theta \in \mathbb{R}^d : \|\theta\|_\infty \leq 1\}$, and with feature mapping $\phi : \mathcal{Y} \rightarrow \mathbb{R}^d$ defined by $\phi(0) := 0$ and

$$\phi(i)_j := \begin{cases} \frac{f_i(j)}{k} & \text{if } j \in S_i \\ 0 & \text{otherwise} \end{cases}$$

for each $i \in [m]$ and $j \in [d]$ (recall from [Definition H.1](#) that S_i and f_i are the variable set and partial assignment, respectively, corresponding to clause \mathcal{C}_i). Finally, the reward function $r^* : \mathcal{Y} \rightarrow [-1, 1]$ is defined by $r^*(y) = \langle \phi(y), \theta^* \rangle$ where $\theta^* \in \{-1, 1\}^d \subset \Theta$ is any vector satisfying $\text{val}_{\text{DNF}}(\varphi; \theta^*) = \text{val}_{\text{DNF}}(\varphi)$. Since the reward is linear in θ^* , [Assumption 1.1](#) is satisfied, and since $\|\phi(y)\|_2 \leq \|\phi(y)\|_1 \leq 1$ for all $y \in \mathcal{Y}$ and Θ is contained in the Euclidean ball of radius \sqrt{d} , we see that [Assumption 2.1](#) is satisfied with $R_{\max} := 1$ and $B := \sqrt{d}$.¹⁷

The following lemma relates the value of φ to the maximum likelihood (over all policies in the policy class Π) of observing some non-zero response. More precisely, it shows that if $\text{val}_{\text{DNF}}(\varphi)$ is large, then π_{θ^*} (the optimal KL-regularized policy) puts non-trivial mass on non-zero responses; conversely, if $\text{val}_{\text{DNF}}(\varphi)$ is small and β is sufficiently small, then *no* policy puts non-trivial mass on non-zero responses.

Lemma H.2. *Suppose that $\beta \leq 1/\log(2)$. It holds that $\sum_{i=1}^m \pi_{\theta^*}(i) \geq \frac{\text{val}_{\text{DNF}}(\varphi)}{2m}$ and, for any $\theta \in \Theta$,*

$$\sum_{i=1}^m \pi_\theta(i) \leq 2 \left(\frac{\text{val}_{\text{DNF}}(\varphi; \text{sgn}(\theta))}{m} + e^{-1/(\beta k)} \right).$$

Proof of Lemma H.2. For each $i \in [m]$ such that the assignment θ^* satisfies clause \mathcal{C}_i , we have by definition that $\theta_j^* = f_i(j)$ for all $j \in S_i$; hence, by definition of $\phi(i)$,

$$e^{\frac{1}{\beta} \langle \theta^*, \phi(i) \rangle} = e^{\frac{1}{k\beta} \sum_{j \in S_i} f_i(j) \theta_j^*} = e^{1/\beta}.$$

Since θ^* satisfies $\text{val}_{\text{DNF}}(\varphi; \theta^*)$ clauses, and $\pi_{\text{ref}}(i) = e^{-1/\beta}/m$ for all $i \in [m]$, we get

$$\sum_{i=1}^m \pi_{\theta^*}(i) = \frac{\sum_{i=1}^m \pi_{\text{ref}}(i) e^{\frac{1}{\beta} \langle \theta^*, \phi(i) \rangle}}{\pi_{\text{ref}}(0) + \sum_{i=1}^m \pi_{\text{ref}}(i) e^{\frac{1}{\beta} \langle \theta^*, \phi(i) \rangle}} \geq \frac{\text{val}_{\text{DNF}}(\varphi; \theta^*)/m}{\pi_{\text{ref}}(0) + \text{val}_{\text{DNF}}(\varphi; \theta^*)/m} \geq \frac{\text{val}_{\text{DNF}}(\varphi; \theta^*)}{2m}$$

17. Technically, [Assumption 2.1](#) requires the rewards to lie in $[0, 1]$. This is straightforward to fix by adding a constant feature $\phi(i)_{d+1} := 1/2$, scaling all other features by a factor of $1/2$, and setting $\theta_{d+1}^* = 1$. With this modification, [Lemma H.2](#) still holds except the additive term $e^{-1/(\beta k)}$ becomes $e^{-1/(2\beta k)}$, and the proof of [Lemma H.1](#) goes through unchanged so long as $k \geq 2$.

where the first inequality uses monotonicity of $z \mapsto \frac{z}{\pi_{\text{ref}}(0)+z}$, and the final inequality uses that $\pi_{\text{ref}}(0) \leq 1$ and $\text{val}_{\text{DNF}}(\varphi; \theta^*) \leq m$.

Next, for any $\theta \in \Theta$, set $x := \text{sgn}(\theta)$. For each $i \in [m]$, we have the bound $e^{\frac{1}{\beta} \langle \theta, \phi(i) \rangle} \leq e^{\frac{1}{\beta} \|\theta\|_\infty \|\phi(i)\|_1} \leq e^{1/\beta}$. Additionally, if the assignment x does not satisfy clause \mathcal{C}_i , then there is some $j^* \in S_i$ such that $\phi(i)_{j^*} \theta_{j^*} \leq 0$, and thus

$$e^{\frac{1}{\beta} \langle \theta, \phi(i) \rangle} \leq e^{\frac{1}{\beta k} \sum_{j \in S_i \setminus \{j^*\}} f_i(j) \theta_j} \leq e^{\frac{k-1}{k\beta}},$$

where the final inequality uses that $|f_i(j)| \leq 1$ and $|\theta_j| \leq 1$ for all $j \in S_i \setminus \{j^*\}$. Recalling that $\pi_{\text{ref}}(i) = \frac{1}{e^{1/\beta} m}$ for $i \in [m]$, it follows that

$$\sum_{i=1}^m \pi_{\text{ref}}(i) e^{\frac{1}{\beta} \langle \theta, \phi(i) \rangle} \leq \frac{1}{e^{1/\beta} m} \left(m e^{\frac{k-1}{k\beta}} + \text{val}_{\text{DNF}}(\varphi; x) e^{1/\beta} \right) = \frac{\text{val}_{\text{DNF}}(\varphi; x)}{m} + e^{-1/(\beta k)}.$$

Thus,

$$\sum_{i=1}^m \pi_\theta(i) = \frac{\sum_{i=1}^m \pi_{\text{ref}}(i) e^{\frac{1}{\beta} \langle \theta, \phi(i) \rangle}}{\pi_{\text{ref}}(0) + \sum_{i=1}^m \pi_{\text{ref}}(i) e^{\frac{1}{\beta} \langle \theta, \phi(i) \rangle}} \leq 2 \sum_{i=1}^m \pi_{\text{ref}}(i) e^{\frac{1}{\beta} \langle \theta, \phi(i) \rangle} \leq 2 \left(\frac{\text{val}_{\text{DNF}}(\varphi; x)}{m} + e^{-1/(\beta k)} \right)$$

where the first inequality uses the bound $\pi_{\text{ref}}(0) = 1 - e^{-1/\beta} \geq 1/2$. \square

The following lemma implies that approximately maximizing $\sum_{i=1}^m \pi_\theta(i)$ is necessary in order to obtain low (KL-regularized) regret.

Lemma H.3. *For any $\pi \in \Delta(\mathcal{Y})$,*

$$J_\beta(\pi_{\theta^*}) - J_\beta(\pi) = \beta D_{\text{KL}}(\pi \parallel \pi_{\theta^*}) \geq \beta \left(\sum_{i=1}^m \pi(i) - \sum_{i=1}^m \pi_{\theta^*}(i) \right)^2.$$

Proof of Lemma H.3. The inequality is a consequence of Pinsker's inequality; the equality is via the following standard manipulation. Define $Z_{\theta^*} := \sum_{y \in \mathcal{Y}} \pi_{\text{ref}}(y) e^{\frac{1}{\beta} \langle \theta^*, \phi(y) \rangle}$. For any $y \in \mathcal{Y}$, we have by definition of $\pi_{\theta^*}(y)$ that

$$\langle \theta^*, \phi(y) \rangle - \beta \log \frac{\pi_{\theta^*}(y)}{\pi_{\text{ref}}(y)} = \beta \log Z_{\theta^*}$$

which is notably independent of y . Thus,

$$\begin{aligned} J_\beta(\pi_{\theta^*}) - J_\beta(\pi) &= \mathbb{E}_{y \sim \pi_{\theta^*}} \left[\langle \theta^*, \phi(y) \rangle - \beta \log \frac{\pi_{\theta^*}(y)}{\pi_{\text{ref}}(y)} \right] - \mathbb{E}_{y \sim \pi} \left[\langle \theta^*, \phi(y) \rangle - \beta \log \frac{\pi(y)}{\pi_{\text{ref}}(y)} \right] \\ &= \mathbb{E}_{y \sim \pi_{\theta^*}} \left[\langle \theta^*, \phi(y) \rangle - \beta \log \frac{\pi_{\theta^*}(y)}{\pi_{\text{ref}}(y)} \right] - \mathbb{E}_{y \sim \pi} \left[\langle \theta^*, \phi(y) \rangle - \beta \log \frac{\pi_{\theta^*}(y)}{\pi_{\text{ref}}(y)} \right] \\ &\quad + \mathbb{E}_{y \sim \pi} \left[\beta \log \frac{\pi(y)}{\pi_{\theta^*}(y)} \right] \\ &= \beta D_{\text{KL}}(\pi \parallel \pi_{\theta^*}). \end{aligned}$$

This completes the proof. \square

Together, [Lemmas H.2](#) and [H.3](#) suggest that for appropriate parameter choices, solving the constructed online alignment problem necessitates solving the original Max- k -DNF problem. The remaining (important!) subtlety is that it is impossible to efficiently simulate the data collection when only given φ , since the reward function r^* depends on the maximally satisfying assignment θ^* . Instead, we *approximately* simulate the data collection by always producing reward 0. This is only incorrect on non-zero responses, so in round t , it is unlikely to be incorrect unless $\sum_{i=1}^m \pi^t(i)$ is already non-negligible. In this event, the simulation may fail, but since Alg was assumed to be a proper-exploration algorithm, we can apply [Lemma H.2](#) to $\pi^t = \pi_{\theta_t}$ to show that $\text{val}_{\text{DNF}}(\varphi; \text{sgn}(\theta_t))$ is a decent approximation for $\text{val}_{\text{DNF}}(\varphi)$. Thus, we get a win-win reduction, where the approximation factor scales with the number of data collection rounds. We make this argument formal below, proving [Lemma H.1](#).

Proof of [Lemma H.1](#). Given a proper online alignment algorithm Alg and a Max- k -DNF formula φ with d variables and m clauses, and a parameter $\delta \in (0, 1)$, we define Alg' to have the below behavior. Throughout the remainder of the proof, we abbreviate $\beta := \beta(k, \delta)$ and $\varepsilon := \varepsilon(k, \delta)$ for simplicity:

1. Simulate Alg on the online alignment instance $\mathcal{I}(\varphi)$ defined above (with regularization parameter β , error tolerance ε , and failure probability $1/4$), but use reward function $\hat{r}(y) := 0$ instead of r^* . In particular:
 - When Alg queries the sampling oracle with $\theta \in \Theta$, Alg' computes $\pi_\theta \in \Delta(\mathcal{Y})$, samples $y \sim \pi_\theta$, and passes response y to Alg.
 - When Alg initiates data collection round t with exploration policy $\pi^t = \pi_{\theta_t}$, Alg' computes π_{θ_t} , samples $y \sim \pi_{\theta_t}$, and passes response y and reward 0 to Alg.

Let q denote the number of data collection rounds. Let $\hat{\pi} = \pi_{\theta_{\text{final}}}$ denote the final policy output by Alg.

2. Compute

$$\hat{\theta} := \arg \max_{\theta \in \{\theta_1, \dots, \theta_q, \theta_{\text{final}}\}} \sum_{i=1}^m \pi_\theta(i).$$

3. Set $\hat{x} := \text{sgn}(\hat{\theta}) \in \{-1, 1\}^d$. Alg' outputs YES if $\text{val}_{\text{DNF}}(\varphi; \hat{x}) > \frac{\delta m}{16 \cdot S(d, \beta(k, \delta), \varepsilon(k, \delta))}$ and NO otherwise.

We now analyze Alg'. Suppose that $\text{val}(\varphi) \geq \delta m$ and $T_{\text{data}}(d, \beta, \varepsilon, 1/4) \leq 2^k$. Let $\widetilde{\text{Alg}}'$ denote the idealized modification of Alg' that simulates Alg with the true reward function r^* (which is computationally hard to implement). Further, let $\overline{\text{Alg}}'$ denote the idealized modification of Alg' that simulates Alg with the true reward function only on queries θ_t with $\sum_{i=1}^m \pi_{\theta_t}(i) > \frac{1}{4T_{\text{data}}(d, \beta, \varepsilon, 1/4)}$ (and with reward 0 otherwise). We can couple the executions of $\widetilde{\text{Alg}}'(\varphi, \delta)$ and $\overline{\text{Alg}}'(\varphi, \delta)$ with the execution of Alg'(φ, δ). Observe that the execution of $\widetilde{\text{Alg}}'$ only deviates from the execution of $\overline{\text{Alg}}'$ if there is some round t where $\sum_{i=1}^m \pi_{\theta_t}(i) \leq \frac{1}{4T_{\text{data}}(d, \beta, \varepsilon, 1/4)}$ and yet the sample $y \sim \pi_{\theta_t}$ is non-zero. For any fixed t , this occurs with probability at most $\frac{1}{4T_{\text{data}}(d, \beta, \varepsilon, 1/4)}$, so by a union bound and the assumption that Alg uses at most $T_{\text{data}}(d, \beta, \varepsilon, 1/4)$ rounds, the two algorithms deviate with

probability at most $1/4$. Next, the execution of $\widetilde{\text{Alg}}'$ deviates from the execution of Alg' only if there is some round t such that $\sum_{i=1}^m \pi_{\theta_t}(i) > \frac{1}{4T_{\text{data}}(d, \beta, \varepsilon, 1/4)}$. Thus,

$$\begin{aligned} \mathbb{P}[\widetilde{\text{Alg}}'(\varphi, \delta) \neq \text{Alg}'(\varphi, \delta)] &\leq \mathbb{P}[\widetilde{\text{Alg}}'(\varphi, \delta) \neq \overline{\text{Alg}}'(\varphi, \delta)] + \mathbb{P}[\overline{\text{Alg}}'(\varphi, \delta) \neq \text{Alg}'(\varphi, \delta)] \\ &\leq \frac{1}{4} + \underbrace{\mathbb{P}^{\text{Alg}'} \left[\max_{1 \leq j \leq q} \sum_{i=1}^m \pi_{\theta_j}(i) > \frac{1}{4T_{\text{data}}(d, \beta, \varepsilon, 1/4)} \right]}_{\dagger}. \end{aligned}$$

We distinguish two cases based on the value of \dagger .

- In the first case, if \dagger is at most $1/4$, then

$$\mathbb{P}[\widetilde{\text{Alg}}'(\varphi, \delta) \neq \text{Alg}'(\varphi, \delta)] \leq 1/2.$$

By the guarantee of Alg , it holds with probability at least $3/4$ over the execution of $\widetilde{\text{Alg}}'$ that the output θ_{final} of the simulated Alg satisfies $J_{\beta}(\pi_{\theta}^*) - J_{\beta}(\pi_{\theta_{\text{final}}}) \leq \varepsilon$. Hence, the same bound holds with probability at least $1/4$ over the execution of Alg' . Condition on this event. Then

$$\sum_{i=1}^m \pi_{\widehat{\theta}}(i) \geq \sum_{i=1}^m \pi_{\theta_{\text{final}}}(i) \geq \sum_{i=1}^m \pi_{\theta}^*(i) - \sqrt{\frac{\varepsilon}{\beta}} \geq \frac{\text{val}_{\text{DNF}}(\varphi)}{2m} - \sqrt{\frac{\varepsilon}{\beta}} \geq \frac{\delta}{4}$$

where the first inequality is by definition of $\widehat{\theta}$, the second inequality is by [Lemma H.3](#), the third inequality is by [Lemma H.2](#), and the fourth inequality is by assumption that $\text{val}_{\text{DNF}}(\varphi) \geq \delta m$ and choice of $\varepsilon = \varepsilon(k, \delta)$. It follows that

$$\frac{\text{val}_{\text{DNF}}(\varphi; \text{sgn}(\widehat{\theta}))}{m} \geq \frac{1}{2} \sum_{i=1}^m \pi_{\widehat{\theta}}(i) - e^{-1/(\beta k)} \geq \frac{1}{2} \left(\frac{\delta}{4} - \left(\frac{\delta}{16} \right)^k \right) \geq \frac{\delta}{16},$$

where the first inequality is by [Lemma H.2](#), and the second inequality is by choice of $\beta = \beta(k, \delta)$.

- In the second case,

$$\dagger = \mathbb{P}^{\text{Alg}'} \left[\max_{1 \leq j \leq q} \sum_{i=1}^m \pi_{\theta_j}(i) > \frac{1}{4T_{\text{data}}(d, \beta, \varepsilon, 1/4)} \right] > \frac{1}{4}.$$

Condition on the event within the probability occurring. In this event, which occurs with probability at least $1/4$, we have $\sum_{i=1}^m \pi_{\widehat{\theta}}(i) \geq \frac{1}{4 \cdot T_{\text{data}}(d, \beta, \varepsilon, 1/4)}$ by definition of $\widehat{\theta}$, and hence

$$\begin{aligned} \frac{\text{val}_{\text{DNF}}(\varphi; \text{sgn}(\widehat{\theta}))}{m} &\geq \frac{1}{2} \sum_{i=1}^m \pi_{\widehat{\theta}}(i) - e^{-1/(\beta k)} \\ &\geq \frac{1}{8 \cdot T_{\text{data}}(d, \beta, \varepsilon, 1/4)} - \left(\frac{\delta}{16} \right)^k \\ &> \frac{\delta}{16 \cdot T_{\text{data}}(d, \beta, \varepsilon, 1/4)} \end{aligned}$$

where the first inequality is by [Lemma H.2](#), the second is by the conditioning and the choice of β , and the third inequality is by the theorem assumption that $T_{\text{data}}(d, \beta, \varepsilon, 1/4) \leq 2^k$ and $\delta \in (0, 1)$.

In both cases, the output of Alg' is therefore YES with probability at least $1/4$. On the other hand, if $\text{val}_{\text{DNF}}(\varphi) \leq \frac{\delta m}{16 \cdot T_{\text{data}}(d, \beta(k, \delta), \varepsilon(k, \delta), 1/4)}$, then it is immediate that Alg' outputs NO. Finally, the time complexity of Alg' is

$$\text{poly}(d, m) \cdot T_{\text{comp}} \left(d, \frac{1}{k^2 \log(16/\delta)}, \frac{\delta^2}{16k^2 \log(16/\delta)}, 1/4 \right)$$

by the assumed time complexity bound for Alg and the fact that for any given $\theta \in \Theta$, the distribution π_θ can be explicitly computed from φ in time $\text{poly}(d, m)$. \square

H.3. Proof of Theorem H.1

We now prove [Theorem H.1](#) by combining [Lemma H.1](#) with a hardness of approximation result for Max- k -DNF ([Theorem H.2](#), stated and proven in [Appendix H.5](#)).

Proof of Theorem H.1. Suppose that there is a proper alignment algorithm Alg with proper output, using a strong sampling oracle and a Euclidean projection oracle for Θ , that has $T_{\text{data}}(d, \beta, \varepsilon, \delta) \leq \text{poly}(d, \beta^{-1}, \varepsilon^{-1}, \delta^{-1}, \log \frac{1}{\min_{x,y} \pi_{\text{ref}}(y|x)})$ and has runtime bounded by

$$T_{\text{comp}}(d, \beta, \varepsilon, \delta) = \text{poly} \left(d, \exp(\beta^{-1}), \varepsilon^{-1}, \delta^{-1}, |\mathcal{Y}|, \log \frac{1}{\min_{x,y} \pi_{\text{ref}}(y|x)} \right).$$

The construction in [Lemma H.1](#) uses $\min_{x,y} \pi_{\text{ref}}(y|x) = 1/(e^{1/\beta} m)$ and, without loss of generality, $m \leq 2^d$ (in the regime $m > 2^d$, the Max- k -DNF problem can be solved in $\text{poly}(m)$ time unconditionally). Thus $\log(1/\min_{x,y} \pi_{\text{ref}}(y|x)) \leq \text{poly}(\beta^{-1}, d)$. It follows from the assumed bound on T_{data} that for the problem instances constructed in the proof of [Lemma H.1](#), there is a constant $c_1 > 0$ such that $T_{\text{data}}(d, \beta, \varepsilon, 1/4) \leq (d/(\beta\varepsilon))^{c_1}$. Set $\delta = \delta(d) := 1/d$ and $k = k(d) := C_{H.2} \cdot (4c_1)^2 \log(d)$. Then, recalling the definitions of $\beta(k, \delta)$ and $\varepsilon(k, \delta)$ from [Lemma H.1](#),

$$T_{\text{data}}(d, \beta(k, \delta), \varepsilon(k, \delta), 1/4) \leq \left(\frac{16dk^4 \log^2(16/\delta)}{\delta^2} \right)^{c_1} \leq O(c_1^{c_1}) \cdot (d^3 \log^6(16d))^{c_1} \leq \frac{d^{4c_1}}{16} \leq 2^k$$

where the third inequality holds for all sufficiently large d . By [Lemma H.1](#), there is an algorithm Alg' with the following guarantees on an input $k(d)$ -DNF formula φ with d variables and m clauses:

- If $\text{val}_{\text{DNF}}(\varphi) \geq \delta m = m/d$, then Alg' outputs YES with probability at least $1/4$.
- If $\text{val}_{\text{DNF}}(\varphi) \leq m/d^{4c_1}$, then since $16 \cdot T_{\text{data}}(d, \beta(k, \delta), \varepsilon(k, \delta), 1/4) \leq d^{4c_1}$ for sufficiently large d , we have $\text{val}_{\text{DNF}}(\varphi) \leq m/(16 \cdot T_{\text{data}}(d, \beta(k, \delta), \varepsilon(k, \delta), 1/4))$, so Alg' outputs NO.

Next we analyze the time complexity of Alg' . The construction in [Lemma H.1](#) uses $|\mathcal{Y}| = m$ and $\min_{x,y} \pi_{\text{ref}}(y|x) = 1/(e^{1/\beta} m)$. Thus, $T_{\text{comp}}(d, \beta(k, \delta), \varepsilon(k, \delta), 1/4) \leq \text{poly}(d, m, \exp(1/\beta(k, \delta)), 1/\varepsilon(k, \delta)) \leq \text{poly}(2^{O(\log^3(d))}, m)$. So from [Lemma H.1](#), we get that the time complexity of Alg' is $\text{poly}(2^{O(\log^3(d))}, m)$. Hence, applying [Theorem H.2](#) with parameter $c := 4c_1$ and using that $k(d) \geq C_{H.2}(4c_1)^2 \log(d)$, we get that the randomized Exponential Time Hypothesis is false (concretely, [Theorem H.2](#) rules out the possibility of time complexity $\text{poly}(2^{O(d^\eta)}, m)$ for a constant $\eta = \eta(4c_1)$). \square

H.4. Proof of Theorem 4.1

We now prove [Theorem 4.1](#) by combining [Theorem H.1](#) with the following result, which shows that any proper alignment algorithm with improper output can be efficiently bootstrapped to a proper alignment algorithm with proper output. The proof of [Lemma H.4](#) follows from an analysis of log-loss behavior cloning due to [\[72\]](#).

Lemma H.4. *Let Alg be a proper alignment algorithm with a strong sampling oracle and a Euclidean projection oracle for Θ , that uses $T_{\text{data}}(\cdot)$ reward oracle queries and has time complexity $T_{\text{comp}}(\cdot)$, under [Assumption 2.1](#) with $R_{\max} := 1$ and $B := \sqrt{d}$. Suppose that the output of Alg is sampleable in time T per prompt. Then, in the same setting, there is a proper alignment algorithm Alg' with proper output $\hat{\pi} \in \Pi$, that uses $T'_{\text{data}}(d, \beta, \epsilon, \delta) = T_{\text{data}}(d, \beta, \epsilon_0, \delta/2)$ reward oracle queries and has sampling oracle complexity*

$$T'_{\text{comp}}(d, \beta, \epsilon, \delta) = T_{\text{comp}}(d, \beta, \epsilon_0, \delta/2) + \text{poly}\left(d, |\mathcal{Y}|, B, T, \epsilon^{-1}, \log(\delta^{-1}), \log \frac{1}{\min_{x,y} \pi_{\text{ref}}(y | x)}\right),$$

where $\epsilon_0 := \frac{c\epsilon}{B \log(1/\min_{x,y} \pi_{\text{ref}}(y|x))}$ for a universal constant $c > 0$, and we assume that $\beta \in (0, 1)$. Furthermore, the time complexity of Alg' is polynomial in the time complexity of Alg with parameters $(d, \beta, \epsilon_0, \delta/2)$ and $\text{poly}\left(d, |\mathcal{Y}|, B, T, \epsilon^{-1}, \log(\delta^{-1}), \log \frac{1}{\min_{x,y} \pi_{\text{ref}}(y|x)}\right)$.

Proof of Lemma H.4. For any $\theta \in \Theta$, $x \in \mathcal{X}$, and $y \in \mathcal{Y}$, observe that $\pi_{\theta}(y | x) \geq e^{-2B} \pi_{\text{ref}}(y | x)$. It follows that for any $\pi : \mathcal{X} \rightarrow \Delta(\mathcal{Y})$,

$$\max_{x,y} \max_{\theta \in \Theta} \frac{\pi(y | x)}{\pi_{\theta}(y | x)} \leq W := \frac{e^{2B}}{\min_{x,y} \pi_{\text{ref}}(y | x)}.$$

We will use a slight generalization of Proposition D.2 in [\[72\]](#), which shows that there is an algorithm that takes an integer n , the norm bound B from [Assumption 2.1](#), some $\delta \in (0, 1)$, and n i.i.d. samples $(x^i, y^i)_{i=1}^n$ from π , and produces $\hat{\theta} \in \Theta$ satisfying, with probability at least $1 - \delta$,

$$D_{\text{H}}^2(\pi_{\hat{\theta}}, \pi) \lesssim \frac{(d \log(Bn) + \log(W)) \log(1/\delta)}{n} + \log(W) \cdot \min_{\theta \in \Theta} D_{\text{H}}^2(\pi_{\theta}, \pi). \quad (65)$$

Additionally, the time complexity is $\text{poly}(n, d, |\mathcal{Y}|, B, \log(1/\delta))$ (to be precise, Proposition D.2 is specialized to the case where π_{ref} is uniform on \mathcal{Y} , so that $W = e^{2B}/|\mathcal{Y}|$, but this generalization is immediate from inspecting the proof; also, in our setting the horizon parameter H from the proposition is equal to one). The algorithm is essentially gradient ascent on the log-likelihood for a set of i.i.d. samples from π , which is concave in θ [\[72\]](#).

This motivates defining Alg' as follows. Execute Alg with parameters ϵ_0 and $\delta/2$, and let π be the output. Then, for a parameter n to be determined, draw n samples $(x^i)_{i=1}^n$ from the prompt distribution ρ , and for each draw $y^i \sim \pi(\cdot | x^i)$. Execute the algorithm described above with failure probability $\delta/2$, to compute $\hat{\theta} \in \Theta$, and output $\hat{\pi} := \pi_{\hat{\theta}}$.

We now analyze Alg' . With probability at least $1 - \delta/2$, it holds that $\beta D_{\text{KL}}(\pi \| \pi_{\theta^*}) = J_{\beta}(\pi_{\theta^*}) - J_{\beta}(\pi) \leq \epsilon_0$, where the equality is by [Lemma G.4](#). Also, by [Eq. \(65\)](#), with probability at least $1 - \delta/2$ it holds that

$$D_{\text{H}}^2(\hat{\pi}, \pi) \lesssim \epsilon_0 + \log(W) \cdot D_{\text{H}}^2(\pi_{\theta^*}, \pi), \quad (66)$$

so long as $n \geq \text{poly}(d, \epsilon_0^{-1}, \log(BW/\delta))$. Condition on the event that both of these bounds hold. Then

$$\begin{aligned}
J_\beta(\pi_{\theta^*}) - J_\beta(\hat{\pi}) &= \beta D_{\text{KL}}(\hat{\pi} \parallel \pi_{\theta^*}) \\
&\lesssim \beta \log(W) \cdot D_{\text{H}}^2(\hat{\pi}, \pi_{\theta^*}) \\
&\lesssim \beta \log(W) \cdot (D_{\text{H}}^2(\hat{\pi}, \pi) + D_{\text{H}}^2(\pi, \pi_{\theta^*})) \\
&\lesssim \beta \log(W) \cdot \epsilon_0 + \beta \log^2(W) D_{\text{H}}^2(\pi, \pi_{\theta^*}) \\
&\leq \log(W) \cdot \epsilon_0 + \beta \log^2(W) D_{\text{KL}}(\pi \parallel \pi_{\theta^*}) \\
&\lesssim \log^2(W) \cdot \epsilon_0
\end{aligned}$$

where the equality is by [Lemma G.4](#), the first inequality is by Lemma 4 of [\[100\]](#), the second inequality uses the fact that $D_{\text{H}}(\cdot, \cdot)$ is a metric, the third inequality is by [Eq. \(66\)](#), and the fourth inequality uses e.g. (7.33) in [\[67\]](#) as well as the bound $\beta \leq 1$. By definition of W and ϵ_0 (so long as the constant $c > 0$ is sufficiently small), we can bound the above by ϵ , so Alg' satisfies the correctness desideratum of a proper alignment algorithm.

Since the second phase of Alg' uses no reward oracle queries, the claimed bound on total reward oracle queries is immediate from the parameter choices in the call to Alg . Additionally, the claimed time complexity bound follows from the guarantee from [\[72\]](#), and the assumption that the output of Alg is sampleable in time T , so long as we choose $n = \text{poly}(d, \epsilon_0^{-1}, \log(BW/\delta))$. \square

Proof of Theorem 4.1. Suppose that there is a proper alignment algorithm with access to a strong sampling oracle and a Euclidean projection oracle for Θ , that uses at most $\text{poly}(d, \beta^{-1}, \epsilon^{-1}, \delta^{-1})$ reward oracle queries and has time complexity at most $\text{poly}(d, \exp(\beta^{-1}), \epsilon^{-1}, \delta^{-1})$, under [Assumption 2.1](#) with $R_{\max} := 1$ and $B := \sqrt{d}$. Without loss of generality (assuming that the output policy is represented by a circuit), the output policy is sampleable in time $T := \text{poly}(d, \exp(\beta^{-1}), \epsilon^{-1}, \delta^{-1})$. Thus, by [Lemma H.4](#) and choice of B , in the same setting, there is a proper alignment algorithm with proper output that uses at most $\text{poly}(d, \beta^{-1}, \epsilon^{-1}, \delta^{-1}, \log(1/\min_{x,y} \pi_{\text{ref}}(y \mid x)))$ reward oracle queries and has time complexity at most $\text{poly}(d, \exp(\beta^{-1}), \epsilon^{-1}, \delta^{-1}, |\mathcal{Y}|, \log(1/\min_{x,y} \pi_{\text{ref}}(y \mid x)))$. Note that the time complexity also bounds the number of sampling oracle queries. It follows from [Theorem H.1](#) that the Randomized Exponential Time Hypothesis is false. \square

H.5. Hardness of Approximation for Max- k -DNF

In this section we prove the following hardness of approximation result [Theorem 4.1](#).

Theorem H.2. Fix any $c > 1$ and function $k : \mathbb{N} \rightarrow \mathbb{N}$. Suppose that $k(n) \geq C_{\text{H.2}} c^2 \log(n)$ for a sufficiently large universal constant $C_{\text{H.2}}$. There is $\eta = \eta(c) > 0$ with the following property. Suppose that there is a time- $\text{poly}(2^{n^\eta}, m)$ algorithm Alg that, given a $k(n)$ -DNF formula φ with n variables and m clauses, has the following behavior:

1. If $\text{val}_{\text{DNF}}(\varphi) \geq m/n$, then Alg outputs YES with probability at least $1/4$.
2. If $\text{val}_{\text{DNF}}(\varphi) \leq m/n^c$, then Alg outputs NO.

Then the randomized Exponential Time Hypothesis is false.

Essentially, the above result states that for a $\text{Max-}k(n)\text{-DNF}$ formula with n variables, for any constant $c > 1$ and so long as $k(n)$ is sufficiently large, it is computationally hard to distinguish between the case that $a := 1/n$ fraction of clauses are satisfiable versus $b := 1/n^c$ fraction of clauses are satisfiable. For the application to [Theorem 4.1](#), it is critical that the approximation gap a/b is larger than $1/a$, since the gap generated by the reduction [Lemma H.1](#) scales with $T_{\text{data}}(n, \beta(k, \delta), \varepsilon(k, \delta), 1/4)$, which in turn scales polynomially with $1/\delta = 1/a$. Additionally, we remark that it is important for k to grow with n , since sampling a random assignment gives an efficient 2^k -approximation algorithm.

To prove [Theorem H.2](#), we use a result by [\[14\]](#), which states that for any constant k , there is a sparse k -predicate P so that $\text{Max-}P$ (i.e. $\text{Max-}k\text{-CSP}$ with predicate P) is hard to approximate to any factor better than $2^k/(2k)$. We then reduce $\text{Max-}P$ to $\text{Max-}k\text{-DNF}$ (using sparsity of P to control the decay in satisfiability thresholds) and then apply serial repetition to boost the gap. To make this approach formal, we start with the following definition.

Definition H.2 (*Max- P formula*). Fix $n, m, k \in \mathbb{N}$ and any $P : \{-1, 1\}^k \rightarrow \{0, 1\}$. A $\text{Max-}P$ formula with n variables and m clauses is a tuple $\varphi = (\mathcal{C}_1, \dots, \mathcal{C}_m)$, where each clause \mathcal{C}_i is a tuple (v_i, b_i) where $v_i \in [n]^k$ and $b_i \in \{-1, 1\}^k$. The value of φ is

$$\text{val}_P(\varphi) := \max_{x \in \{-1, 1\}^n} \text{val}_P(\varphi; x)$$

where

$$\text{val}_P(\varphi; x) := \sum_{i=1}^m \mathbb{1}[P(b_{i1}x_{v_{i1}}, \dots, b_{ik}x_{v_{ik}}) = 1].$$

We say that the set of accepting assignments of P is $P^{-1}(1)$.

Note, for example, that [Definition H.1](#) corresponds to the predicate $P(x_1, \dots, x_k) = \mathbb{1}[x_1 = \dots = x_k = 1]$. The following hardness result is essentially due to [\[14\]](#).

Theorem H.3. Let $k \in \mathbb{N}$ and $\varepsilon > 0$. There is a predicate $P : \{-1, 1\}^k \rightarrow \{0, 1\}$ with at most $2k$ accepting assignments, and a real number $\gamma = \gamma(k, \varepsilon) \in (0, 1)$ such that the following property holds: suppose that there is a time- $O(2^{n^\gamma})$ algorithm that, given a $\text{Max-}P$ formula φ with at most n variables and n clauses, has the following behavior:

1. If φ is at least $(1 - \varepsilon)$ -satisfiable, then it outputs YES with probability at least $1/2$.
2. If φ is at most $(2k/2^k + \varepsilon)$ -satisfiable, then it outputs NO.

Then the Randomized Exponential Time Hypothesis ([Conjecture H.1](#)) is false.

Proof of Theorem H.3. Corollary 1.2 of [\[14\]](#) states that for any positive integer k and any $\varepsilon > 0$, for the Hadamard predicate $P : \{-1, 1\}^k \rightarrow \{0, 1\}$, which has at most $2k$ accepting assignments, distinguishing between $(1 - \varepsilon)$ -satisfiability and $(2k/2^k + \varepsilon)$ -satisfiability of a $\text{Max-}P$ formula is NP-hard.¹⁸ Thus, there is a polynomial-time reduction \mathcal{R} that takes as input any 3SAT formula τ with n variables and n clauses, and produces a $\text{Max-}P$ formula φ with $m \leq \text{poly}(n)$ clauses such that $\text{val}_P(\varphi) \geq (1 - \varepsilon)m$ if τ is satisfiable, and $\text{val}_P(\varphi) \leq (2k/2^k + \varepsilon)m$ otherwise. There is a constant $C = C(k, \varepsilon) > 0$ such that \mathcal{R} has time complexity $O(n^{C(k, \varepsilon)})$. Now let $\gamma := 1/(2C(k, \varepsilon))$ and

18. In fact, [\[14\]](#) shows that one can even take $\varepsilon = o(1)$, but for our purposes a constant suffices.

suppose that there is indeed a time- $O(2^{n^\gamma})$ randomized algorithm Alg with the behavior specified in the theorem statement. Then on input τ with n variables and n clauses, it follows from the time complexity bound on \mathcal{R} that the output φ of \mathcal{R} has at most $O(n^{C(k,\varepsilon)})$ variables and clauses, so $\text{Alg}(\varphi)$ runs in time $2^{O(n^{\gamma \cdot C(k,\varepsilon)})} = 2^{o(n)}$ by choice of γ . If τ is satisfiable, then it outputs YES with probability at least $1/2$, and otherwise it outputs NO. This contradicts [Conjecture H.1](#). \square

Lemma H.5 (DNF embedding). *Fix $k \in \mathbb{N}$ and let $P : \{-1, 1\}^k \rightarrow \{0, 1\}$ be a predicate with ℓ accepting assignments. Then there is a $\text{poly}(n, m)$ -time algorithm that takes as input a P -formula φ with n variables and m clauses, and outputs a Max- k -DNF formula φ' with n variables and ℓm clauses, satisfying $\text{val}_P(\varphi) = \text{val}_{\text{DNF}}(\varphi')$.*

Proof of Lemma H.5. Given $\varphi = (\mathcal{C}_1, \dots, \mathcal{C}_m)$, we define φ' on the same variable set as follows. For each clause $\mathcal{C}_i = (v_i, b_i)$, for each $y \in P^{-1}(1)$, we add to φ' the clause $(S_{i,y}, f_{i,y})$ defined by $S_{i,y} = \{v_{i1}, \dots, v_{ik}\}$ and $f_{i,y}(v_{ij}) = b_{ij}y_{ij}$.

Since P has ℓ accepting assignments, φ' has ℓm clauses. Moreover, fix any assignment $x \in \{-1, 1\}^n$ and clause \mathcal{C}_i . If x satisfies \mathcal{C}_i in φ , then there is exactly one $y \in P^{-1}(1)$ such that $(b_{i1}x_{v_{i1}}, \dots, b_{ik}x_{v_{ik}}) = y$. Equivalently, there is exactly one $y \in P^{-1}(1)$ such that $x_a = f_{i,y}(a)$ for all $a \in S_{i,y}$. On the other hand, if x does not satisfy \mathcal{C}_i , then there is no such y . Thus, $\text{val}_P(\varphi; x) = \text{val}_{\text{DNF}}(\varphi'; x)$. Since this holds for all x , we get $\text{val}_P(\varphi) = \text{val}_{\text{DNF}}(\varphi')$. Finally, it's clear that the reduction is polynomial-time in the input size. \square

Lemma H.6 (Serial repetition). *There is an algorithm that takes as input a Max- k -DNF formula φ with n variables and m clauses, and a parameter $t \in \mathbb{N}$, and outputs a kt -DNF formula φ' with n variables and m^t clauses, that has value $\text{val}_{\text{DNF}}(\varphi') = (\text{val}_{\text{DNF}}(\varphi))^t$. Moreover, the time complexity of the algorithm is $\text{poly}(n, k, m^t)$.*

Proof of Lemma H.6. Given $\varphi = (\mathcal{C}_1, \dots, \mathcal{C}_m)$, we define φ' on the same variable set as φ , with clauses defined as follows. For each ordered tuple $(i_1, \dots, i_t) \in [m]^t$, we introduce the clause $\mathcal{C}_{i_1, \dots, i_t} := \mathcal{C}_{i_1} \wedge \dots \wedge \mathcal{C}_{i_t}$ to φ' . Then φ' has exactly m^t clauses. Moreover, for any assignment $x \in \{-1, 1\}^n$ which satisfies some subset $\{\mathcal{C}_i : i \in S\}$ of the original clauses, we have that x satisfies $\mathcal{C}_{i_1, \dots, i_t}$ if and only if $(i_1, \dots, i_t) \in S^t$. Thus $\text{val}_{\text{DNF}}(\varphi'; x) = \text{val}_{\text{DNF}}(\varphi; x)^t$ and so, since x was arbitrary, $\text{val}_{\text{DNF}}(\varphi') = \text{val}_{\text{DNF}}(\varphi)^t$. Finally, it's clear that the reduction is polynomial-time in the output size. \square

Proof of Theorem H.2. Fix $k_0 = k_0(c) \in \mathbb{N}$ sufficiently large and $\varepsilon_0 = \varepsilon_0(c) \in (0, 1)$ sufficiently small that the following inequality holds:

$$2^{-k_0} + \varepsilon_0 \leq \left(\frac{1 - \varepsilon_0}{2k_0} \right)^{2c}. \quad (67)$$

In particular, there is a universal constant $C_{H.2}$ so that we can always take $k_0 \leq C_{H.2}c^2$ and $\varepsilon_0 = 2^{-C_{H.2}c^2}$. Set $\eta := \gamma(k_0, \varepsilon_0)/2$, where $\gamma(k_0, \varepsilon_0) \in (0, 1)$ is the parameter guaranteed by [Theorem H.3](#). Next, let $P : \{0, 1\}^{k_0} \rightarrow \{0, 1\}$ be the predicate guaranteed by [Theorem H.3](#). We define an algorithm Alg' for Max- P that does the following, given a P -formula with n variables and m clauses:

1. Using [Lemma H.5](#) and the fact that P has only $2k_0$ accepting assignments, construct a k_0 -DNF formula φ' with n variables, $2k_0m$ clauses, and with $\text{val}_{\text{DNF}}(\varphi') = \text{val}_P(\varphi)$.
2. Set $t := \left\lfloor \frac{\log n}{\log \frac{2k_0}{1-\varepsilon_0}} \right\rfloor$. Using [Lemma H.6](#), construct a k_0t -DNF formula φ'' with n variables, $(2k_0m)^t$ clauses, and with $\text{val}_{\text{DNF}}(\varphi'') = \text{val}_{\text{DNF}}(\varphi')^t = \text{val}_P(\varphi)^t$.
3. Apply Alg to φ'' and output whatever Alg outputs.

Notice that $k_0t \leq C_{H.2}c^2 \log(n) \leq k(n)$, so φ'' is a Max- k -DNF formula and hence the application of Alg to φ'' is well-defined. If $\text{val}_P(\varphi) \geq (1 - \varepsilon_0)m$, then

$$\text{val}_{\text{DNF}}(\varphi'') \geq (1 - \varepsilon_0)^t m^t = (2k_0m)^t \left(\frac{1 - \varepsilon_0}{2k_0} \right)^t \geq \frac{(2k_0m)^t}{n}$$

since $t \leq \frac{\log n}{\log \frac{2k_0}{1-\varepsilon_0}}$. Thus, Alg' outputs YES in this case, with probability at least $1/4$; we can boost this probability to $1/2$ by running the algorithm independently $O(1)$ times. On the other hand, if $\text{val}_P(\varphi) \leq (2k_0/2^{k_0} + \varepsilon_0)m$, then

$$\begin{aligned} \text{val}_{\text{DNF}}(\varphi'') &\leq (2k_0/2^{k_0} + \varepsilon_0)^t m^t \\ &\leq (2^{-k_0} + \varepsilon_0)^t (2k_0m)^t \\ &\leq \left(\frac{1 - \varepsilon_0}{2k_0} \right)^{2ct} (2k_0m)^t \\ &\leq \left(\frac{1 - \varepsilon_0}{2k_0} \right)^{2c \left(\frac{\log n}{\log \frac{2k_0}{1-\varepsilon_0}} - 1 \right)} (2k_0m)^t = \left(\frac{2k_0}{1 - \varepsilon_0} \right)^{2c} \frac{(2k_0m)^t}{n^{2c}} \leq \frac{(2k_0m)^t}{n^c} \end{aligned}$$

where the third inequality is by [Eq. \(67\)](#), the fourth inequality is by definition of t , and the final inequality holds so long as $n \geq (2k_0/(1 - \varepsilon_0))^2$. Thus, Alg' outputs NO with probability 1 in this case.

The time complexity of Alg', dominated by the invocation of Alg on φ'' , is $\text{poly}(2^{n^\eta}, (2k_0m)^t)$. For $m \leq n$, this is dominated by $2^{O(n^\eta)}$ for sufficiently large n , since $t \leq \log(n)$. Since $\eta = \gamma(k_0, \varepsilon_0)/2$, we conclude that Alg' has time complexity at most $O(2^{n^{\gamma(k_0)}})$, so by [Theorem H.3](#), the Randomized Exponential Time Hypothesis is false. \square

Part III

Multi-Turn Exploration: Learning Autoregressive Softmax Policies

This section of the appendix is dedicated to presenting and analyzing the MTSS algorithm ([Algorithm 4](#)). MTSS learns a near-optimal policy for any Markov Decision Process in which the optimal KL-regularized policy has autoregressive linear softmax structure (a generalization of linear- Q_β^* realizability) under reset access. This formulation subsumes the setting in [Section 5](#), encompassing general Markov Decision Processes (MDPs) that extend well beyond the token-level MDP; we prove [Theorem 5.1](#), our main result for the multi-turn exploration setting in [Section 5](#) as a special case. We adopt this formulation because (i) it makes the essential ingredients in our algorithm and analysis as clear as possible; and (ii) we believe the results are likely to be of interest more broadly, beyond language modeling. This section is organized as follows:

- [Appendix I](#): We introduce the general reinforcement learning setting and statistical assumptions.
- [Appendix J](#): We present and describe the multi-turn algorithm, `MultiTurnSpannerSampling` (MTSS; [Algorithm 4](#)), and state its main guarantee, [Theorem J.1](#) (generalizing [Theorem 5.1](#)).
- [Appendices F, K and L](#): We provide the main guarantees for the subroutines used within MTSS.
- [Appendix M](#): We combine the preceding results to prove the main guarantee for MTSS ([Theorem J.1](#)).
- [Appendix N](#): Supporting technical lemmas for the proofs of the results above.

Appendix I. Preliminaries for Multi-Turn Exploration

In this section, we formally introduce the setting and assumptions for our general multi-turn exploration results. Recall that the language model alignment problem in [Section 5](#) is a special case of episodic reinforcement learning in a specific (“token-level”) Markov Decision Process, where *actions* are tokens or sub-sequences of tokens, and the *state* consists of the prompt and all of the tokens generated so far, with the *reward* determined by the alignment objective. This is a rather simple MDP, as the transition dynamics are deterministic, and are known a-priori.

Our results in this section encompass a more general setting where the MDP transitions are unknown and stochastic, but where the agent has the ability to reset to previously observed states during the learning process (in addition to standard episodic access); this setting is also known as reinforcement learning with *local simulator access* [[51](#), [103](#), [91](#), [60](#)]. In the context of language model alignment, the assumption of a local simulator is without loss of generality because the MDP dynamics are known; resetting the state simply involves feeding the prompt and partial prefix of tokens back into the policy [[15](#)]. We present our results for *any stochastic MDP*, provided that local simulator access is available. To achieve statistical and computational efficiency, we make statistical and computational assumptions that generalize [Section 5](#), focusing on KL-regularized regret, and assuming that the optimal regularized policy has linear softmax structure that generalizes [Eq. \(11\)](#).

A remark on notation. Throughout [Part III](#) of the appendix, we use boldface notation (e.g., \mathbf{x}_h , \mathbf{y}_h , and \mathbf{a}_h) to denote realizations of random variables. This allows certain arguments that require conditioning to be presented in the clearest way possible.

I.1. MDP Setting and Multi-Turn Reinforcement Learning Framework

A Markov Decision Process (MDP) is a tuple $\mathcal{M}^* = (\mathcal{X}, \mathcal{A}, H, P, r^*)$, where \mathcal{X} is a (large or potentially infinite) state space, \mathcal{A} is the action space (we abbreviate $A = |\mathcal{A}|$), $H \in \mathbb{N}$ is the horizon, $r^* = \{r_h^*\}_{h=1}^H$ is the reward function (where $r_h^* : \mathcal{X} \times \mathcal{A} \rightarrow [0, 1]$) and $P = \{P_h\}_{h=0}^H$ is the transition distribution (where $P_h : \mathcal{X} \times \mathcal{A} \rightarrow \Delta(\mathcal{X})$), with the convention that $P_0(\cdot \mid \emptyset)$ is the initial state distribution. A policy is a sequence of functions $\pi = \{\pi_h : \mathcal{X} \rightarrow \Delta(\mathcal{A})\}_{h=1}^H$. When a policy π is executed, it generates a trajectory $(\mathbf{x}_1, \mathbf{a}_1, \mathbf{r}_1), \dots, (\mathbf{x}_H, \mathbf{a}_H, \mathbf{r}_H)$ via the process $\mathbf{a}_h \sim \pi_h(\mathbf{x}_h)$, $\mathbf{r}_h \sim r_h^*(\mathbf{x}_h, \mathbf{a}_h)$, $\mathbf{x}_{h+1} \sim P_h(\cdot \mid \mathbf{x}_h, \mathbf{a}_h)$, initialized from $\mathbf{x}_1 \sim P_0(\cdot \mid \emptyset)$ (we use \mathbf{x}_{H+1} to denote a terminal state with zero reward). We write $\mathbb{P}_\pi[\cdot]$ and $\mathbb{E}_\pi[\cdot]$ to denote the law and expectation under this process. We assume that $\mathbf{r}_h \in [0, R_{\max}]$ for all h .

As discussed above, the *token-level MDP* formulation of language model reinforcement learning (e.g., Rafailov et al. [71]) corresponds to the case where \mathbf{x}_1 is the prompt, \mathbf{a}_h is the next token to predict, and $\mathbf{x}_h = (\mathbf{x}_1, \mathbf{a}_1, \dots, \mathbf{a}_{h-1})$ is the concatenation of the prompt with all of the tokens so far; the final response is $\mathbf{y} = (\mathbf{a}_1, \dots, \mathbf{a}_H)$. For the formulation in [Section 5](#), we have $r_h^* = 0$ for all $h < H$, and r_H^* represents the reward for the complete response. A slightly more general formulation (e.g., Xiong et al. [96]) which our setup also encompasses, is where each \mathbf{a}_h represents a sub-sequence of tokens rather than a single token (e.g., corresponding to a portion of a proof).

Online reinforcement learning framework. In the standard online reinforcement learning framework, the learner repeatedly interacts with an unknown stochastic MDP (where the transition distribution is not known) by executing a policy and observing the resulting trajectory, with the goal of maximizing the total reward. We begin with a base policy $\pi_{\text{ref}} = \{\pi_{h,\text{ref}}\}_{h=1}^H$. Formally, for each episode $t \in [T_{\text{prompt}}]$, the learner selects a policy $\pi^t = \{\pi_h^t\}_{h=1}^H$, executes it in the underlying MDP \mathcal{M}^* and observes the trajectory $\{(\mathbf{x}_h^t, \mathbf{a}_h^t, \mathbf{r}_h^t)\}_{h=1}^H$. After all T_{prompt} episodes conclude, the goal of the learner is to produce a policy $\hat{\pi}$ such that

$$\sup_{\pi} J_{\beta}(\pi) - J_{\beta}(\hat{\pi}) \leq \varepsilon, \quad (68)$$

for some $\varepsilon, \beta > 0$, where $J_{\beta}(\pi) := \mathbb{E}_{\pi}[\sum_{h=1}^H \mathbf{r}_h] - \beta D_{\text{KL}}(\pi \parallel \pi_{\text{ref}})$ denotes the *regularized* cumulative reward, with

$$D_{\text{KL}}(\pi \parallel \pi_{\text{ref}}) := \mathbb{E}_{\pi} \left[\sum_{h=1}^H D_{\text{KL}}(\pi_h(\mathbf{x}_h) \parallel \pi_{h,\text{ref}}(\mathbf{x}_h)) \right]. \quad (69)$$

We define $\pi_{\beta}^* = \{\pi_{h,\beta}^*\}_{h=1}^H$ as the optimal KL-regularized policy: $\pi_{\beta}^* = \arg \max_{\pi} J_{\beta}(\pi)$.

Online RL with resets (local simulator access). We focus on online RL with state resetting (local simulator access) [90, 51, 103, 91, 104, 60], which augments the online RL protocol as follows: At each episode $t \in [T_{\text{prompt}}]$, instead of starting from a random initial state $\mathbf{x}_1 \sim P_0(\cdot \mid \emptyset)$, the agent can *reset* the MDP to any layer $h \in [H]$ and any state \mathbf{x}_h previously encountered, and proceed with a new episode starting from this point. As in the online RL protocol, the goal is to produce a policy

$\hat{\pi} \in \Pi$ that satisfies (68) with as few episodes of interaction as possible. Note that when \mathcal{M}^* is the token-level MDP, this formulation precisely corresponds to the setting in Section 5.¹⁹

Linear softmax policies. We consider the class of linear softmax policies given by

$$\Pi = \left\{ \pi_\theta = \{\pi_{h,\theta}\}_{h=1}^H \mid \theta_1, \dots, \theta_H \in \mathbb{B}_d(B) \right\}$$

for a parameter $B \geq R_{\max}$, where

$$\pi_{h,\theta}(a \mid x) \propto \pi_{h,\text{ref}}(a \mid x) \cdot \exp(\beta^{-1} \langle \theta_h, \phi_h(x, a) \rangle).$$

We assume that the feature map ϕ satisfies $\sup_{h \in [H], (x,a) \in \mathcal{X} \times \mathcal{A}} \|\phi_h(x, a)\| \leq 1$. Our main assumption is that the optimal regularized policy is itself softmax-linear.

Assumption I.1 (Linear π_β^*). *We assume that for all $h \in [H]$, there exists $\theta_{h,\beta}^* \in \mathbb{B}_d(B)$ such that*

$$\forall (x, a) \in \mathcal{X} \times \mathcal{A}, \quad \pi_{h,\beta}^*(a \mid x) \propto \pi_{h,\text{ref}}(a \mid x) \cdot \exp(\beta^{-1} \langle \theta_{h,\beta}^*, \phi_h(x, a) \rangle). \quad (70)$$

KL-regularized dynamic programming. The KL-regularized RL formulation admits regularized counterparts to the standard Q - and V -value functions, defined as follows (e.g., Rafailov et al. [71], Xie et al. [94]).

Definition I.1 (State-action value function). *For any $\pi_{1:H} : \mathcal{X} \rightarrow \Delta(\mathcal{A})$ and $(x, a) \in \mathcal{X} \times \mathcal{A}$, define*

$$Q_{h,\beta}^\pi(x, a) := r_h^*(x, a) + \mathbb{E}_\pi \left[\sum_{\ell=h+1}^H r_\ell^*(\mathbf{x}_\ell, \mathbf{a}_\ell) - \beta \sum_{\ell=h+1}^H \log \frac{\pi_\ell(\mathbf{a}_\ell \mid \mathbf{x}_\ell)}{\pi_{\ell,\text{ref}}(\mathbf{a}_\ell \mid \mathbf{x}_\ell)} \mid \mathbf{x}_h = x, \mathbf{a}_h = a \right]. \quad (71)$$

Definition I.2 (KL-regularized value function ($Q_{h,\beta}^*$)). *We define the optimal regularized state-action value functions $(Q_{h,\beta}^*)_{h \in [H]}$ via backward induction over $h \in [H]$ as follows: for all $(x, a) \in \mathcal{X} \times \mathcal{A}$, $Q_{H+1,\beta}^*(x, a) = 0$ and for $h = H, \dots, 1$:*

$$Q_{h,\beta}^*(x, a) = r_h^*(x, a) + \mathcal{T}_{h,\beta}[Q_{h+1,\beta}^*](x, a),$$

where

$$\mathcal{T}_{h,\beta}[f](x, a) := \mathbb{E}_{\pi_{\text{ref}}} [V_f(\mathbf{x}_{h+1}) \mid \mathbf{x}_h = x, \mathbf{a}_h = a],$$

and

$$V_f(x) := \beta \log \sum_{a \in \mathcal{A}} \pi_{\text{ref}}(a \mid x) \cdot e^{f(x,a)/\beta}.$$

We show in Lemma N.4 that the optimal KL-regularized policy π_β^* satisfies

$$\pi_{h,\beta}^*(\cdot \mid x) \propto \pi_{h,\text{ref}}(\cdot \mid x) \cdot \exp(\beta^{-1} Q_{h,\beta}^*(x, \cdot)) \quad (72)$$

and $\pi_{h+1:H,\beta}^* \in \arg \max_{\pi_{h+1:H} \in \Pi} Q_{h,\beta}^\pi(x, a)$, for all $(x, a) \in \mathcal{X} \times \mathcal{A}$ and $h \in [H]$. Consequently, Assumption I.1 is equivalent to asserting that for all h ,

$$Q_{h,\beta}^*(x, a) - Q_{h,\beta}^*(x, a') = \langle \theta_{h,\beta}^*, \phi_h(x, a) - \phi_h(x, a') \rangle. \quad (73)$$

19. In particular, the KL-divergence in Eq. (69) coincides with the sequence-level KL used in Section 5 via the chain rule.

Thus, [Assumption I.1](#) may be viewed as a KL-regularized analogue of the so-called *linear- Q^** assumption explored throughout prior work [23, 86, 90, 51, 103, 91, 104, 60]. In particular, prior work has shown that RL with linear- Q^* and an action gap Δ is statistically intractable in the episodic RL protocol, but is tractable under reset access [90, 51], motivating this access model. Our results show that the regularization parameter β plays a similar role to the action gap Δ in enabling tractability.

Non-triviality of autoregressive realizability. The following result, as mentioned in [Section 5](#), shows that [Assumption I.1](#) is a non-trivial assumption, in the sense that it may not be satisfied even if the rewards themselves are linear.

Proposition I.1. *Consider the token-level MDP. Let $\mathcal{X} = \{\perp\}$, $\mathcal{A} = [2]$, $H = d = 2$, and $\beta = 1$. For any $\delta \in (0, 1/2)$, there exist feature maps $\phi_h \in \mathbb{R}^d$ with $\|\phi_h(x, y_{1:h})\| \leq 1$ and parameters $\theta_h^* \in \mathbb{R}^d$ with $\|\theta_h^*\| \leq B := \log(3/\delta)$, so that (i) the optimal KL-regularized policy π_β^* for rewards $r^*(x, y) = \sum_{h=1}^2 \langle \theta_h^*, \phi_h(x, a_{1:h}) \rangle$ satisfies*

$$\pi_\beta^*(x, y_{1:2}) = \pi_{\theta^*}^{\text{seq}}(x, y_{1:2}) \geq 1 - \delta > \frac{1}{2}$$

for some $y_{1:2} \in \mathcal{Y}$, yet (ii) for all $\theta_h \in \mathbb{R}^d$, $\pi_\theta^{\text{auto}}(x, y_{1:2}) \leq \frac{1}{2}$. In particular, this means there are no $\theta_h \in \mathbb{R}^d$ such that $\pi_\theta^{\text{auto}} = \pi_{\theta^}^{\text{seq}}$.*

Proof of Proposition I.1. We adapt the proof of Proposition E.2 in Huang et al. [35]. Throughout, we omit the dependence on the prompt \perp . Let $\pi_{h,\text{ref}} := \text{unif}(\mathcal{A})$. Define ϕ_1 by $\phi_1(i) = e_1$, and define ϕ_2 by:

$$\phi_2(i, j) = \begin{cases} e_1 & \text{if } i = 2, j = 1 \\ e_2 & \text{o.w.} \end{cases}.$$

For $h \in \{1, 2\}$, define $r_h^*(y_{1:h}) = \langle \theta_h^*, \phi_h(y_{1:h}) \rangle$, where $\theta_1^* = \theta_2^* = B \cdot e_i$ for $B \geq \log(3/\delta)$. Then we have

$$\pi_\beta^*(y_1 = 2, y_2 = 1) = \frac{e^{2B}}{e^{2B} + 3e^B} = \frac{1}{1 + 3e^{-B}} = \frac{1}{1 + \delta} \geq 1 - \delta.$$

On the other hand, since $\phi_1(1) = \phi_1(2) = B \cdot e_1$, all $\theta \in \mathbb{R}^d$ have $\pi_\theta^{\text{auto}}(y_1 = 2, y_2 = 1) \leq \pi_\theta^{\text{auto}}(y_1 = 2) = \frac{1}{2}$. \square

I.2. Sample Complexity, Computational Oracles, and Coverage

Recall that our goal is to design an algorithm that achieves the objective in (68) with as few episodes of interaction with the environment as possible. We measure the sample efficiency of an algorithm in terms of the total number T_{data} of reward, transition, and local simulator (reset) queries required to achieve (68) for $\varepsilon > 0$. To allow for computationally efficient algorithms, we assume access to the following sampling oracle for π_{ref} , generalizing [Definition 5.1](#).

Definition I.3 (Policy sampling oracle (weak version)). *In one query, the learner can propose a state $x \in \mathcal{X}$ and layer $h \in [H]$, and receive a conditional sample $\mathbf{a}_h \sim \pi_{h,\text{ref}}(\cdot \mid x)$ as well as the corresponding feature $\phi_h(x, \mathbf{a}_h)$. Additionally, in one query the learner can propose a state $x \in \mathcal{X}$, action $a \in \mathcal{A}$, and layer $h \in [H]$, and receive $\phi_h(x, a)$. We let T_{comp} denote the total number of policy sampling queries used by the algorithm.*

For technical reasons, we require explicit query access to $\phi_h(x, a)$ in this framework, whereas in our original framework (Definition 2.1) we only required observing $\phi(x, y)$ for sampled (x, y) pairs. This is because our multi-turn algorithm MTSS uses an “anchor action” to normalize features across all states; it may be possible to this avoid with a method similar to what is used by SpannerSampling—see Remark J.4.

This technical point aside, note that Definition I.3 is a generalization of the *weak* autoregressive sampling oracle in Definition 5.1, which instantiates Definition I.3 in the token-level MDP. An analogue of the *strong* oracle in Definition 5.1 would be to allow sampling $\mathbf{a}_h \sim \pi_{h,\theta}(\cdot | x)$ in unit time for any θ , but the weak oracle is all that is required by our main algorithm, MTSS.

Finally, our results depend on the following coverage coefficient, generalizing Eq. (13).

Definition I.4 (Conditional Coverage). *For any policy $\pi = \{\pi_h\}_{h=1}^H$ and state $x \in \mathcal{X}$, the conditional coverage of $\pi_{\text{ref}} = \{\pi_{h,\text{ref}}\}_{h=1}^H$ relative to a reference policy $\pi_{\text{ref}} = a$ at x is defined as:*

$$C_{\text{cond}}(\pi | x) := \sup_{a \in \mathcal{A}} \max_{h \in [H]} \frac{\pi_h(a | x)}{\pi_{h,\text{ref}}(a | x)}. \quad (74)$$

Similarly, the conditional coverage of π relative to π_{ref} is defined as:

$$C_{\text{cond}}(\pi) := \sup_{(x,a) \in \mathcal{X} \times \mathcal{A}} \max_{h \in [H]} \frac{\pi_h(a | x)}{\pi_{h,\text{ref}}(a | x)}. \quad (75)$$

For $h \in [H]$, we occasionally overload notation and write $C_{\text{cond}}(\pi_h)$ and $C_{\text{cond}}(\pi_h | x)$ to indicate the quantities $\sup_{a \in \mathcal{A}} \frac{\pi_h(a|x)}{\pi_{h,\text{ref}}(a|x)}$ and $\sup_{(x,a) \in \mathcal{X} \times \mathcal{A}} \frac{\pi_h(a|x)}{\pi_{h,\text{ref}}(a|x)}$, respectively; that is, the quantities in (74) and (75) without the max over h .

Our goal is to ensure $T_{\text{data}} \leq \text{poly}(d, B, H, \beta^{-1}, \varepsilon^{-1}, \log(\delta^{-1}))$ and $T_{\text{comp}} \leq \text{poly}(C_{\text{cond}}(\pi_{\beta}^*), T_{\text{data}})$, where d is the dimension of the feature map ϕ in Assumption I.1, B is the bound on the parameter norm, and $C_{\text{cond}}(\pi_{\beta}^*)$ is the coverage number of the optimal regularized policy.

Additional notation. For any $m, n \in \mathbb{N}$, we denote by $[m .. n]$ the integer interval $\{m, \dots, n\}$. We also let $[n] := [1 .. n]$. For any $(x, a) \in \mathcal{X} \times \mathcal{A}$, we use the convention that

$$\mathbb{E}[\cdot | \mathbf{x}_0 = x, \mathbf{a}_0 = a] \equiv \mathbb{E}[\cdot] \quad \text{and} \quad \mathbb{P}[\cdot | \mathbf{x}_0 = x, \mathbf{a}_0 = a] \equiv \mathbb{P}[\cdot]. \quad (76)$$

Appendix J. MTSS Algorithm and Guarantees

In this section, we formally introduce our algorithm, MTSS, present some intuition behind its design, and state its guarantee ([Theorem J.1](#)).

Algorithm 4 MTSS: Multi-Turn Spanner Sampling.

input: Base policy π_{ref} . Parameters $\beta, \delta, \varepsilon \in (0, 1)$ and $B > 0$.
initialize: Set $j \leftarrow 1$, and $\Sigma_h^1 \leftarrow \lambda I, \forall h \in [H]$.
 1: Set $T_{\text{prompt}}, N_{\text{reg}}, N_{\text{span}}, \bar{N}_{\text{span}}, M_{\text{rej}}, \delta_{\text{rej}}, \lambda, \nu > 0$ as in [Section J.1.4](#).
 2: Set $\mathcal{C}_h^1 \leftarrow \emptyset$, for all $h \in [0..H]$, and set $\mathcal{C}_0^1 \leftarrow \mathcal{C}_0^1 \cup \{(x_0, a_0)\}$ for arbitrary $(x_0, a_0) \in \mathcal{X} \times \mathcal{A}$.
 3: **for** $t = 1, \dots, T$ **do**
 / Fit state-action value function in a dynamic programming fashion. */*
 4: **for** $h = H, \dots, 1$ **do**
 5: Update $\theta_h^t \leftarrow \text{FitValue}_h(\mathcal{C}_h^t, \theta_{h+1:H}^t, \Sigma_{h+1:H}^t; B, \mathfrak{a}, N_{\text{reg}}, M_{\text{rej}}, \delta_{\text{rej}}, \pi_{\text{ref}})$.
 6: For all $x \in \mathcal{X}$, set

$$\hat{\pi}_h^t(\cdot | x) \propto \text{SoftmaxSampler}_{\beta, M_{\text{rej}}, \delta_{\text{rej}}}(\langle \bar{\varphi}_h(x, \cdot), \theta_h^t \rangle; x, \pi_{\text{ref}}),$$

 where $\bar{\varphi}_h^t(\cdot, \cdot) = \varphi_h(\cdot, \cdot) \cdot \mathbb{I} \left\{ \|\varphi_h(\cdot, \cdot)\|_{(\Sigma_h^t)^{-1}}^2 \leq \nu^2 \right\}$ and $\varphi_h(\cdot, \cdot) := \phi_h(\cdot, \cdot) - \phi_h(\cdot, \mathfrak{a})$.
 / Add uncertain state action pairs to the core set. */*
 7: Set $\mathcal{C}_{1:H}^{t+1} \leftarrow \mathcal{C}_{1:H}^t$ and $\Sigma_{1:H}^{t+1} \leftarrow \Sigma_{1:H}^t$.
 8: **for** $h = 1, \dots, H$ **do**
 9: $(x_h^t, a_h^t) \leftarrow \text{UncertainStateAction}_h(\mathcal{C}_{0:h-1}^t, \hat{\pi}_{1:h}^t, \Sigma_h^t; N_{\text{span}}, \bar{N}_{\text{span}})$. *// Algorithm 6*
 10: Update $\mathcal{C}_h^{t+1} \leftarrow \mathcal{C}_h^{t+1} \cup \{(x_h^t, a_h^t), (x_h^t, \mathfrak{a})\}$. *// \mathcal{C}_h^{t+1} is a multiset.*
 11: Update $\Sigma_h^{t+1} \leftarrow \Sigma_h^{t+1} + \varphi_h(x_h^t, a_h^t) \varphi_h(x_h^t, a_h^t)^\top$.
 / If (x_h^t, a_h^t) is not too uncertain, $\hat{\pi}^t$ is a good candidate policy to return. */*
 12: **if** $\max_{h \in [H]} \|\varphi_h(x_h^t, a_h^t)\|_{(\Sigma_h^t)^{-1}}^2 \leq \nu^2/4$ **then**
 13: $j \leftarrow t$.
 14: **return** $\hat{\pi}_{1:H} = \hat{\pi}_{1:H}^j$.

J.1. MTSS Pseudocode and Overview

Our main algorithm, MTSS ([Algorithm 4](#)), learns a policy in a dynamic programming fashion by fitting value functions for each layer $h = H, \dots, 1$, while maintaining a growing *core-set* of informative state-action pairs. This core-set guides exploration, and is closely related to recent works on linearly parameterized RL with local simulators [[51](#), [103](#), [91](#), [60](#), [59](#)]; of the works, the structure of MTSS is most closely aligned with the Optimistic-PSDP algorithm introduced by [[59](#)] for the linearly- Q^π realizable RL setting, which itself builds on ideas from the classical Policy Search by Dynamic Programming (PSDP) algorithm (see, e.g., [[4](#)]) and the Recursive Value Function Search (RVFS) algorithm of Mhammedi et al. [[60](#)]. We combine this with the spanner technique and truncated

softmax policies from SpannerSampling, which are critical to achieve computational efficiency under the sampling oracle model in Definition I.3. At various points in the section, we will comment on key similarities and differences between these approaches.

MTSS comprises three main subroutines: FitValue (Algorithm 5) and UncertainStateAction (Algorithm 6). Before getting into the details of MTSS, we first provide an overview of the key variables used in Algorithm 4.

J.1.1. KEY VARIABLES IN MTSS (ALGORITHM 4)

MTSS takes a fixed reference policy π_{ref} as input (e.g., a pre-trained language model) and runs for $T_{\text{prompt}} = \tilde{O}(d)$ iterations. At each iteration $t \in [T_{\text{prompt}}]$, the algorithm maintains the following key variables:

- \mathcal{C}_h^t : A core-set of t state-action pairs at layer h . The subroutine FitValue_h uses these state-action pairs as starting points to generate rollout trajectories, which are then used to fit the parameters of the optimal policy via regression. The core-set \mathcal{C}_h^t —generalizing the notion of spanner in SpannerSampling—aims at provide sufficient coverage of the state-action space at layer h . Note that \mathcal{C}_h^t is a *multiset*, allowing for multiple instances of the same state-action pair.
- \mathbf{a} : A fixed, arbitrarily chosen “anchor” action used in the regression problem solved by FitValue.
- Σ_h^t : A “design matrix” for layer h , defined as the sum of outer products of feature differences $\varphi_h(x_h, a_h) := \phi_h(x_h, a_h) - \phi_h(x_h, \mathbf{a})$ for the core-set states-action pairs $(x_h, a_h) \in \mathcal{C}_h^t$:

$$\Sigma_h^t = \lambda I + \sum_{(x_h, a_h) \in \mathcal{C}_h^t} \varphi_h(x_h, a_h) \varphi_h(x_h, a_h)^\top, \quad (77)$$

where $\lambda > 0$ is a regularization parameter defined in Algorithm 4.

- θ_h^t : An estimate of the parameter vector $\theta_{h,\beta}^*$ associated with $\pi_{h,\beta}^*$ (Assumption I.1).
- $\hat{\pi}_h^t$: The policy used to generate actions at layer h during iteration t . This policy is computed using the SoftmaxSamplerDensity subroutine (Algorithm 2 in Appendix F) and approximates the “truncated” policy:

$$\bar{\pi}_h^t(\cdot | x) \propto \pi_{\text{ref}}(\cdot | x) \cdot e^{\beta^{-1} \cdot \langle \bar{\varphi}_h^t(x, \cdot), \theta_h^t \rangle}, \quad \text{where} \quad \bar{\varphi}_h^t(\cdot, \cdot) = \varphi_h(\cdot, \cdot) \cdot \mathbb{I} \left\{ \|\varphi_h(\cdot, \cdot)\|_{(\Sigma_h^t)^{-1}}^2 \leq \nu^2 \right\}; \quad (78)$$

this notion generalizes the truncated softmax policies used in SpannerSampling. As t increases, θ_h^t converges to $\theta_{h,\beta}^*$, and $\|\varphi_h(x, a)\|_{(\Sigma_h^t)^{-1}}^2$ decreases for all (x, a) pairs. Eventually, $\|\varphi_h(x, a)\|_{(\Sigma_h^t)^{-1}}^2$ becomes smaller than ν^2 for “most” state-action pairs (x, a) , ensuring that $\bar{\pi}_h^t$ (and thus $\hat{\pi}_h^t$) approximates the optimal policy $\pi_{h,\beta}^*$.

In each iteration $t \in [T_{\text{prompt}}]$, MTSS computes the estimates $\theta_{1:H}^t$ in a dynamic programming fashion by fitting the difference of value functions $Q_{\hat{\pi}_h^t, \beta}^t(\cdot, \cdot) - Q_{\hat{\pi}_h^t, \beta}^t(\cdot, \mathbf{a})$ at each layer h (motivated by Eq. (73)). In what follows, we describe the FitValue subroutine responsible for this step.

Algorithm 5 FitValue_h: Estimate KL-regularized value function using rollouts.

input: Layer h , core set \mathcal{C}_h , $\theta_{h+1:H}$, $\Sigma_{h+1:H}$, $B > 0$, fixed action \mathfrak{a} , and N , M , $\tilde{\delta}$, π_{ref} .

1: For all $\ell \in [h + 1 .. H]$ and $x \in \mathcal{X}$, define

$$\bar{\varphi}_\ell(\cdot, \cdot) = \varphi_\ell(\cdot, \cdot) \cdot \mathbb{I} \left\{ \|\varphi_\ell(\cdot, \cdot)\|_{\Sigma_\ell^{-1}}^2 \leq \nu^2 \right\},$$

where $\varphi_\ell(\cdot, \cdot) := \phi_\ell(\cdot, \cdot) - \phi_\ell(\cdot, \mathfrak{a})$.

/ Gather trajectory data. */*

2: **for** $(x_h, a_h) \in \mathcal{C}_h$ **do**

3: Set $\mathcal{D}(x_h, a_h) \leftarrow \emptyset$.

4: Set $\mathbf{x}'_h = \mathbf{x}''_h = x_h$, $\mathbf{a}'_h = a_h$, and $\mathbf{a}''_h = \mathfrak{a}$.

5: Set $\mathbf{r}'_h \sim r_h^*(\mathbf{x}'_h, \mathbf{a}'_h)$ and $\mathbf{r}''_h \sim r_h^*(\mathbf{x}''_h, \mathbf{a}''_h)$.

6: **for** N times **do**

7: **for** $\ell = h + 1, \dots, H$ **do**

8: Sample $\mathbf{x}'_\ell \sim \mathbb{P}[\cdot \mid \mathbf{x}_{\ell-1} = \mathbf{x}'_{\ell-1}, \mathbf{a}_{\ell-1} = \mathbf{a}'_{\ell-1}]$.

9: Sample $\mathbf{x}''_\ell \sim \mathbb{P}[\cdot \mid \mathbf{x}_{\ell-1} = \mathbf{x}''_{\ell-1}, \mathbf{a}_{\ell-1} = \mathbf{a}''_{\ell-1}]$.

10: Set $(\mathbf{a}'_\ell, \boldsymbol{\rho}'_\ell) \leftarrow \text{SoftmaxSamplerDensity}_{\beta, M, \tilde{\delta}}(\langle \bar{\varphi}_\ell(\mathbf{x}'_\ell, \cdot), \theta_\ell \rangle; \mathbf{x}'_\ell, \pi_{\text{ref}})$. *//*

Algorithm 2.

11: Set $(\mathbf{a}''_\ell, \boldsymbol{\rho}''_\ell) \leftarrow \text{SoftmaxSamplerDensity}_{\beta, M, \tilde{\delta}}(\langle \bar{\varphi}_\ell(\mathbf{x}''_\ell, \cdot), \theta_\ell \rangle; \mathbf{x}''_\ell, \pi_{\text{ref}})$.

12: Set $\mathbf{r}'_\ell \sim r_\ell^*(\mathbf{x}'_\ell, \mathbf{a}'_\ell)$ and $\mathbf{r}''_\ell \sim r_\ell^*(\mathbf{x}''_\ell, \mathbf{a}''_\ell)$.

13: Compute $\mathbf{y}'_h \leftarrow \mathbf{r}'_h + \sum_{\ell=h+1}^H (\mathbf{r}'_\ell - \beta \log \boldsymbol{\rho}'_\ell)$. *//* $\boldsymbol{\rho}'_\ell \approx \frac{\bar{\pi}_{\ell, \theta}(\mathbf{y}'_\ell | \mathbf{x}'_\ell)}{\pi_{\ell, \text{ref}}(\mathbf{y}'_\ell | \mathbf{x}'_\ell)}$.

14: Compute $\mathbf{y}''_h \leftarrow \mathbf{r}''_h + \sum_{\ell=h+1}^H (\mathbf{r}''_\ell - \beta \log \boldsymbol{\rho}''_\ell)$. *//* $\boldsymbol{\rho}''_\ell \approx \frac{\bar{\pi}_{\ell, \theta}(\mathbf{y}''_\ell | \mathbf{x}''_\ell)}{\pi_{\ell, \text{ref}}(\mathbf{y}''_\ell | \mathbf{x}''_\ell)}$.

15: Set $\mathbf{z}_h \leftarrow \mathbf{y}'_h - \mathbf{y}''_h$.

16: Update $\mathcal{D}(x_h, a_h) \leftarrow \mathcal{D}(x_h, a_h) \cup \{\mathbf{z}_h\}$. *//* \mathcal{D} is a multiset.

/ Fit value function. */*

17: **if** $\mathcal{D} \neq \emptyset$ **then** $\hat{\theta}_h \leftarrow \arg \min_{\tilde{\theta} \in \mathbb{B}(B)} \sum_{(x_h, a_h) \in \mathcal{C}_h} \sum_{\mathbf{z}_h \in \mathcal{D}(x_h, a_h)} \left(\langle \varphi_h(x_h, a_h), \tilde{\theta} \rangle - z_h \right)^2$, **else**
 $\hat{\theta}_h \leftarrow 0$.

18: **return** $\hat{\theta}_h$.

J.1.2. FitValue (ALGORITHM 5)

In each iteration $t \in [T_{\text{prompt}}]$, starting from $h = H$ and progressing down to $h = 1$, MTSS invokes FitValue_h with the input $(\mathcal{C}_h^t, \theta_{h+1:H}^t)$. This subroutine returns the vector θ_h^t , an estimate of the parameter vector $\theta_{h,\beta}^*$ for the optimal policy $\pi_{h,\beta}^*$ (see Assumption I.1). To compute the estimate θ_h^t , for each state-action pair $(x_h, a_h) \in \mathcal{C}_h^t$, FitValue generates multiple i.i.d. regression targets \mathbf{z}_h by sampling two trajectories $(\mathbf{x}'_h, \mathbf{a}'_h, \mathbf{r}'_h, \boldsymbol{\rho}'_h, \dots, \mathbf{x}'_H, \mathbf{a}'_H, \mathbf{r}'_H, \boldsymbol{\rho}'_H)$ and $(\mathbf{x}''_h, \mathbf{a}''_h, \mathbf{r}''_h, \boldsymbol{\rho}''_h, \dots, \mathbf{x}''_H, \mathbf{a}''_H, \mathbf{r}''_H, \boldsymbol{\rho}''_H)$ via the following process. Initialize $\mathbf{x}'_h = \mathbf{x}''_h = x_h$, $\mathbf{a}'_h = a_h$, and $\mathbf{a}''_h = \mathfrak{a}$ (recall that \mathfrak{a} is a fixed, arbitrary action defined in Algorithm 4), and sample $\mathbf{r}'_h \sim r_h^*(\mathbf{x}'_h, \mathbf{a}'_h)$ and $\mathbf{r}''_h \sim r_h^*(\mathbf{x}''_h, \mathbf{a}''_h)$. Then,

for $\ell = h+1, \dots, H$, use `SoftmaxSamplerDensity` (Algorithm 2 in Appendix F) to approximately sample from the policy $\bar{\pi}^t$ as follows:

- Sample $\mathbf{x}'_\ell \sim \mathbb{P}[\cdot \mid \mathbf{x}_{\ell-1} = \mathbf{x}'_{\ell-1}, \mathbf{a}_{\ell-1} = \mathbf{a}'_{\ell-1}]$ and $\mathbf{x}''_\ell \sim \mathbb{P}[\cdot \mid \mathbf{x}_{\ell-1} = \mathbf{x}''_{\ell-1}, \mathbf{a}_{\ell-1} = \mathbf{a}''_{\ell-1}]$;
- Set $(\mathbf{a}'_\ell, \boldsymbol{\rho}'_\ell) \leftarrow \text{SoftmaxSamplerDensity}_{\beta, M_{\text{rej}}, \delta_{\text{rej}}}(\langle \bar{\varphi}_\ell^t(\mathbf{x}'_\ell, \cdot), \theta_\ell^t \rangle; \mathbf{x}'_\ell, \pi_{\text{ref}})$;
- Set $(\mathbf{a}''_\ell, \boldsymbol{\rho}''_\ell) \leftarrow \text{SoftmaxSamplerDensity}_{\beta, M_{\text{rej}}, \delta_{\text{rej}}}(\langle \bar{\varphi}_\ell^t(\mathbf{x}''_\ell, \cdot), \theta_\ell^t \rangle; \mathbf{x}''_\ell, \pi_{\text{ref}})$;
- Sample rewards $r'_\ell \sim r_\ell^*(\mathbf{x}'_\ell, \mathbf{a}'_\ell)$ and $r''_\ell \sim r_\ell^*(\mathbf{x}''_\ell, \mathbf{a}''_\ell)$;

then, finally, set

$$\mathbf{z}_h = \mathbf{r}'_h + \sum_{\ell=h+1}^H (\mathbf{r}'_\ell - \beta \log \boldsymbol{\rho}'_\ell) - \mathbf{r}''_h - \sum_{\ell=h+1}^H (\mathbf{r}''_\ell - \beta \log \boldsymbol{\rho}''_\ell).$$

Here, $\boldsymbol{\rho}'_\ell$ and $\boldsymbol{\rho}''_\ell$ approximate $\frac{\bar{\pi}_\ell^t(\mathbf{y}'_\ell | \mathbf{x}'_\ell)}{\pi_{\ell, \text{ref}}(\mathbf{y}'_\ell | \mathbf{x}'_\ell)}$ and $\frac{\bar{\pi}_\ell^t(\mathbf{y}''_\ell | \mathbf{x}''_\ell)}{\pi_{\ell, \text{ref}}(\mathbf{y}''_\ell | \mathbf{x}''_\ell)}$, respectively, and the expected value of \mathbf{z}_h (up to small approximation error) corresponds to the difference

$$Q_{h, \beta}^{\hat{\pi}^t}(x_h, a_h) - Q_{h, \beta}^{\hat{\pi}^t}(x_h, \mathbf{a}).$$

We regress onto \mathbf{z}_h with least squares, setting the vector θ_h^t as the minimizer of the sum of squared errors across all $(x_h, a_h) \in \mathcal{C}_h^t$.

Note that if we could somehow ensure $\hat{\pi}^t = \pi^*$, the difference $Q_{h, \beta}^{\hat{\pi}^t}(x_h, a_h) - Q_{h, \beta}^{\hat{\pi}^t}(x_h, \mathbf{a})$ would be linear in $\varphi_h(x_h, a_h)$ by Eq. (73), but it is not guaranteed to be linear in general; this poses challenges for deriving a regression guarantee, as the regression problem may not be realizable or even approximately realizable. Fortunately, the components of MTSS (in particular `UncertainStateAction`) ensure that for sufficiently large T_{prompt} , there exists an iteration $t \in [T_{\text{prompt}}]$ such that for all $h \in [H]$ and state-action pairs (x_h, a_h) satisfying $\|\varphi_h(x_h, a_h)\|_{(\Sigma_h^t)^{-1}}^2 \leq O(1)$, the following quantity is small:

$$\left| Q_{h, \beta}^*(x_h, a_h) - Q_{h, \beta}^*(x_h, \mathbf{a}) - Q_{h, \beta}^{\hat{\pi}^t}(x_h, a_h) + Q_{h, \beta}^{\hat{\pi}^t}(x_h, \mathbf{a}) \right|. \quad (79)$$

Consequently, for such an iteration t , the regression problem becomes approximately linearly realizable. This allows us to establish that for sufficiently large T_{prompt} , there exists an iteration $t \in [T_{\text{prompt}}]$ such that the subroutine `FitValueh` returns θ_h^t satisfying:

$$\|\theta_h^t - \theta_{h, \beta}^*\|_{\Sigma_h^t}^2 = \lambda I + \sum_{(x_h, a_h) \in \mathcal{C}_h^t} \langle \varphi_h(x_h, a_h), \theta_h^t - \theta_{h, \beta}^* \rangle^2 \leq \varepsilon_{\text{reg}}^2, \quad (80)$$

with high probability, for some small $\varepsilon_{\text{reg}} > 0$.

Remark J.1 (Fitting the difference). *The reason `FitValue` targets the difference $Q_{h, \beta}^*(\cdot, \cdot) - Q_{h, \beta}^*(\cdot, \mathbf{a})$ rather than $Q_{h, \beta}^*(\cdot, \cdot)$ directly is that the former (but not the latter) is linear in $\phi_h(\cdot, \cdot) - \phi_h(\cdot, \mathbf{a})$, as guaranteed by the softmax-linear assumption in Assumption I.1 and Lemma G.1. Without additional assumptions, the same would not hold for the un-centered value $Q_{h, \beta}^*(\cdot, \cdot)$.*

The Optimistic-PSDP algorithm of Mhammedi [59] uses a subroutine similar to `FitValue`. The difference is that Optimistic-PSDP fits *optimistic* value functions, whereas `FitValue` does not. Optimism is not needed in our setting because we can effectively drive exploration using the core-sets under reset/local simulator access.

J.1.3. UncertainStateAction (ALGORITHM 6)

Algorithm 6 UncertainStateAction_h: Identify uncertain state-action pair.

input: $h, \mathcal{C}_{0:h-1}, \hat{\pi}_{1:h}, \Sigma_h, \mathfrak{a}, N, \bar{N}$.
 /* Gathering candidate state action pairs. */
 1: Set $\kappa \leftarrow 0$ and define $\varphi_h(\cdot, \cdot) = \phi_h(\cdot, \cdot) - \phi_h(\cdot, \mathfrak{a})$.
 2: **for** $\ell = 0, \dots, h-1$ **do**
 3: **for** $(x_\ell, a_\ell) \in \mathcal{C}_\ell$ **do**
 4: Initialize $\mathcal{D}_\ell(x_\ell, a_\ell) \leftarrow \emptyset$. // \mathcal{D}_ℓ is a multiset (only used in the analysis).
 5: **for** N times **do**
 6: Sample $(\mathbf{x}_{\ell+1}, \mathbf{a}_{\ell+1}, \dots, \mathbf{x}_h, \mathbf{a}_h) \sim \mathbb{P}_{\hat{\pi}_{\ell+1:h}}[\cdot \mid \mathbf{x}_\ell = x_\ell, \mathbf{a}_\ell = a_\ell]$.
 7: // Above, we use the convention that $\mathbb{P}_{\hat{\pi}_{\ell+1:h}}[\cdot \mid \mathbf{x}_0 = x_0, \mathbf{a}_0 = a_0] \equiv \mathbb{P}_{\hat{\pi}_{\ell+1:h}}[\cdot]$.
 8: Update $\mathcal{D}_\ell(x_\ell, a_\ell) \leftarrow \mathcal{D}_\ell(x_\ell, a_\ell) \cup \{(\mathbf{x}_h, \mathbf{a}_h)\}$.
 9: **if** $\|\varphi_h(\mathbf{x}_h, \mathbf{a}_h)\|_{\Sigma_h^{-1}} > \kappa$ **then**
 10: Set $\kappa \leftarrow \|\varphi_h(\mathbf{x}_h, \mathbf{a}_h)\|_{\Sigma_h^{-1}}$.
 11: Set $(\hat{x}_h, \hat{a}_h) \leftarrow (\mathbf{x}_h, \mathbf{a}_h)$.
 12: /* Testing \bar{N} samples from π_{ref} given \mathbf{x}_h */
 13: Initialize $\bar{\mathcal{D}}_\ell(\mathbf{x}_h)$. // $\bar{\mathcal{D}}_\ell$ is a multiset (only used in the analysis).
 14: **for** \bar{N} times **do**
 15: Sample $\bar{\mathbf{a}}_h \sim \pi_{h,\text{ref}}(\cdot \mid \mathbf{x}_h)$.
 16: Update $\bar{\mathcal{D}}_\ell(\mathbf{x}_h) \leftarrow \bar{\mathcal{D}}_\ell(\mathbf{x}_h) \cup \{\bar{\mathbf{a}}_h\}$.
 17: **if** $\|\varphi_h(\mathbf{x}_h, \bar{\mathbf{a}}_h)\|_{\Sigma_h^{-1}} > \kappa$ **then**
 18: Set $\kappa \leftarrow \|\varphi_h(\mathbf{x}_h, \bar{\mathbf{a}}_h)\|_{\Sigma_h^{-1}}$.
 19: Set $(\hat{x}_h, \hat{a}_h) \leftarrow (\mathbf{x}_h, \bar{\mathbf{a}}_h)$.
 20: **return** (\hat{x}_h, \hat{a}_h) .

MTSS uses the subroutine UncertainStateAction to update the core-sets (\mathcal{C}_h^t) . When MTSS invokes it with the input $(\mathcal{C}_{0:h-1}^t, \hat{\pi}_{1:h}^t, \Sigma_{1:h}^t)$, UncertainStateAction_h uses $\hat{\pi}_{1:h}^t$ to generate multiple partial trajectories starting from the state-action pairs $(x_\ell, a_\ell) \in \mathcal{C}_\ell^t$ at all layers $\ell \in [0..h-1]$ and terminating at layer h . Among all state-action pairs reached at layer h through this process, UncertainStateAction selects the pair (x_h^t, a_h^t) that maximizes the elliptical objective $\|\varphi_h(\cdot, \cdot)\|_{(\Sigma_h^t)^{-1}}$. As a result, the output (x_h^t, a_h^t) of UncertainStateAction satisfies the following property: with high probability, for all $\ell \in [0..h-1]$ and $(x_\ell, a_\ell) \in \mathcal{C}_\ell^t$,

$$\mathbb{P}_{\hat{\pi}_{\ell+1:h}^t} \left[\|\varphi_h(\mathbf{x}_h, \mathbf{a}_h)\|_{(\Sigma_h^t)^{-1}}^2 > \nu^2 \vee \left(2\|\varphi_h(x_h^t, a_h^t)\|_{(\Sigma_h^t)^{-1}}^2 \mid \mathbf{x}_\ell = x_\ell, \mathbf{a}_\ell = a_\ell \right) \right] \leq \varepsilon_{\text{span}}, \quad (81)$$

for some small $\varepsilon_{\text{span}} > 0$; this generalizes the spanner property used in SpannerSampling.

Using the definition of Σ_h^t in (77), a standard elliptical potential argument (see the proof of Lemma M.1) implies that for sufficiently large $t = \Omega(d)$ in MTSS, the tuple (x_h^t, a_h^t) returned by UncertainStateAction must satisfy $\|\varphi_h(x_h^t, a_h^t)\|_{(\Sigma_h^t)^{-1}} \leq \nu^2$. Substituting this into (81) ensures

that for such t , the following holds for all $\ell \in [0 .. h - 1]$ and $(x_\ell, a_\ell) \in \mathcal{C}_\ell^t$:

$$\mathbb{P}_{\hat{\pi}_{\ell+1:h}^t} \left[\|\varphi_h(\mathbf{x}_h, \mathbf{a}_h)\|_{(\Sigma_h^t)^{-1}}^2 > \nu^2 \mid \mathbf{x}_\ell = x_\ell, \mathbf{a}_\ell = a_\ell \right] \leq \varepsilon_{\text{span}}. \quad (82)$$

This property is crucial in the analysis of FitValue_ℓ , as it allows us to bound the misspecification error for the regression problems solved in FitValue (cf. Eq. (79)) at future layers $h \in [\ell + 1 .. H]$. Concretely, using Hölder's inequality, we have for all $(x_\ell, a_\ell) \in \mathcal{C}_\ell^t$,

$$\begin{aligned} & \mathbb{E}_{\hat{\pi}_{\ell+1:h}^t} \left[\langle \varphi_h(\mathbf{x}_h, \mathbf{a}_h), \theta_h^t - \theta_{h,\beta}^* \rangle^2 \mid \mathbf{x}_\ell = x_\ell, \mathbf{a}_\ell = a_\ell \right] \\ & \leq \mathbb{E}_{\hat{\pi}_{\ell+1:h}^t} \left[\|\varphi_h(\mathbf{x}_h, \mathbf{a}_h)\|_{(\Sigma_h^t)^{-1}}^2 \mid \mathbf{x}_\ell = x_\ell, \mathbf{a}_\ell = a_\ell \right] \cdot \|\theta_h^t - \theta_{h,\beta}^*\|_{\Sigma_h^t}^2, \\ & \leq \left(\nu^2 + \lambda^{-1} \mathbb{P}_{\hat{\pi}_{\ell+1:h}^t} \left[\|\varphi_h(\mathbf{x}_h, \mathbf{a}_h)\|_{(\Sigma_h^t)^{-1}}^2 \mid \mathbf{x}_\ell = x_\ell, \mathbf{a}_\ell = a_\ell \right] \right) \cdot \|\theta_h^t - \theta_{h,\beta}^*\|_{\Sigma_h^t}^2, \\ & \leq (\nu^2 + \lambda^{-1} \varepsilon_{\text{span}}) \cdot \|\theta_h^t - \theta_{h,\beta}^*\|_{\Sigma_h^t}^2. \end{aligned} \quad (83)$$

where the last inequality follows from $\sigma_{\min}(\Sigma_h^t) \geq \lambda$ and Assumption I.1. This bound is particularly useful as it allows us to control the “on-policy” error:

$$\mathbb{E}_{\hat{\pi}_{\ell+1:h}^t} \left[\langle \varphi_h(\mathbf{x}_h, \mathbf{a}_h), \theta_h^t - \theta_{h,\beta}^* \rangle^2 \mid \mathbf{x}_\ell = x_\ell, \mathbf{a}_\ell = a_\ell \right],$$

for all state-action pairs $(x_\ell, a_\ell) \in \mathcal{C}_\ell^t$, in terms of the regression error at layer h .

Remark J.2. Optimistic-PSDP [59] features a subroutine similar to `UncertainStateAction`. The key difference is that Optimistic-PSDP uses a core-set of policies rather than state-action pairs. Consequently, its corresponding subroutine greedily selects the policy that maximizes the elliptical objective $\|\phi_h(\mathbf{x}_h, \mathbf{a}_h)\|_{(\Sigma_h^t)^{-1}}$, evaluated in expectation over the core-set of policies and the algorithm's current policy $\hat{\pi}_h^t$.

J.1.4. PARAMETER CHOICES FOR MTSS

For $\mathfrak{c} = \text{polylog}(d, C_{\text{cond}}(\pi_\beta^*), 1/\delta, 1/\varepsilon, H, B)$ sufficiently large, we set the parameters in MTSS as:

$$\begin{aligned} \varepsilon_{\text{reg}}^2 &= \varepsilon, \quad \nu = 1/2, \quad T_{\text{prompt}} = dH^2\mathfrak{c}, \quad \lambda = \frac{\varepsilon_{\text{reg}}^2}{\mathfrak{c}B^2}, \quad M_{\text{rej}} = \frac{C_{\text{cond}}(\pi_\beta^*)^2 T_{\text{prompt}}^2 H^2 B^4 \mathfrak{c}}{\varepsilon_{\text{reg}}^2}, \\ \delta_{\text{rej}} &= \frac{\varepsilon_{\text{reg}}^2}{H^2 B^3 T_{\text{prompt}} \mathfrak{c}}, \quad \bar{N}_{\text{span}} = \frac{T_{\text{prompt}} H^4 B^4 C_{\text{cond}}(\pi_\beta^*) \mathfrak{c}}{\varepsilon_{\text{reg}}^2}, \quad N_{\text{span}} = \frac{T_{\text{prompt}} H^2 B^4 \mathfrak{c}}{\varepsilon_{\text{reg}}^2}, \\ N_{\text{reg}} &= \frac{H^2 B^4 d T_{\text{prompt}} \mathfrak{c}}{\varepsilon_{\text{reg}}^2}. \end{aligned}$$

J.2. Main Guarantee for MTSS (Generalization of Theorem 5.1)

Building on the intuition in the prequel, the main guarantee for MTSS is as follows.

Theorem J.1 (Main guarantee for MTSS). *Let $\beta, \varepsilon, \delta \in (0, 1)$, $B > 0$, and π_{ref} be such that $\varepsilon \leq \beta^2/4$ and suppose [Assumption I.1](#) holds with $B > 0$. Then, the policy $\hat{\pi}$ returned by MTSS($\beta, \delta, \varepsilon, B, \pi_{\text{ref}}$) ([Algorithm 4](#)) satisfies*

$$J_\beta(\pi_\beta^*) - J_\beta(\hat{\pi}) \leq \varepsilon.$$

Furthermore, the algorithm requires $T_{\text{data}}(\varepsilon, \delta) \leq \text{poly}(d, B, H, \beta^{-1}, \varepsilon^{-1} \log(1/\delta))$ reward, transition, and local simulator queries and $T_{\text{comp}}(\varepsilon, \delta) \leq \text{poly}\left(C_{\text{cond}}(\pi_\beta^*), T_{\text{data}}(\varepsilon, \delta)\right)$ runtime and sampling oracle queries.

Critically, we see that the sample complexity $T_{\text{comp}}(\varepsilon, \delta)$ is polynomial in all of the relevant problem parameters, and the runtime and oracle complexity $T_{\text{comp}}(\varepsilon, \delta)$ scales with the action-level coverage coefficient $\max_{h \in [H]} C_{\text{cond}}(\pi_{h,\beta}^*)$. [Theorem 5.1](#) is an immediate corollary. Overall, the polynomial dependence on other problem parameters is significantly worse than that of SpannerSampling; this can likely be tightened with more effort, but we leave this for future work.

The remainder of [Part III](#) is dedicated to proving [Theorem J.1](#). The main idea behind the proof is to show that after a sufficient number of iterations, the linear regression problems solved by FitValue become approximately realizable, in the sense that the error

$$\left| Q_{h,\beta}^*(x_h, a_h) - Q_{h,\beta}^*(x_h, \mathfrak{a}) - Q_h^{\hat{\pi}^t}(x_h, a_h) + Q_h^{\hat{\pi}^t}(x_h, \mathfrak{a}) \right|$$

is small on average. For this, the key challenge is to show that misspecification errors propagate favorably across the layers $h \in [H]$, avoiding the dreaded *error amplification* phenomenon [\[86\]](#) in which misspecification errors compound exponentially. For this, our main insight is that the regularization parameter β allows for benign error propagation, with errors at layer $h+1, \dots, H$ only having a higher-order impact on the misspecification at layer h . This shows that regularization enables statistical tractability in a similar way to the assumption of an action gap Δ found in prior work on linearly-realizable Q^* [\[90, 51, 60\]](#), an observation which we expect to find broader use.

We emphasize that while MTSS draws inspiration from prior work on the linear- Q^* problem and relatives—particularly Mhammedi et al. [\[60\]](#), Mhammedi [\[59\]](#)—it requires fairly substantial modifications, both in design and analysis—to (i) leverage KL regularization, and (ii) achieve computational efficiency in the sampling oracle framework.

Remark J.3 (Action-level coverage). *One can always choose $\pi_{h,\text{ref}}(\cdot | x) = \text{unif}(\mathcal{A})$, so $\max_{h \in [H]} C_{\text{cond}}(\pi_{h,\beta}^*) \leq |\mathcal{A}|$. For the token-level MDP formulation described in [Appendix I](#), where actions correspond to tokens, this bound may be reasonable (though likely pessimistic). However, for the multi-turn language modeling formulation (Xiong et al. [96]) where each action \mathbf{a}_h represents a sub-sequence of tokens rather than a single token (e.g., corresponding to a portion of a proof), paying for $|\mathcal{A}|$ is unacceptable, and access to a base policy with good coverage is crucial.*

Remark J.4 (On the anchor action). *As discussed in [Appendix I.2](#), the sampling oracle used in MTSS ([Definition I.3](#)) goes slightly outside of the sampling oracle definition in [Definition 5.1](#) by assuming that we can query the features $\phi_h(\mathbf{x}_h, \mathbf{a})$ at an arbitrary fixed anchor action \mathbf{a} for all of the states \mathbf{x}_h encountered by the algorithm. We use the anchor action \mathbf{a} to regress onto differences in regularized rewards, which is motivated by the fact that the difference $Q_{h,\beta}^*(x, a) - Q_{h,\beta}^*(x, \mathbf{a})$ is linear (per [Eq. \(73\)](#)), while $Q_{h,\beta}^*(x, a)$ itself may not be. This assumption of access to $\phi_h(\mathbf{x}_h, \mathbf{a})$ can be avoided by incorporating “pairwise” truncated policies $\bar{\pi}_\theta$ of the type used in SpannerSampling (see [Eq. \(9\)](#)), which can be thought of as sampling a fresh anchor action $\mathbf{a}_h \sim \pi_{h,\text{ref}}(\cdot | \mathbf{x}_h)$ for each state the algorithm encounters. We opt to use the fixed anchor approach—in spite of requiring a slightly stronger oracle—to keep presentation as simple as possible, as the analysis of MTSS is already quite involved. We mention in passing that the anchor action assumption can also be removed if we directly assume that $Q_{h,\beta}^*(x, a)$ is linear, by regressing onto absolute rewards.*

Appendix K. Guarantee for UncertainStateAction

In this section, we present the main guarantee of UncertainStateAction (Algorithm 6) as a standalone algorithm; see Lemma K.1. Then, in Lemma K.2, we provide its guarantee when used as a subroutine within MTSS (Algorithm 4). For a discussion of the motivation for these results, we refer back to Section J.1.3.

Lemma K.1. *Consider a call to $\text{UncertainStateAction}_h(\mathcal{C}_{0:h-1}, \hat{\pi}_{1:h}, \Sigma_h; \mathfrak{a}, N, \bar{N})$ (Algorithm 6) for some given $h, \mathcal{C}_{0:h-1}, \hat{\pi}_{1:h}, \Sigma_h, \mathfrak{a} \in \mathcal{A}$, N , and \bar{N} such that $\sigma_{\min}(\Sigma_h) \geq \lambda$, for some $\lambda \in (0, 1)$. Then, for any $\delta' \in (0, 1)$ and $\zeta \in (0, 1/2)$, with probability at least $1 - \delta'$, the output (\hat{x}_h, \hat{a}_h) of UncertainStateAction satisfies:*

- For all $\ell \in [0..h-1]$ and $(x_\ell, a_\ell) \in \mathcal{C}_\ell$,

$$\begin{aligned} & \mathbb{P}_{\hat{\pi}_{\ell+1:h}} \left[\|\varphi_h(\mathbf{x}_h, \mathbf{a}_h)\|_{\Sigma_h^{-1}}^2 > 2 \left(\zeta \vee \|\varphi_h(\hat{x}_h, \hat{a}_h)\|_{\Sigma_h^{-1}}^2 \right) \mid \mathbf{x}_\ell = x_\ell, \mathbf{a}_\ell = a_\ell \right] \\ & \leq \max_{h \in [H]} \frac{4 \log \left(\frac{16H|\mathcal{C}_h|}{\lambda\delta'\zeta} \right)}{N}, \end{aligned} \quad (84)$$

where $\varphi_h(\cdot, \cdot) := \phi_h(\cdot, \cdot) - \phi_h(\cdot, \mathfrak{a})$.

- Furthermore, there exists $\mathcal{X}_{h,\text{span}} \subseteq \mathcal{X}$ such that for all $\ell \in [0..h-1]$ and $(x_\ell, a_\ell) \in \mathcal{C}_\ell$, $\mathbb{P}_{\hat{\pi}_{\ell+1:h-1}}[\mathbf{x}_h \in \mathcal{X}_{h,\text{span}} \mid \mathbf{x}_\ell = x_\ell, \mathbf{a}_\ell = a_\ell] \geq 1 - \max_{h \in [H]} \frac{4}{N} \log \frac{32HN|\mathcal{C}_h|}{\lambda\delta'\zeta}$ and for all $x_h \in \mathcal{X}_{h,\text{span}}$:

$$\mathbb{P}_{\mathbf{a} \sim \pi_{h,\text{ref}}(\cdot | x_h)} \left[\|\varphi_h(x_h, \mathbf{a})\|_{\Sigma_h^{-1}}^2 > 2 \left(\zeta \vee \|\varphi_h(\hat{x}_h, \hat{a}_h)\|_{\Sigma_h^{-1}}^2 \right) \right] \leq \max_{h \in [H]} \frac{4 \log \left(\frac{16H|\mathcal{C}_h|}{\lambda\delta'\zeta} \right)}{\bar{N}}. \quad (85)$$

Proof of Lemma K.1. Fix $\delta' \in (0, 1)$ and $\zeta \in (0, 1/2)$, and let $\Gamma := \{\zeta, 2\zeta, \dots, \lceil \frac{4}{\zeta\lambda} \rceil \zeta\}$. Further, for $\ell \in [0..h-1]$ and $(x_\ell, a_\ell) \in \mathcal{C}_\ell$, let $\mathcal{D}_\ell(x_\ell, a_\ell)$ be the dataset in Algorithm 6 when the algorithm returns. Note that $\mathcal{D}_\ell(x_\ell, a_\ell)$ consists of N i.i.d. pairs sampled from $\mathbb{P}_{\hat{\pi}_{\ell+1:h}}[(\mathbf{x}_h, \mathbf{a}_h) = \cdot \mid \mathbf{x}_\ell = x_\ell, \mathbf{a}_\ell = a_\ell]$. Thus, by Freedman's inequality (Lemma D.2) and the union bound over $\ell \in [0..h-1]$, $(x_\ell, a_\ell) \in \mathcal{C}_\ell$, and $\gamma \in \Gamma$, there is an event \mathcal{E} of probability at least $1 - \delta'/2$ under which, for all $\ell \in [0..h-1]$, $(x_\ell, a_\ell) \in \mathcal{C}_\ell$, and $\gamma \in \Gamma$:

$$\begin{aligned} & \mathbb{P}_{\hat{\pi}_{\ell+1:h}} \left[\|\varphi_h(\mathbf{x}_h, \mathbf{a}_h)\|_{\Sigma_h^{-1}}^2 \geq \gamma \mid \mathbf{x}_\ell = x_\ell, \mathbf{a}_\ell = a_\ell \right] \\ & \leq \frac{2}{N} \sum_{(x,a) \in \mathcal{D}_\ell(x_\ell, a_\ell)} \mathbb{I} \left\{ \|\varphi_h(x, a)\|_{\Sigma_h^{-1}}^2 > \gamma \right\} + \frac{4 \log(2H|\mathcal{C}_\ell| |\Gamma| / \delta')}{N}, \\ & \leq \frac{2}{N} \sum_{(x,a) \in \mathcal{D}_\ell(x_\ell, a_\ell)} \mathbb{I} \left\{ \|\varphi_h(x, a)\|_{\Sigma_h^{-1}}^2 > \gamma \right\} + \frac{4 \log \left(\frac{16H|\mathcal{C}_\ell|}{\lambda\delta'\zeta} \right)}{N}, \end{aligned} \quad (86)$$

where the last step follows by the facts that $|\Gamma| \leq \lceil \frac{4}{\zeta\lambda} \rceil$, $\lambda \in (0, 1)$, and $\zeta \in (0, 1/2)$. Now, since $\sigma_{\min}(\Sigma_h) \geq \lambda$ and $\sup_{(x,a) \in \mathcal{X} \times \mathcal{A}} \|\phi_h(\cdot, \cdot)\| \leq 1$ (Assumption I.1), we have that

$$\sup_{(x,a) \in \mathcal{X} \times \mathcal{A}} \|\varphi_h(x, a)\|_{\Sigma_h^{-1}}^2 \leq \frac{4}{\lambda}.$$

Therefore, by the definition of Γ , we have that for all $\ell \in [0..h-1]$ and $(x_\ell, a_\ell) \in \mathcal{C}_\ell$, there exists $\gamma_\ell(x_\ell, a_\ell) \in \Gamma$ such that

$$\begin{aligned}
\max_{(x,a) \in \mathcal{D}_\ell(x_\ell, a_\ell)} \|\varphi_h(x, a)\|_{\Sigma_h^{-1}}^2 &\leq \gamma_\ell(x_\ell, a_\ell), \\
&\leq \max_{(x,a) \in \mathcal{D}_\ell(x_\ell, a_\ell)} \|\varphi_h(x, a)\|_{\Sigma_h^{-1}}^2 + \zeta, \\
&\leq 2 \left(\zeta \vee \max_{(x,a) \in \mathcal{D}_\ell(x_\ell, a_\ell)} \|\varphi_h(x, a)\|_{\Sigma_h^{-1}}^2 \right), \\
&\leq 2 \left(\zeta \vee \|\varphi_h(\hat{x}_h, \hat{a}_h)\|_{\Sigma_h^{-1}}^2 \right), \tag{87}
\end{aligned}$$

where the last inequality follows by the fact that

$$\max_{\ell \in [0..h-1]} \max_{(x,a) \in \mathcal{D}_\ell(x_\ell, a_\ell)} \|\varphi_h(x, a)\|_{\Sigma_h^{-1}}^2 \leq \|\varphi_h(\hat{x}_h, \hat{a}_h)\|_{\Sigma_h^{-1}}^2$$

by definition of (\hat{x}_h, \hat{a}_h) (see [Algorithm 6](#)). Substituting $\gamma_\ell(x_\ell, a_\ell)$ for γ in (86) and using (87), we get that under \mathcal{E} , for all $\ell \in [0..h-1]$ and $(x_\ell, a_\ell) \in \mathcal{C}_\ell$:

$$\mathbb{P}_{\hat{\pi}_{\ell+1:h}} \left[\|\varphi_h(\mathbf{x}_h, \mathbf{a}_h)\|_{\Sigma_h^{-1}}^2 > 2 \left(\zeta \vee \|\varphi_h(\hat{x}_h, \hat{a}_h)\|_{\Sigma_h^{-1}}^2 \right) \mid \mathbf{x}_\ell = x_\ell, \mathbf{a}_\ell = a_\ell \right] \leq \frac{4 \log \left(\frac{16H|\mathcal{C}_\ell|}{\lambda \delta' \zeta} \right)}{N}.$$

This shows that there is an event of probability at least $1 - \delta'/2$ under which (84) holds.

Second claim. We now prove the second claim of the lemma. Let (\mathcal{D}'_ℓ) be the datasets in [Algorithm 6](#) when the algorithm returns. Note that for $\ell \in [0..h-1]$, $(x_\ell, a_\ell) \in \mathcal{C}_\ell$, and $(x_h, a_h) \in \mathcal{D}_\ell(x_\ell, a_\ell)$, the dataset $\bar{\mathcal{D}}_\ell(x_h)$ consists of \bar{N} i.i.d. actions sampled from $\pi_{h,\text{ref}}(\cdot \mid x_h)$. Thus, by Freedman's inequality ([Lemma D.2](#)) and a union bound over $\ell \in [h-1]$, $(x_\ell, a_\ell) \in \mathcal{C}_\ell$, $(x_h, a_h) \in \mathcal{D}_\ell(x_\ell, a_\ell)$, and $\gamma \in \Gamma$, there is an event \mathcal{E}' of probability at least $1 - \delta'/4$ under which for all $\ell \in [0..h-1]$, $(x_\ell, a_\ell) \in \mathcal{C}_\ell$, $(x, a) \in \mathcal{D}(x_\ell, a_\ell)$, and $\gamma \in \Gamma = \{\zeta, 2\zeta, \dots, \lceil \frac{4}{\zeta\lambda} \rceil \zeta\}$:

$$\begin{aligned}
&\mathbb{P}_{\pi_{\text{ref}}} \left[\|\varphi_h(\mathbf{x}_h, \mathbf{a}_h)\|_{\Sigma_h^{-1}}^2 > \gamma \mid \mathbf{x}_h = x \right] \\
&\leq \frac{2}{\bar{N}} \sum_{\bar{a} \in \bar{\mathcal{D}}_\ell(x)} \mathbb{I} \left\{ \|\varphi_h(x, \bar{a})\|_{\Sigma_h^{-1}}^2 > \gamma \right\} + \frac{4 \log \left(\frac{32HN|\mathcal{C}_\ell|}{\lambda \delta' \zeta} \right)}{\bar{N}}. \tag{88}
\end{aligned}$$

Substituting $\gamma_\ell(x_\ell, a_\ell)$ for γ in (88) and using (87), we get that under \mathcal{E}' , for all $\ell \in [0..h-1]$, $(x_\ell, a_\ell) \in \mathcal{C}_\ell$, and $(x, a) \in \mathcal{D}_\ell(x_\ell, a_\ell)$:

$$\mathbb{P}_{\pi_{\text{ref}}} \left[\|\varphi_h(\mathbf{x}_h, \mathbf{a}_h)\|_{\Sigma_h^{-1}}^2 > 2 \left(\zeta \vee \|\varphi_h(\hat{x}_h, \hat{a}_h)\|_{\Sigma_h^{-1}}^2 \right) \mid \mathbf{x}_h = x \right] \leq \frac{4 \log \left(\frac{32HN|\mathcal{C}_\ell|}{\lambda \delta' \zeta} \right)}{\bar{N}}. \tag{89}$$

On the other hand, since for all $(x_\ell, a_\ell) \in \mathcal{C}_\ell$, $\mathcal{D}_\ell(x_\ell, a_\ell)$ consists of N i.i.d. pairs sampled from $\mathbb{P}_{\hat{\pi}_{\ell+1:h}}[(\mathbf{x}_h, \mathbf{a}_h) = \cdot \mid \mathbf{x}_\ell = x_\ell, \mathbf{a}_\ell = a_\ell]$, Freedman's inequality ([Lemma D.2](#)) and a union bound over $\ell \in [0..h-1]$, $(x_\ell, a_\ell) \in \mathcal{C}_\ell$, and $\gamma \in \Gamma$ implies that there is an event \mathcal{E}'' of probability at least $1 - \delta'/4$ under which for all $\ell \in [0..h-1]$, $(x_\ell, a_\ell) \in \mathcal{C}_\ell$, and $\gamma \in \Gamma$ we have that

$$\mathbb{P}_{\hat{\pi}_{\ell+1:h-1}} \left[\mathbb{P}_{\pi_{\text{ref}}} \left[\|\varphi_h(\mathbf{x}_h, \mathbf{a}_h)\|_{\Sigma_h^{-1}}^2 > \gamma \mid \mathbf{x}_h \right] > \frac{4 \log \left(\frac{32HN|\mathcal{C}_\ell|}{\lambda \delta' \zeta} \right)}{\bar{N}} \mid \mathbf{x}_\ell = x_\ell, \mathbf{a}_\ell = a_\ell \right]$$

$$\begin{aligned} &\leq \frac{2}{N} \sum_{(x,a) \in \mathcal{D}_\ell(x_\ell, a_\ell)} \mathbb{I} \left\{ \mathbb{P}_{\pi_{\text{ref}}} \left[\|\varphi_h(\mathbf{x}_h, \mathbf{a}_h)\|_{\Sigma_h^{-1}}^2 > \gamma \mid \mathbf{x}_h = x \right] > \frac{4 \log \left(\frac{32HN|\mathcal{C}_\ell|}{\lambda\delta'\zeta} \right)}{\bar{N}} \right\} \\ &\quad + \frac{4 \log \left(\frac{16H|\mathcal{C}_\ell|}{\lambda\delta'\zeta} \right)}{N}. \end{aligned} \quad (90)$$

Substituting $\gamma_\ell(x_\ell, a_\ell)$ for γ in (90) and using (89), we get that under $\mathcal{E}' \cap \mathcal{E}''$, for all $\ell \in [0..h-1]$ and $(x_\ell, a_\ell) \in \mathcal{C}_\ell$:

$$\begin{aligned} &\mathbb{P}_{\hat{\pi}_{\ell+1:h-1}} \left[\begin{array}{c} \mathbb{P}_{\pi_{\text{ref}}} \left[\|\varphi_h(\mathbf{x}_h, \mathbf{a}_h)\|_{\Sigma_h^{-1}}^2 > 2 \left(\zeta \vee \|\varphi_h(\hat{x}_h, \hat{a}_h)\|_{\Sigma_h^{-1}}^2 \right) \mid \mathbf{x}_h \right] \\ > \frac{4 \log \left(\frac{32HN|\mathcal{C}_\ell|}{\lambda\delta'\zeta} \right)}{\bar{N}} \end{array} \mid \mathbf{x}_\ell = x_\ell, \mathbf{a}_\ell = a_\ell \right] \\ &\leq \frac{4 \log \left(\frac{16H|\mathcal{C}_\ell|}{\lambda\delta'\zeta} \right)}{N}. \end{aligned} \quad (91)$$

We define

$$\mathcal{X}_{h,\text{span}} := \left\{ x \in \mathcal{X} : \begin{array}{c} \mathbb{P}_{\pi_{\text{ref}}} \left[\|\varphi_h(\mathbf{x}_h, \mathbf{a}_h)\|_{\Sigma_h^{-1}}^2 > 2 \left(\zeta \vee \|\varphi_h(\hat{x}_h, \hat{a}_h)\|_{\Sigma_h^{-1}}^2 \right) \mid \mathbf{x}_h = x \right] \\ > \frac{4 \log \left(\frac{32HN|\mathcal{C}_\ell|}{\lambda\delta'\zeta} \right)}{\bar{N}} \end{array} \right\}.$$

By the union bound, $\mathbb{P}[\mathcal{E} \cap \mathcal{E}' \cap \mathcal{E}''] \geq 1 - \delta'$, which combined with (91) completes the proof. \square

Lemma K.2 (Guarantee of UncertainStateAction for MTSS). *Let $\beta, \delta, \varepsilon \in (0, 1)$, $B > 0$, and π_{ref} be given and consider a call to $\text{MTSS}(\beta, \delta, \varepsilon, B, \pi_{\text{ref}})$ (Algorithm 4). Let $\lambda, \nu \in (0, 1)$ and T_{prompt} be as in Algorithm 4. Then, there is an event $\mathcal{E}^{\text{span}}$ of probability at least $1 - \delta/2$ under which for all $t \in [T_{\text{prompt}}]$ and $h \in [H]$, the output (x_h^t, a_h^t) of $\text{UncertainStateAction}_h$ in Line 9 satisfies*

- For all $\ell \in [0..h-1]$ and $(x_\ell, a_\ell) \in \mathcal{C}_\ell^t$,

$$\begin{aligned} &\mathbb{P}_{\hat{\pi}_{\ell+1:h}^t} \left[\|\varphi_h(\mathbf{x}_h, \mathbf{a}_h)\|_{(\Sigma_h^t)^{-1}}^2 > \nu^2 \vee \left(2\|\varphi_h(x_h^t, a_h^t)\|_{(\Sigma_h^t)^{-1}}^2 \right) \mid \mathbf{x}_\ell = x_\ell, \mathbf{a}_\ell = a_\ell \right] \\ &\leq \varepsilon_{\text{span}} := \frac{8 \log \left(\frac{32HT_{\text{prompt}}}{\lambda\delta\nu^2} \right)}{N_{\text{span}}}, \end{aligned}$$

where $\varphi_h(\cdot, \cdot) := \phi_h(\cdot, \cdot) - \phi_h(\cdot, \mathbf{a})$ (\mathbf{a} as in Algorithm 4) and $\hat{\pi}^t$ is as in Algorithm 4.

- Furthermore, there exists $\mathcal{X}_{h,\text{span}}^t \subseteq \mathcal{X}$ such that for all $\ell \in [0..h-1]$ and $(x_\ell, a_\ell) \in \mathcal{C}_\ell^t$, $\mathbb{P}_{\hat{\pi}^t}[\mathbf{x}_h \in \mathcal{X}_{h,\text{span}}^t \mid \mathbf{x}_\ell = x_\ell, \mathbf{a}_\ell = a_\ell] \geq 1 - \varepsilon_{\text{span}}$ and

$$\forall x_h \in \mathcal{X}_{h,\text{span}}^t, \quad \mathbb{P}_{\mathbf{a} \sim \pi_{h,\text{ref}}(\cdot \mid x_h)} \left[\|\varphi_h(x_h, \mathbf{a})\|_{(\Sigma_h^t)^{-1}}^2 > \nu^2 \vee \left(2\|\varphi_h(x_h^t, a_h^t)\|_{(\Sigma_h^t)^{-1}}^2 \right) \right] \leq \varepsilon_{\text{span}},$$

$$\text{where } \varepsilon_{\text{span}} := \frac{8}{N_{\text{span}}} \log \frac{16HT_{\text{prompt}}}{\lambda\delta\nu^2}.$$

Proof of Lemma K.2. The result follows from Lemma K.1 with parameters

$$(\Sigma_h, \delta', \zeta, N, \bar{N}) = (\Sigma_h^t, \delta/(2HT_{\text{prompt}}), \nu^2/2, N_{\text{span}}, \bar{N}_{\text{span}}),$$

and Lemma D.5 (essentially the union bound over $t \in [T_{\text{prompt}}]$ and $h \in [H]$), and the fact that $|\mathcal{C}_h^t| \leq 2T_{\text{prompt}}$, for all $t \in [T_{\text{prompt}}]$ and $h \in [H]$. \square

Appendix L. Guarantee for FitValue

In this section, we present the main guarantee of FitValue (Algorithm 5) as a standalone algorithm, as shown in Lemma L.1. In Appendix L.1, we state and prove supporting lemmas for Lemma L.1. Then, in Appendix L.2, we describe the guarantee of FitValue when used as a subroutine within MTSS (Algorithm 4). For a discussion of the significance of these results and their implications, refer to Section J.1.2.

To state the result, we recall that

$$\bar{\pi}_{h,\theta}(\cdot | x) \propto \pi_{h,\text{ref}}(\cdot | x) \cdot e^{\bar{\varphi}_h(x,\cdot)^\top \theta / \beta}; \quad \text{and} \quad \bar{\pi}_{h,\beta}^*(\cdot | x) \propto \pi_{h,\text{ref}}(\cdot | x) \cdot e^{\bar{\varphi}_h(x,\cdot)^\top \theta_{h,\beta}^* / \beta}, \quad (92)$$

with $\bar{\varphi}_h(x, \cdot) := \varphi_h(x, \cdot) \cdot \mathbb{I} \left\{ \|\varphi_h(x, \cdot)\|_{\Sigma_h^{-1}}^2 \leq \nu^2 \right\}$ (with ν as in Algorithm 4). Further, we recall that $\theta_{h,\beta}^*$ and $\pi_{h,\beta}^*$ denote the optimal KL-regularized policy and corresponding parameter, as in Assumption I.1 and Definition I.2.

Lemma L.1. *Let $h \in [H]$, $C_h, \theta_{h+1:H} \subset \mathbb{B}(B)$, $\Sigma_{h+1:H}, \lambda, \mathfrak{a}, N, M, \tilde{\delta}$, and π_{ref} be given and suppose Assumption I.1 holds with $B > 0$. Further, suppose that C_h is a multiset of the form $C_h = \bigcup_{i \in n} \{(x_i, a_i), (x_i, \mathfrak{a})\}$ for some sequence $(x_i, a_i)_i \subset \mathcal{X} \times \mathcal{A}$ and $n \geq 1$. Consider a call to $\text{FitValue}_h(C_h, \theta_{h+1:H}, \Sigma_{h+1:H}; \mathfrak{a}, N, M, \tilde{\delta}, \pi_{\text{ref}})$ (Algorithm 5) and let \mathcal{D} be the dataset in the algorithm when it returns. Further, define $\varphi_h(\cdot, \cdot) := \phi_h(\cdot, \cdot) - \phi_h(\cdot, \mathfrak{a})$ and $\Sigma_h := \lambda I + \frac{1}{N} \sum_{(x_h:H, a_h:H, r_h:H) \in \mathcal{D}} \varphi_h(x_h, a_h) \varphi_h(x_h, a_h)^\top$. Then, for any $\delta' \in (0, 1)$, with probability at least $1 - \delta'$, the output $\hat{\theta}_h$ of FitValue satisfies:*

$$\begin{aligned} & \|\hat{\theta}_h - \theta_{h,\beta}^*\|_{\Sigma_h}^2 \\ & \leq 4\lambda B^2 + \frac{C_1}{N} + C_2 \tilde{\delta} + C_3 \sum_{(x_h, a_h) \in C_h} \sum_{\ell=h+1}^H \mathbb{P}_{\hat{\pi}_\theta} [M < 16C_{\text{cond}}(\bar{\pi}_{\ell,\theta} | \mathbf{x}_\ell)^2 | \mathbf{x}_h = x_h, \mathbf{a}_h = a_h] \\ & \quad + 2304HB\beta \sum_{(x_h, a_h) \in C_h} \sum_{\ell=h+1}^H \mathbb{E}_{\hat{\pi}_\theta} \left[\min \left(1, C_{\text{cond}}(\bar{\pi}_{\ell,\theta} | \mathbf{x}_\ell) \cdot \sqrt{\frac{2}{M}} \right) | \mathbf{x}_h = x_h, \mathbf{a}_h = a_h \right] \\ & \quad + 19200H\beta^2 \sum_{(x_h, a_h) \in C_h} \sum_{\ell=h+1}^H \mathbb{E}_{\hat{\pi}_\theta} \left[D_{\text{KL}}(\bar{\pi}_{\ell,\theta}(\cdot | \mathbf{x}_\ell) \| \bar{\pi}_{\ell,\beta}^*(\cdot | \mathbf{x}_\ell))^2 | \mathbf{x}_h = x_h, \mathbf{a}_h = a_h \right] \\ & \quad + 7680HR_{\max}^2 \sum_{(x_h, a_h) \in C_h} \sum_{\ell=h+1}^H \mathbb{E}_{\hat{\pi}_\theta} [B_{\ell,\theta}(\mathbf{x}_\ell)^2 | \mathbf{x}_h = x_h, \mathbf{a}_h = a_h], \end{aligned} \quad (93)$$

where

$$(94)$$

We remark that for our application of this result within MTSS, the fact that the right-hand side of Eq. (119) scales with the squared KL divergence $D_{\text{KL}}(\bar{\pi}_{\ell,\theta}(\cdot | \mathbf{x}_\ell) \| \bar{\pi}_{\ell,\beta}^*(\cdot | \mathbf{x}_\ell))^2$ is crucial in enabling favorable error propagation across layers h .

Proof of Lemma L.1. Fix $\delta' \in (0, 1)$. Throughout this proof, for any $h \in [H]$ and $x \in \mathcal{X}$, we let $\hat{\pi}_{h,\theta}(\cdot | x)$ denote the distribution of \mathbf{a} , where $(\mathbf{a}, \rho) = \text{SoftmaxSamplerDensity}_{\beta, M, \tilde{\delta}}(\langle \bar{\varphi}_h(x, \cdot), \theta_h \rangle; x, \pi_{\text{ref}})$.

For $(x_h, a_h) \in \mathcal{C}_h$, let $\mathcal{D}(x_h, a_h)$ be as in [Algorithm 5](#) when the algorithm returns. Note that $\mathcal{D}(x_h, a_h)$ consists of N i.i.d. points z_h which are obtained by first sampling two trajectories $(\mathbf{x}'_h, \mathbf{a}'_h, \mathbf{r}'_h, \boldsymbol{\rho}'_h, \dots, \mathbf{x}'_H, \mathbf{a}'_H, \mathbf{r}'_H, \boldsymbol{\rho}'_H)$ and $(\mathbf{x}''_h, \mathbf{a}''_h, \mathbf{r}''_h, \boldsymbol{\rho}''_h, \dots, \mathbf{x}''_H, \mathbf{a}''_H, \mathbf{r}''_H, \boldsymbol{\rho}''_H)$ via the following process. Initialize $\mathbf{x}'_h = \mathbf{x}''_h = x_h$, $\mathbf{a}'_h = a_h$, and $\mathbf{a}''_h = \mathbf{a}$, and sample $\mathbf{r}'_h \sim r_h^*(\mathbf{x}'_h, \mathbf{a}'_h)$ and $\mathbf{r}''_h \sim r_h^*(\mathbf{x}''_h, \mathbf{a}''_h)$. Then, for $\ell = h+1, \dots, H$,

- Sample $\mathbf{x}'_\ell \sim \mathbb{P}[\cdot \mid \mathbf{x}_{\ell-1} = \mathbf{x}'_{\ell-1}, \mathbf{a}_{\ell-1} = \mathbf{a}'_{\ell-1}]$ and $\mathbf{x}''_\ell \sim \mathbb{P}[\cdot \mid \mathbf{x}_{\ell-1} = \mathbf{x}''_{\ell-1}, \mathbf{a}_{\ell-1} = \mathbf{a}''_{\ell-1}]$;
- Set $(\mathbf{a}'_\ell, \boldsymbol{\rho}'_\ell) \leftarrow \text{SoftmaxSamplerDensity}_{\beta, M, \tilde{\delta}}(\langle \bar{\varphi}_\ell(\mathbf{x}'_\ell, \cdot), \theta_\ell \rangle; \mathbf{x}'_\ell, \pi_{\text{ref}})$;
- Set $(\mathbf{a}''_\ell, \boldsymbol{\rho}''_\ell) \leftarrow \text{SoftmaxSamplerDensity}_{\beta, M, \tilde{\delta}}(\langle \bar{\varphi}_\ell(\mathbf{x}''_\ell, \cdot), \theta_\ell \rangle; \mathbf{x}''_\ell, \pi_{\text{ref}})$;
- Sample rewards $\mathbf{r}'_\ell \sim r_\ell^*(\mathbf{x}'_\ell, \mathbf{a}'_\ell)$ and $\mathbf{r}''_\ell \sim r_\ell^*(\mathbf{x}''_\ell, \mathbf{a}''_\ell)$;

then, finally, set

$$z_h(x_h, a_h) = \mathbf{r}'_h + \sum_{\ell=h+1}^H (\mathbf{r}'_\ell - \beta \log \boldsymbol{\rho}'_\ell) - \mathbf{r}''_h - \sum_{\ell=h+1}^H (\mathbf{r}''_\ell - \beta \log \boldsymbol{\rho}''_\ell). \quad (95)$$

For the rest of this proof, for any $(x_h, a_h, z_h) \in \mathcal{X} \times \mathcal{A} \times \mathbb{R}$, we define

$$\begin{aligned} \hat{f}(x_h, a_h) &:= \varphi_h(x_h, a_h)^\top \hat{\theta}_h, \\ f_\star(x_h, a_h) &:= Q_{h,\beta}^*(x_h, a_h) - Q_{h,\beta}^*(x_h, \mathbf{a}), \\ b(x_h, a_h) &:= Q_{h,\beta}^{\hat{\pi}_\theta}(x_h, a_h) - Q_{h,\beta}^{\hat{\pi}_\theta}(x_h, \mathbf{a}) - Q_{h,\beta}^*(x_h, a_h) - Q_{h,\beta}^*(x_h, \mathbf{a}), \\ \xi(x_h, a_h, z_h) &:= z_h - Q_{h,\beta}^{\hat{\pi}_\theta}(x_h, a_h) + Q_{h,\beta}^{\hat{\pi}_\theta}(x_h, \mathbf{a}). \end{aligned} \quad (96)$$

Further, for all $f, g : \mathcal{X} \times \mathcal{A} \rightarrow \mathbb{R}$, define

$$\begin{aligned} \hat{L}(f) &= \sum_{(x_h, a_h) \in \mathcal{C}_h} \sum_{z_h \in \mathcal{D}(x_h, a_h)} (f(x_h, a_h) - z_h)^2, \\ \|f - g\|^2 &:= N \sum_{(x_h, a_h) \in \mathcal{C}_h} (f(x_h, a_h) - g(x_h, a_h))^2. \end{aligned}$$

Basic least squares analysis. We begin with a standard analysis of least squares. If $\theta_\ell = \theta_\ell^*$ for all $\ell > h$, then the conditional mean of z_h is equal to $f_\star(x_h, a_h) = Q_{h,\beta}^*(x_h, a_h) - Q_{h,\beta}^*(x_h, \mathbf{a})$ up to negligible error caused by approximate sampling via `SoftmaxSamplerDensity`, and hence θ_h is solving (nearly) well-specified linear regression. The crux of the proof that follows will be to bound the misspecification corresponding to the term $b(x_h, a_h)$, which reflects the fact that $Q_{h,\beta}^{\hat{\pi}_\theta}(x_h, a_h) - Q_{h,\beta}^{\hat{\pi}_\theta}(x_h, \mathbf{a})$ may not be linear in general.

To begin, note that with the notation introduced so far, $\hat{\theta}_h$ in [Algorithm 5](#) satisfies

$$\hat{\theta}_h \in \arg \min_{\tilde{\theta} \in \mathbb{B}(B)} \hat{L}(\langle \varphi_h(\cdot, \cdot), \tilde{\theta} \rangle). \quad (97)$$

This, together with the facts that $f_\star(x, a) = \varphi(x, a)^\top \theta_{h,\beta}^*$ (by [Assumption I.1](#) and [Lemma G.1](#)) and $\theta_{h,\beta}^* \in \mathbb{B}(B)$ implies that

$$0 \geq \hat{L}_n(\hat{f}) - \hat{L}_n(f_\star),$$

$$= 2 \sum_{(x_h, a_h) \in \mathcal{C}_h} \sum_{z_h \in \mathcal{D}(x_h, a_h)} (f_\star(x_h, a_h) - z_h) \cdot (\hat{f}(x_h, a_h) - f_\star(x_h, a_h)) + \|\hat{f} - f_\star\|^2.$$

Rearranging, we get that

$$\begin{aligned}
& \|\hat{f} - f_\star\|^2 \\
& \leq 4 \sum_{(x_h, a_h) \in \mathcal{C}_h} \sum_{z_h \in \mathcal{D}(x_h, a_h)} (z_h - f_\star(x_h, a_h)) \cdot (\hat{f}(x_h, a_h) - f_\star(x_h, a_h)) - \|\hat{f} - f_\star\|^2, \\
& \leq 4 \sum_{(x_h, a_h) \in \mathcal{C}_h} \sum_{z_h \in \mathcal{D}(x_h, a_h)} (\xi(x_h, a_h, z_h) + b(x_h, a_h)) \cdot (\hat{f}(x_h, a_h) - f_\star(x_h, a_h)) \\
& \quad - \|\hat{f} - f_\star\|^2, \\
& \leq 4 \sum_{(x_h, a_h) \in \mathcal{C}_h} \sum_{z_h \in \mathcal{D}(x_h, a_h)} \xi(x_h, a_h, z_h) \cdot (\hat{f}(x_h, a_h) - f_\star(x_h, a_h)) - \|\hat{f} - f_\star\|^2 \\
& \quad + 4 \sum_{(x_h, a_h) \in \mathcal{C}_h} \sum_{z_h \in \mathcal{D}(x_h, a_h)} b(x_h, a_h) \cdot (\hat{f}(x_h, a_h) - f_\star(x_h, a_h)), \\
& \leq 4 \sum_{(x_h, a_h) \in \mathcal{C}_h} \sum_{z_h \in \mathcal{D}(x_h, a_h)} \xi(x_h, a_h, z_h) \cdot (\hat{f}(x_h, a_h) - f_\star(x_h, a_h)) - \|\hat{f} - f_\star\|^2 \\
& \quad + 8 \sum_{(x_h, a_h) \in \mathcal{C}_h} \sum_{z_h \in \mathcal{D}(x_h, a_h)} b(x_h, a_h)^2 + \frac{1}{2} \|\hat{f} - f_\star\|^2, \tag{98}
\end{aligned}$$

where the last inequality follows by AM-GM. Rearranging (98), we get that

$$\begin{aligned}
\|\hat{f} - f_\star\|^2 & \leq 8 \underbrace{\sum_{(x_h, a_h) \in \mathcal{C}_h} \sum_{z_h \in \mathcal{D}(x_h, a_h)} \xi(x_h, a_h, z_h) \cdot (\hat{f}(x_h, a_h) - f_\star(x_h, a_h)) - 2\|\hat{f} - f_\star\|^2}_{\text{I}} \\
& \quad + 16 \underbrace{\sum_{(x_h, a_h) \in \mathcal{C}_h} \sum_{z_h \in \mathcal{D}(x_h, a_h)} b(x_h, a_h)^2}_{\text{II}}. \tag{99}
\end{aligned}$$

Bounding Term I. We first bound Term I, which reflects the (nearly) mean-zero noise in the regression targets. Concretely, by Lemma L.2 (stated and proven in the sequel), we have for all $(x_h, a_h) \in \mathcal{C}_h$:

$$\begin{aligned}
& |\mathbb{E}[\xi(x_h, a_h, z_h(x_h, a_h))]| \\
& \leq \beta \sum_{\ell=h+1}^H \sum_{a \in \{a_h, \mathbf{a}\}} \mathbb{E}_{\hat{\pi}_\theta} \left[\mathbb{I}\{M \geq 4C_{\text{cond}}(\bar{\pi}_{\ell, \theta} \mid \mathbf{x}_\ell)^2\} \cdot C_{\text{cond}}(\bar{\pi}_{\ell, \theta} \mid \mathbf{x}_\ell) \cdot \sqrt{\frac{2}{M}} \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a \right] \\
& \quad + 16HB\tilde{\delta} + 8H\beta \log(4M \log(4\tilde{\delta}^{-1}))\tilde{\delta} \\
& \quad + 8B \sum_{\ell=h+1}^H \sum_{a \in \{a_h, \mathbf{a}\}} \mathbb{P}_{\hat{\pi}_\theta} [M < 4C_{\text{cond}}(\bar{\pi}_{\ell, \theta} \mid \mathbf{x}_\ell)^2 \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a], \tag{100}
\end{aligned}$$

where $\mathbb{E}[\cdot]$ denotes the expectation over $z_h(x_h, a_h)$ under the process in Eq. (98) (beginning from (x_h, a_h)). Given this, we apply Freedman's inequality (Lemma D.2), using

- The union bound over an $\frac{1}{N}$ -net of $\mathbb{B}(B)$ in $\|\cdot\|$ -distance;
- The fact that

$$\begin{aligned} & \max(\|\hat{f}\|_\infty, \|f^\star\|_\infty, \|\xi\|_\infty) \\ & \leq 2BH + 2\beta H \max_{\ell \in [h+1..H]} \sup_{(x,a) \in \mathcal{X} \times \mathcal{A}} \max \left(\left| \log \frac{\bar{\pi}_{\ell,\theta}(a \mid x)}{\pi_{\text{ref}}(a \mid x)} \right|, \left| \log \frac{\pi_{\ell,\beta}^\star(a \mid x)}{\pi_{\text{ref}}(a \mid x)} \right|, |\log \rho'_\ell|, |\log \rho''_\ell| \right), \\ & \leq 6HB, \end{aligned} \tag{101}$$

since $e^{-2B/\beta} \leq \frac{\bar{\pi}_{\ell,\theta}(\cdot \mid \cdot)}{\pi_{\text{ref}}(\cdot \mid \cdot)} \wedge \frac{\pi_{\ell,\beta}^\star(\cdot \mid \cdot)}{\pi_{\text{ref}}(\cdot \mid \cdot)} \leq \frac{\bar{\pi}_{\ell,\theta}(\cdot \mid \cdot)}{\pi_{\text{ref}}(\cdot \mid \cdot)} \vee \frac{\pi_{\ell,\beta}^\star(\cdot \mid \cdot)}{\pi_{\text{ref}}(\cdot \mid \cdot)} \leq e^{2B/\beta}$ for all $\theta_\ell \in \mathbb{B}(B)$, and for all $\ell \in [H]$, $\mathbf{r}'_\ell, \mathbf{r}''_\ell \in [0, B]$ and $\rho'_\ell, \rho''_\ell \in [e^{-2B/\beta}, e^{2B/\beta}]$ (by [Theorem F.1](#));

to conclude that with probability at least $1 - \delta'$,

$$\begin{aligned} & \sum_{(x_h, a_h) \in \mathcal{C}_h} \sum_{z_h \in \mathcal{D}(x_h, a_h)} \xi(x_h, a_h, z_h) \cdot (\hat{f}(x_h, a_h) - f_\star(x_h, a_h)) \\ & \leq N \sum_{(x_h, a_h) \in \mathcal{C}_h} \mathbb{E}[\xi(x_h, a_h, \mathbf{z}_h(x_h, a_h))] \cdot (\hat{f}(x_h, a_h) - f_\star(x_h, a_h)) \\ & \quad + \frac{N}{144B^2H^2} \sum_{(x_h, a_h) \in \mathcal{C}_h} \mathbb{E} \left[\left(\xi(x_h, a_h, \mathbf{z}_h(x_h, a_h))^2 \cdot (\hat{f}(x_h, a_h) - f_\star(x_h, a_h)) \right)^2 \right] \\ & \quad + 100B^2H^2d \log(3N/\delta') + 60H^2B^2|\mathcal{C}_h|, \\ & \leq 6HBN \sum_{(x_h, a_h) \in \mathcal{C}_h} |\mathbb{E}[\xi(x_h, a_h, \mathbf{z}_h(x_h, a_h))]| + \frac{1}{4} \|\hat{f} - f_\star\|^2 + 100B^2H^2d \log(3N/\delta') + 60H^2B^2|\mathcal{C}_h|, \end{aligned}$$

and so by [\(100\)](#) and [Lemma N.7](#) (and that \mathcal{C}_h is a multiset satisfying $\mathcal{C}_h = \bigcup_{i \in n} \{(x_i, a_i), (x_i, \mathbf{a})\}$)

$$\begin{aligned} & \leq \frac{1}{4} \|\hat{f} - f_\star\|^2 + 100B^2H^2d \log(3N/\delta') + 60H^2B^2|\mathcal{C}_h| \\ & \quad + 288NHB\beta \sum_{(x_h, a_h) \in \mathcal{C}_h} \sum_{\ell=h+1}^H \mathbb{E}_{\hat{\pi}_\theta} \left[\mathbb{I}\{M \geq 4C_{\text{cond}}(\bar{\pi}_{\ell,\theta} \mid \mathbf{x}_\ell)^2\} \cdot C_{\text{cond}}(\bar{\pi}_{\ell,\theta} \mid \mathbf{x}_\ell) \cdot \sqrt{\frac{2}{M}} \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h \right] \\ & \quad + 384|\mathcal{C}_h|NH^2B^2\tilde{\delta} + 48|\mathcal{C}_h|NH^2B\beta \log(4M \log(4\tilde{\delta}^{-1}))\tilde{\delta} \\ & \quad + 288NHB^2 \sum_{(x_h, a_h) \in \mathcal{C}_h} \sum_{\ell=h+1}^H \mathbb{P}_{\hat{\pi}_\theta} [M < 4C_{\text{cond}}(\bar{\pi}_{\ell,\theta} \mid \mathbf{x}_\ell)^2 \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h]. \end{aligned} \tag{102}$$

Using this together with the expression of Term I in [\(99\)](#), we have that with probability at least $1 - \delta'$,

Term I

$$\begin{aligned} & \leq 800B^2H^2d \log(3N/\delta') + 480H^2B^2|\mathcal{C}_h| \\ & \quad + 2304NHB\beta \sum_{(x_h, a_h) \in \mathcal{C}_h} \sum_{\ell=h+1}^H \mathbb{E}_{\hat{\pi}_\theta} \left[\mathbb{I}\{M \geq 4C_{\text{cond}}(\bar{\pi}_{\ell,\theta} \mid \mathbf{x}_\ell)^2\} \cdot C_{\text{cond}}(\bar{\pi}_{\ell,\theta} \mid \mathbf{x}_\ell) \cdot \sqrt{\frac{2}{M}} \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h \right] \end{aligned}$$

$$\begin{aligned}
& + 3072|\mathcal{C}_h|NH^2B^2\tilde{\delta} + 2304|\mathcal{C}_h|NH^2B\beta\log(4M\log(4\tilde{\delta}^{-1}))\tilde{\delta} \\
& + 2304NHB^2 \sum_{(x_h, a_h) \in \mathcal{C}_h} \sum_{\ell=h+1}^H \mathbb{P}_{\hat{\pi}_\theta} [M < 4C_{\text{cond}}(\bar{\pi}_{\ell, \theta} \mid \mathbf{x}_\ell)^2 \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h]. \quad (103)
\end{aligned}$$

Bounding Term II. To bound the second term in (99), which reflects the misspecification level in the regression problem, we need to bound $b(x_h, a_h) = Q_{h, \beta}^{\hat{\pi}_\theta}(x_h, a_h) - Q_{h, \beta}^{\hat{\pi}_\theta}(x_h, \mathbf{a}) - Q_{h, \beta}^*(x_h, a_h) + Q_{h, \beta}^*(x_h, \mathbf{a})$ for $(x_h, a_h) \in \mathcal{C}_h$. By the performance difference lemma (Lemma N.6) and Lemma N.4, we have that for any $(x_h, a_h) \in \mathcal{X} \times \mathcal{A}$:

$$\begin{aligned}
& \left| Q_{h, \beta}^*(x_h, a_h) - Q_{h, \beta}^{\hat{\pi}_\theta}(x_h, a_h) \right| \\
& \leq \left| \sum_{\ell=h+1}^H \mathbb{E}_{\hat{\pi}_\theta} \left[\sum_{a \in \mathcal{A}} \pi_{\ell, \beta}^*(a \mid \mathbf{x}_\ell) \cdot \left(Q_{\ell, \beta}^*(\mathbf{x}_\ell, a) - \beta \cdot \log \frac{\pi_{\ell, \beta}^*(a \mid \mathbf{x}_\ell)}{\pi_{\ell, \text{ref}}(a \mid \mathbf{x}_\ell)} \right) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h \right] \right. \\
& \quad \left. - \sum_{\ell=h+1}^H \mathbb{E}_{\hat{\pi}_\theta} \left[\sum_{a \in \mathcal{A}} \hat{\pi}_{\ell, \theta}(a \mid \mathbf{x}_\ell) \cdot \left(Q_{\ell, \beta}^*(\mathbf{x}_\ell, a) - \beta \cdot \log \frac{\hat{\pi}_{\ell, \theta}(a \mid \mathbf{x}_\ell)}{\pi_{\ell, \text{ref}}(a \mid \mathbf{x}_\ell)} \right) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h \right] \right|, \\
& \leq \left| \sum_{\ell=h+1}^H \mathbb{E}_{\hat{\pi}_\theta} \left[\sum_{a \in \mathcal{A}} \pi_{\ell, \beta}^*(a \mid \mathbf{x}_\ell) \cdot \left(Q_{\ell, \beta}^*(\mathbf{x}_\ell, a) - \beta \cdot \log \frac{\pi_{\ell, \beta}^*(a \mid \mathbf{x}_\ell)}{\pi_{\ell, \text{ref}}(a \mid \mathbf{x}_\ell)} \right) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h \right] \right. \\
& \quad \left. - \sum_{\ell=h+1}^H \mathbb{E}_{\hat{\pi}_\theta} \left[\sum_{a \in \mathcal{A}} \hat{\pi}_{\ell, \theta}(a \mid \mathbf{x}_\ell) \cdot \left(Q_{\ell, \beta}^*(\mathbf{x}_\ell, a) - \beta \cdot \log \frac{\hat{\pi}_{\ell, \theta}(a \mid \mathbf{x}_\ell)}{\pi_{\ell, \text{ref}}(a \mid \mathbf{x}_\ell)} \right) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h \right] \right| \\
& \quad + \sum_{\ell=h+1}^H \beta \cdot \mathbb{E}_{\hat{\pi}_\theta} [D_{\text{KL}}(\hat{\pi}_{\ell, \theta}(\cdot \mid \mathbf{x}_\ell) \parallel \bar{\pi}_{\ell, \theta}(\cdot \mid \mathbf{x}_\ell)) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h]. \quad (104)
\end{aligned}$$

Now, by Lemma L.3 (stated and proven in the sequel), we can bound the KL term in (104) as follows: for all $\ell \in [H]$ and $(x_h, a_h) \in \mathcal{X} \times \mathcal{A}$:

$$\begin{aligned}
& \mathbb{E}_{\hat{\pi}_\theta} [D_{\text{KL}}(\hat{\pi}_{\ell, \theta}(\cdot \mid \mathbf{x}_\ell) \parallel \bar{\pi}_{\ell, \theta}(\cdot \mid \mathbf{x}_\ell)) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h] \\
& \leq 4 \left(\frac{R_{\max}}{\beta} + \log(4M\log(4\tilde{\delta}^{-1})) \right) \tilde{\delta} \\
& \quad + \mathbb{E}_{\hat{\pi}_\theta} [\mathbb{I}\{M < C_{\text{cond}}(\bar{\pi}_{\ell, \theta} \mid \mathbf{x}_\ell)\} \cdot D_{\text{KL}}(\hat{\pi}_{\ell, \theta}(\cdot \mid \mathbf{x}_\ell) \parallel \bar{\pi}_{\ell, \theta}(\cdot \mid \mathbf{x}_\ell)) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h], \\
& \leq 4 \left(\frac{R_{\max}}{\beta} + \log(4M\log(4\tilde{\delta}^{-1})) \right) \tilde{\delta} \\
& \quad + \frac{R_{\max}}{\beta} \log(4M\log(4\tilde{\delta}^{-1})) \cdot \mathbb{P}_{\hat{\pi}_\theta} [M < 4C_{\text{cond}}(\bar{\pi}_{\ell, \theta} \mid \mathbf{x}_\ell) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h].
\end{aligned}$$

Now, since $M \geq 1$, we have that $M < 4C_{\text{cond}}(\bar{\pi}_{\ell, \theta} \mid \mathbf{x}_\ell)$ only if $M < 16C_{\text{cond}}(\bar{\pi}_{\ell, \theta} \mid \mathbf{x}_\ell)^2$, and so for all $\ell \in [h+1 \dots H]$ and $(x_h, a_h) \in \mathcal{X} \times \mathcal{A}$:

$$\mathbb{P}_{\hat{\pi}_\theta} [M < 4C_{\text{cond}}(\bar{\pi}_{\ell, \theta} \mid \mathbf{x}_\ell) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h] \leq \mathbb{P}_{\hat{\pi}_\theta} [M < 16C_{\text{cond}}(\bar{\pi}_{\ell, \theta} \mid \mathbf{x}_\ell)^2 \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h]. \quad (105)$$

Therefore, we have

$$\mathbb{E}_{\hat{\pi}_\theta} [D_{\text{KL}}(\hat{\pi}_{\ell, \theta}(\cdot \mid \mathbf{x}_\ell) \parallel \bar{\pi}_{\ell, \theta}(\cdot \mid \mathbf{x}_\ell)) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h]$$

$$\begin{aligned} &\leq 4 \left(\frac{R_{\max}}{\beta} + \log(4M \log(4\tilde{\delta}^{-1})) \right) \tilde{\delta} \\ &\quad + \frac{R_{\max}}{\beta} \log(4M \log(4\tilde{\delta}^{-1})) \cdot \mathbb{P}_{\hat{\pi}_\theta} [M < 16C_{\text{cond}}(\bar{\pi}_{\ell,\theta} | \mathbf{x}_\ell)^2 | \mathbf{x}_h = x_h, \mathbf{a}_h = a_h]. \end{aligned} \quad (106)$$

It remains to bound the absolute value term on the right-hand side of (104). As a starting point, note that for all $\ell \in [h+1 .. H]$,

$$\left| Q_{\ell,\beta}^*(\cdot, \cdot) - \beta \cdot \log \frac{\bar{\pi}_{\ell,\theta}(\cdot | \cdot)}{\pi_{\ell,\text{ref}}(\cdot | \cdot)} \right| \leq HB + 2H\beta \sup_{(x,a) \in \mathcal{X} \times \mathcal{A}} \left| \log \frac{\bar{\pi}_{\ell,\theta}(a | x)}{\pi_{\ell,\text{ref}}(a | x)} \right| \leq 5HB,$$

since $e^{-2B/\beta} \leq \frac{\bar{\pi}_{\ell,\theta'}(\cdot | \cdot)}{\pi_{\ell,\text{ref}}(\cdot | \cdot)} \leq e^{2B/\beta}$ for all $\theta'_\ell \in \mathbb{B}(B)$. Thus, by Lemma F.3, we have that for all $\ell \in [h+1 .. H]$ and $(x_h, a_h) \in \mathcal{X} \times \mathcal{A}$:

$$\begin{aligned} &\left| \mathbb{E}_{\hat{\pi}_\theta} \left[\sum_{a \in \mathcal{A}} \hat{\pi}_{\ell,\theta}(a | \mathbf{x}_\ell) \cdot \left(Q_{\ell,\beta}^*(\mathbf{x}_\ell, a) - \beta \cdot \log \frac{\bar{\pi}_{\ell,\theta}(a | \mathbf{x}_\ell)}{\pi_{\ell,\text{ref}}(a | \mathbf{x}_\ell)} \right) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h \right] \right. \\ &\quad \left. - \mathbb{E}_{\hat{\pi}_\theta} \left[\sum_{a \in \mathcal{A}} \bar{\pi}_{\ell,\theta}(a | \mathbf{x}_\ell) \cdot \left(Q_{\ell,\beta}^*(\mathbf{x}_\ell, a) - \beta \cdot \log \frac{\bar{\pi}_{\ell,\theta}(a | \mathbf{x}_\ell)}{\pi_{\ell,\text{ref}}(a | \mathbf{x}_\ell)} \right) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h \right] \right| \\ &\leq 5HB\tilde{\delta} + 5HB \cdot \mathbb{P}_{\hat{\pi}_\theta} [M < 4C_{\text{cond}}(\bar{\pi}_{\ell,\theta} | \mathbf{x}_\ell) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h], \\ &\leq 5HB\tilde{\delta} + 5HB \cdot \mathbb{P}_{\hat{\pi}_\theta} [M < 16C_{\text{cond}}(\bar{\pi}_{\ell,\theta} | \mathbf{x}_\ell)^2 \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h], \end{aligned} \quad (107)$$

where the last inequality follows by (105). On the other hand, by Jensen's inequality and the triangle inequality, we have for all $(x_h, a_h) \in \mathcal{X} \times \mathcal{A}$:

$$\begin{aligned} &\left| \sum_{\ell=h+1}^H \mathbb{E}_{\hat{\pi}_\theta} \left[\sum_{a \in \mathcal{A}} \pi_{\ell,\beta}^*(a | \mathbf{x}_\ell) \cdot \left(Q_{\ell,\beta}^*(\mathbf{x}_\ell, a) - \beta \cdot \log \frac{\pi_{\ell,\beta}^*(a | \mathbf{x}_\ell)}{\pi_{\ell,\text{ref}}(a | \mathbf{x}_\ell)} \right) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h \right] \right. \\ &\quad \left. - \sum_{\ell=h+1}^H \mathbb{E}_{\hat{\pi}_\theta} \left[\sum_{a \in \mathcal{A}} \bar{\pi}_{\ell,\theta}(a | \mathbf{x}_\ell) \cdot \left(Q_{\ell,\beta}^*(\mathbf{x}_\ell, a) - \beta \cdot \log \frac{\bar{\pi}_{\ell,\theta}(a | \mathbf{x}_\ell)}{\pi_{\ell,\text{ref}}(a | \mathbf{x}_\ell)} \right) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h \right] \right| \\ &\leq \sum_{\ell=h+1}^H \mathbb{E}_{\hat{\pi}_\theta} \left[\left| \sum_{a \in \mathcal{A}} \pi_{\ell,\beta}^*(a | \mathbf{x}_\ell) \cdot \left(Q_{\ell,\beta}^*(\mathbf{x}_\ell, a) - \beta \cdot \log \frac{\pi_{\ell,\beta}^*(a | \mathbf{x}_\ell)}{\pi_{\ell,\text{ref}}(a | \mathbf{x}_\ell)} \right) \right. \right. \\ &\quad \left. \left. - \sum_{a \in \mathcal{A}} \bar{\pi}_{\ell,\theta}(a | \mathbf{x}_\ell) \cdot \left(Q_{\ell,\beta}^*(\mathbf{x}_\ell, a) - \beta \cdot \log \frac{\bar{\pi}_{\ell,\theta}(a | \mathbf{x}_\ell)}{\pi_{\ell,\text{ref}}(a | \mathbf{x}_\ell)} \right) \right| \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h \right], \end{aligned}$$

and so by Lemma N.3, we have for $B_{\ell,\theta}$ as in the lemma statement:

$$\begin{aligned} &\leq \beta \sum_{\ell=h+1}^H \mathbb{E}_{\hat{\pi}_\theta} [D_{\text{KL}}(\bar{\pi}_{\ell,\theta}(\cdot | \mathbf{x}_\ell) \| \pi_{\ell,\beta}^*(\cdot | \mathbf{x}_\ell)) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h] \\ &\quad + 2R_{\max} \sum_{\ell=h+1}^H \mathbb{E}_{\hat{\pi}_\theta} [B_{\ell,\theta}(\mathbf{x}_\ell) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h]. \end{aligned}$$

Combining this with (104), (106), and (107), we get that for all $(x_h, a_h) \in \mathcal{X} \times \mathcal{A}$:

$$|b(x_h, a_h)|$$

$$\begin{aligned}
 &\leq 2H(4R_{\max} + 4\beta \log(4M \log(4\tilde{\delta}^{-1})) + 5HB) \cdot \tilde{\delta} \\
 &\quad + \left(2R_{\max} \log(4M \log(4\tilde{\delta}^{-1})) + 10HB\right) \sum_{a \in \{a_h, \mathfrak{a}\}} \sum_{\ell=h+1}^H \mathbb{P}_{\hat{\pi}_\theta} [M < 16C_{\text{cond}}(\bar{\pi}_{\ell, \theta} \mid \mathbf{x}_\ell)^2 \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a] \\
 &\quad + 2\beta \sum_{a \in \{a_h, \mathfrak{a}\}} \sum_{\ell=h+1}^H \mathbb{E}_{\hat{\pi}_\theta} [D_{\text{KL}}(\bar{\pi}_{\ell, \theta}(\cdot \mid \mathbf{x}_\ell) \parallel \bar{\pi}_{\ell, \beta}^*(\cdot \mid \mathbf{x}_\ell)) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a] \\
 &\quad + 4R_{\max} \sum_{a \in \{a_h, \mathfrak{a}\}} \sum_{\ell=h+1}^H \mathbb{E}_{\hat{\pi}_\theta} [B_{\ell, \theta}(\mathbf{x}_\ell) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a]. \tag{108}
 \end{aligned}$$

Thus, using Jensen's inequality and [Lemma N.7](#) (together with the fact that \mathcal{C}_h is multiset satisfying $\mathcal{C}_h = \bigcup_{i \in [n]} \{(x_i, a_i), (x_i, \mathfrak{a})\}$), we have

$$\begin{aligned}
 &\sum_{(x_h, a_h) \in \mathcal{C}_h} \sum_{z_h \in \mathcal{D}(x_h, a_h)} b(x_h, a_h)^2 \\
 &= 20N|\mathcal{C}_h| \cdot (12R_{\max}^2 + 48\beta^2 \log(4M \log(4\tilde{\delta}^{-1}))^2 + 75H^2B^2) \cdot \tilde{\delta}^2 \\
 &\quad + 6HN \cdot (8R_{\max}^2 \log(4M \log(4\tilde{\delta}^{-1}))^2 + 200H^2B^2) \tag{109}
 \end{aligned}$$

$$\begin{aligned}
 &\cdot \sum_{(x_h, a_h) \in \mathcal{C}_h} \sum_{\ell=h+1}^H \mathbb{P}_{\hat{\pi}_\theta} [M < 16C_{\text{cond}}(\bar{\pi}_{\ell, \theta} \mid \mathbf{x}_\ell)^2 \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h]^2 \\
 &\quad + 120NH\beta^2 \sum_{(x_h, a_h) \in \mathcal{C}_h} \sum_{\ell=h+1}^H \mathbb{E}_{\hat{\pi}_\theta} [D_{\text{KL}}(\bar{\pi}_{\ell, \theta}(\cdot \mid \mathbf{x}_\ell) \parallel \bar{\pi}_{\ell, \beta}^*(\cdot \mid \mathbf{x}_\ell))^2 \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h] \\
 &\quad + 240NHR_{\max}^2 \sum_{(x_h, a_h) \in \mathcal{C}_h} \sum_{\ell=h+1}^H \mathbb{E}_{\hat{\pi}_\theta} [B_{\ell, \theta}(\mathbf{x}_\ell)^2 \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h]. \tag{110}
 \end{aligned}$$

Putting everything together. Combining (110) with (103) and (99), we get that with probability at least $1 - \delta'$,

$$\begin{aligned}
 &\|\hat{f} - f_\star\|^2 \\
 &\leq C_1 + NC_2\tilde{\delta} \\
 &\quad + NC_3 \sum_{(x_h, a_h) \in \mathcal{C}_h} \sum_{\ell=h+1}^H \mathbb{P}_{\hat{\pi}_\theta} [M < 16C_{\text{cond}}(\bar{\pi}_{\ell, \theta} \mid \mathbf{x}_\ell)^2 \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h] \\
 &\quad + 2304NHB\beta \sum_{(x_h, a_h) \in \mathcal{C}_h} \sum_{\ell=h+1}^H \mathbb{E}_{\hat{\pi}_\theta} \left[\mathbb{I}\{M \geq 4C_{\text{cond}}(\bar{\pi}_{\ell, \theta} \mid \mathbf{x}_\ell)^2\} \cdot C_{\text{cond}}(\bar{\pi}_{\ell, \theta} \mid \mathbf{x}_\ell) \cdot \sqrt{\frac{2}{M}} \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h \right] \\
 &\quad + 19200NH\beta^2 \sum_{(x_h, a_h) \in \mathcal{C}_h} \sum_{\ell=h+1}^H \mathbb{E}_{\hat{\pi}_\theta} [D_{\text{KL}}(\bar{\pi}_{\ell, \theta}(\cdot \mid \mathbf{x}_\ell) \parallel \bar{\pi}_{\ell, \beta}^*(\cdot \mid \mathbf{x}_\ell))^2 \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h] \\
 &\quad + 7680NHR_{\max}^2 \sum_{(x_h, a_h) \in \mathcal{C}_h} \sum_{\ell=h+1}^H \mathbb{E}_{\hat{\pi}_\theta} [B_{\ell, \theta}(\mathbf{x}_\ell)^2 \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h], \tag{111}
 \end{aligned}$$

where

$$\begin{aligned} C_1 &:= 6400B^2H^2d\log(3N/\delta') + 3840H^2B^2|\mathcal{C}_h|, \\ C_2 &:= 19200|\mathcal{C}_h| \cdot (12R_{\max}^2 + 48\beta^2\log(4M\log(4\tilde{\delta}^{-1}))^2 + 85H^2B^2) + 3072|\mathcal{C}_h|H^2B\beta\log(4M\log(4\tilde{\delta}^{-1})), \\ C_3 &:= 16H \cdot (8R_{\max}^2\log(4M\log(4\tilde{\delta}^{-1}))^2 + 200H^2B^2) + 3072HB^2. \end{aligned}$$

Combining this with the fact that

$$\begin{aligned} \|\hat{\theta}_h - \theta_{h,\beta}^*\|_{\Sigma_h}^2 &= \lambda\|\hat{\theta}_h - \theta_{h,\beta}^*\|^2 + \frac{1}{N}\|\hat{f} - f_\star\|^2, \quad (\text{by definition of } \Sigma_h) \\ &\leq 4\lambda B^2 + \frac{1}{N}\|\hat{f} - f_\star\|^2, \quad (\text{by Assumption I.1 and } \hat{\theta}_h \in \mathbb{B}(B)), \end{aligned}$$

we obtain the desired result. It remains to prove [Lemma L.2](#) and [Lemma L.3](#). □

L.1. Helper Lemmas for FitValue Guarantee

Lemma L.2. *Consider the setting of [Lemma L.1](#) and the notation in its proof. Let $\theta_{h+1:H} \in \mathbb{R}^{d \cdot (H-h)}$ be as [Lemma L.1](#). Fix $(x_h, a_h) \in \mathcal{C}_h$, and let $\mathbf{z}_h(x_h, a_h)$ be the random variable in (95) in the proof of [Lemma L.1](#). Then, the function ξ in (96) satisfies*

$$\begin{aligned} &|\mathbb{E}[\xi(x_h, a_h, \mathbf{z}_h(x_h, a_h))]| \\ &\leq \beta \sum_{\ell=h+1}^H \sum_{a \in \{a_h, \mathbf{a}\}} \mathbb{E}_{\hat{\pi}_\theta} \left[\mathbb{I}\{M \geq 4C_{\text{cond}}(\bar{\pi}_{\ell,\theta} \mid \mathbf{x}_\ell)^2\} \cdot C_{\text{cond}}(\bar{\pi}_{\ell,\theta} \mid \mathbf{x}_\ell) \cdot \sqrt{\frac{2}{M}} \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a \right] \\ &\quad + 16HB\tilde{\delta} + 8H\beta\log(4M\log(4\tilde{\delta}^{-1}))\tilde{\delta} \\ &\quad + 4B \sum_{\ell=h+1}^H \sum_{a \in \{a_h, \mathbf{a}\}} \mathbb{P}_{\hat{\pi}_\theta} [M < 4C_{\text{cond}}(\bar{\pi}_{\ell,\theta} \mid \mathbf{x}_\ell)^2 \mid \mathbf{x}_h = x, \mathbf{a}_h = a]. \end{aligned} \tag{112}$$

Proof of Lemma L.2. Let (x_h, a_h) be fixed, as in the lemma statement. In addition to $\theta_{h+1:H}$ as in the lemma statement, fix $\theta_{1:h} \in \mathbb{R}^{dh}$. Let $(\mathbf{x}_1, \mathbf{a}_1, \boldsymbol{\rho}_1, \mathbf{r}_1), \dots, (\mathbf{x}_H, \mathbf{a}_H, \boldsymbol{\rho}_H, \mathbf{r}_H)$ be the sequence of random variables generated via the process

$$(\mathbf{a}_h, \boldsymbol{\rho}_h) = \text{SoftmaxSamplerDensity}_{\beta, M, \tilde{\delta}}(\langle \bar{\varphi}_h(\mathbf{x}_h, \cdot), \theta_h \rangle; \mathbf{x}_h, \pi_{\text{ref}}), \quad \mathbf{r}_h \sim r_h^*(\mathbf{x}_h, \mathbf{a}_h), \quad \mathbf{x}_{h+1} \sim P_h(\cdot \mid \mathbf{x}_h, \mathbf{a}_h),$$

initialized from $\mathbf{x}_1 \sim P_0(\cdot \mid \emptyset)$ (we use \mathbf{x}_{H+1} to denote a terminal state with zero reward). We write $\mathbb{P}_{\hat{\pi}_\theta}[\cdot]$ and $\mathbb{E}_{\hat{\pi}_\theta}[\cdot]$ to denote the law and expectation under this process.

With this observe that $\xi(x_h, a_h, \mathbf{z}_h(x_h, a_h))$ satisfies

$$\begin{aligned} \mathbb{E}[\xi(x_h, a_h, \mathbf{z}_h(x_h, a_h))] &= \mathbb{E}_{\hat{\pi}_\theta} \left[\mathbf{r}_h + \sum_{\ell=h+1}^H (\mathbf{r}_\ell - \beta \log \boldsymbol{\rho}_\ell) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h \right] - Q_{h,\beta}^{\hat{\pi}_\theta}(x_h, a_h), \\ &\quad - \mathbb{E}_{\hat{\pi}_\theta} \left[\mathbf{r}_h + \sum_{\ell=h+1}^H (\mathbf{r}_\ell - \beta \log \boldsymbol{\rho}_\ell) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = \mathbf{a} \right] + Q_{h,\beta}^{\hat{\pi}_\theta}(x_h, \mathbf{a}). \end{aligned} \tag{113}$$

Thus, to prove the claim, we will bound the absolute differences

$$\left| \mathbb{E}_{\hat{\pi}_\theta} \left[\mathbf{r}_h + \sum_{\ell=h+1}^H (\mathbf{r}_\ell - \beta \log \boldsymbol{\rho}_\ell) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a \right] - Q_{h,\beta}^{\hat{\pi}_\theta}(x_h, a) \right|,$$

for $a \in \{a_h, \mathfrak{a}\}$, and then apply the triangle inequality.

For all $\ell \in [h+1 .. H]$ and $a \in \mathcal{A}$, we can write:

$$\begin{aligned} \mathbb{E}_{\hat{\pi}_\theta} [\log \boldsymbol{\rho}_\ell \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a] &= \mathbb{E}_{\hat{\pi}_\theta} [\mathbb{I}\{M \geq 4C_{\text{cond}}(\bar{\pi}_{\ell,\theta} \mid \mathbf{x}_\ell)^2\} \cdot \log \boldsymbol{\rho}_\ell \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a] \\ &\quad + \mathbb{E}_{\hat{\pi}_\theta} [\mathbb{I}\{M < 4C_{\text{cond}}(\bar{\pi}_{\ell,\theta} \mid \mathbf{x}_\ell)^2\} \cdot \log \boldsymbol{\rho}_\ell \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a]. \end{aligned} \quad (114)$$

Now, by [Lemma F.2](#) (guarantee of SoftmaxSamplerDensity), for all $\ell \in [h+1 .. H]$ there exists $\zeta_\ell : \mathcal{A} \rightarrow \mathbb{R}$ such that for all $a \in \mathcal{A}$

$$\begin{aligned} |\zeta_\ell(a)| &\leq \mathbb{E}_{\hat{\pi}_\theta} \left[\mathbb{I}\{M \geq 4C_{\text{cond}}(\bar{\pi}_{\ell,\theta} \mid \mathbf{x}_\ell)^2\} \cdot C_{\text{cond}}(\bar{\pi}_{\ell,\theta} \mid \mathbf{x}_\ell) \cdot \sqrt{\frac{2}{M}} \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a \right] \\ &\quad + \left(\frac{8B}{\beta} + 4 \log(4M \log(4\tilde{\delta}^{-1})) \right) \cdot \tilde{\delta}, \end{aligned} \quad (115)$$

and

$$\begin{aligned} &\mathbb{E}_{\hat{\pi}_\theta} [\mathbb{I}\{M \geq 4C_{\text{cond}}(\bar{\pi}_{\ell,\theta} \mid \mathbf{x}_\ell)^2\} \cdot \log \boldsymbol{\rho}_\ell \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a] \\ &= \mathbb{E}_{\hat{\pi}_\theta} \left[\mathbb{I}\{M \geq 4C_{\text{cond}}(\bar{\pi}_{\ell,\theta} \mid \mathbf{x}_\ell)^2\} \cdot \log \frac{\hat{\pi}_{\ell,\theta}(\mathbf{a}_\ell \mid \mathbf{x}_\ell)}{\pi_{\ell,\text{ref}}(\mathbf{a}_\ell \mid \mathbf{x}_\ell)} \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a \right] + \zeta_\ell(a). \end{aligned}$$

Plugging this into (114), we get that for all $a \in \mathcal{A}$:

$$\begin{aligned} &\mathbb{E}_{\hat{\pi}_\theta} [\log \boldsymbol{\rho}_\ell \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a] \\ &= \mathbb{E}_{\hat{\pi}_\theta} \left[\mathbb{I}\{M \geq 4C_{\text{cond}}(\bar{\pi}_{\ell,\theta} \mid \mathbf{x}_\ell)^2\} \cdot \log \frac{\hat{\pi}_{\ell,\theta}(\mathbf{a}_\ell \mid \mathbf{x}_\ell)}{\pi_{\ell,\text{ref}}(\mathbf{a}_\ell \mid \mathbf{x}_\ell)} \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a \right] + \zeta_\ell(a) \\ &\quad + \mathbb{E}_{\hat{\pi}_\theta} [\mathbb{I}\{M < 4C_{\text{cond}}(\bar{\pi}_{\ell,\theta} \mid \mathbf{x}_\ell)^2\} \cdot \log \boldsymbol{\rho}_\ell \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a], \\ &= \mathbb{E}_{\hat{\pi}_\theta} \left[\log \frac{\hat{\pi}_{\ell,\theta}(\mathbf{a}_\ell \mid \mathbf{x}_\ell)}{\pi_{\ell,\text{ref}}(\mathbf{a}_\ell \mid \mathbf{x}_\ell)} \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a \right] \\ &\quad - \mathbb{E}_{\hat{\pi}_\theta} \left[\mathbb{I}\{M < 4C_{\text{cond}}(\bar{\pi}_{\ell,\theta} \mid \mathbf{x}_\ell)^2\} \cdot \log \frac{\hat{\pi}_{\ell,\theta}(\mathbf{a}_\ell \mid \mathbf{x}_\ell)}{\pi_{\ell,\text{ref}}(\mathbf{a}_\ell \mid \mathbf{x}_\ell)} \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a \right] + \zeta_\ell(a) \\ &\quad + \mathbb{E}_{\hat{\pi}_\theta} [\mathbb{I}\{M < 4C_{\text{cond}}(\bar{\pi}_{\ell,\theta} \mid \mathbf{x}_\ell)^2\} \cdot \log \boldsymbol{\rho}_\ell \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a]. \end{aligned}$$

Thus, rearranging and using that $\boldsymbol{\rho}_\ell, \frac{\hat{\pi}_{\ell,\theta}(\mathbf{a}_\ell \mid \mathbf{x}_\ell)}{\pi_{\ell,\text{ref}}(\mathbf{a}_\ell \mid \mathbf{x}_\ell)} \in [e^{-2B/\beta}, e^{2B/\beta}]$, for all $\ell \in [h+1 .. H]$ and $(x, a) \in \mathcal{X} \times \mathcal{A}$, we get that for all $\ell \in [h+1 .. H]$ and $a \in \mathcal{A}$:

$$\left| \mathbb{E}_{\hat{\pi}_\theta} \left[\log \boldsymbol{\rho}_\ell - \log \frac{\hat{\pi}_{\ell,\theta}(\mathbf{a}_\ell \mid \mathbf{x}_\ell)}{\pi_{\ell,\text{ref}}(\mathbf{a}_\ell \mid \mathbf{x}_\ell)} \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a \right] \right| \quad (116)$$

$$\leq \frac{4B}{\beta} \mathbb{P}_{\hat{\pi}_\theta} [M < 4C_{\text{cond}}(\bar{\pi}_{\ell,\theta} \mid \mathbf{x}_\ell)^2 \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a] + |\zeta_\ell(a)|.$$

Using this with (113) and the triangle inequality, we get that

$$\begin{aligned} |\mathbb{E}[\xi(x_h, a_h, \mathbf{z}_h(x_h, a_h))]| &\leq 4B \sum_{\ell=h+1}^H \mathbb{P}_{\hat{\pi}_\theta} [M < 4C_{\text{cond}}(\bar{\pi}_{\ell,\theta} \mid \mathbf{x}_\ell)^2 \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h] \\ &\quad + 4B \sum_{\ell=h+1}^H \mathbb{P}_{\hat{\pi}_\theta} [M < 4C_{\text{cond}}(\bar{\pi}_{\ell,\theta} \mid \mathbf{x}_\ell)^2 \mid \mathbf{x}_h = x_h, \mathbf{a}_h = \mathbf{a}] \\ &\quad + H\beta \max_{\ell \in [h+1..H]} |\zeta_\ell(a_h)| + H\beta \max_{\ell \in [h+1..H]} |\zeta_\ell(\mathbf{a})|. \end{aligned}$$

Substituting the bound on ζ_ℓ in (115) completes the proof. \square

Lemma L.3. *Let $h \in [0..H]$ be given. Under the setting of Lemma L.1 and the notation in its proof, we have that for all $\ell \in [h+1..H]$ and $(x_h, a_h) \in \mathcal{X} \times \mathcal{A}$:*

$$\begin{aligned} &\mathbb{E}_{\hat{\pi}_\theta} [D_{\text{KL}}(\hat{\pi}_{\ell,\theta}(\cdot \mid \mathbf{x}_\ell) \parallel \bar{\pi}_{\ell,\theta}(\cdot \mid \mathbf{x}_\ell)) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h] \\ &\leq 4 \left(\frac{R_{\max}}{\beta} + \log(4M \log(4\tilde{\delta}^{-1})) \right) \tilde{\delta} \\ &\quad + \frac{R_{\max}}{\beta} \log(4M \log(4\tilde{\delta}^{-1})) \cdot \mathbb{P}_{\hat{\pi}_\theta} [M < 4C_{\text{cond}}(\bar{\pi}_{\ell,\theta} \mid \mathbf{x}_\ell) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h]. \end{aligned}$$

Proof of Lemma L.3. We have for all $\ell \in [h+1..H]$ and $(x_h, a_h) \in \mathcal{X} \times \mathcal{A}$:

$$\begin{aligned} &\mathbb{E}_{\hat{\pi}_\theta} [D_{\text{KL}}(\hat{\pi}_{\ell,\theta}(\cdot \mid \mathbf{x}_\ell) \parallel \bar{\pi}_{\ell,\theta}(\cdot \mid \mathbf{x}_\ell)) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h] \\ &\leq \mathbb{E}_{\hat{\pi}_\theta} [\mathbb{I}\{M \geq 4C_{\text{cond}}(\bar{\pi}_{\ell,\theta} \mid \mathbf{x}_\ell)\} \cdot D_{\text{KL}}(\hat{\pi}_{\ell,\theta}(\cdot \mid \mathbf{x}_\ell) \parallel \bar{\pi}_{\ell,\theta}(\cdot \mid \mathbf{x}_\ell)) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h] \\ &\quad + \mathbb{E}_{\hat{\pi}_\theta} [\mathbb{I}\{M < 4C_{\text{cond}}(\bar{\pi}_{\ell,\theta} \mid \mathbf{x}_\ell)\} \cdot D_{\text{KL}}(\hat{\pi}_{\ell,\theta}(\cdot \mid \mathbf{x}_\ell) \parallel \bar{\pi}_{\ell,\theta}(\cdot \mid \mathbf{x}_\ell)) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h]. \end{aligned} \tag{117}$$

Now, by Lemma F.1, we have that for all $x \in \mathcal{X}$ and $\ell \in [h+1..H]$, $\frac{\hat{\pi}_{\ell,\theta}(\cdot \mid x)}{\bar{\pi}_{\ell,\theta}(\cdot \mid x)} \leq 4Me^{R_{\max}/\beta} \log(4\tilde{\delta}^{-1})$. Combining this with (117) and using Lemma F.1, we get that for all $\ell \in [h+1..H]$ and $(x_h, a_h) \in \mathcal{X} \times \mathcal{A}$:

$$\begin{aligned} &\mathbb{E}_{\hat{\pi}_\theta} [D_{\text{KL}}(\hat{\pi}_{\ell,\theta}(\cdot \mid \mathbf{x}_\ell) \parallel \bar{\pi}_{\ell,\theta}(\cdot \mid \mathbf{x}_\ell)) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h] \\ &\leq 4 \left(\frac{R_{\max}}{\beta} + \log(4M \log(4\tilde{\delta}^{-1})) \right) \tilde{\delta} \\ &\quad + \mathbb{E}_{\hat{\pi}_\theta} [\mathbb{I}\{M < 4C_{\text{cond}}(\bar{\pi}_{\ell,\theta} \mid \mathbf{x}_\ell)\} \cdot D_{\text{KL}}(\hat{\pi}_{\ell,\theta}(\cdot \mid \mathbf{x}_\ell) \parallel \bar{\pi}_{\ell,\theta}(\cdot \mid \mathbf{x}_\ell)) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h], \\ &\leq 4 \left(\frac{R_{\max}}{\beta} + \log(4M \log(4\tilde{\delta}^{-1})) \right) \tilde{\delta} \\ &\quad + \frac{R_{\max}}{\beta} \log(4M \log(4\tilde{\delta}^{-1})) \cdot \mathbb{P}_{\hat{\pi}_\theta} [M < 4C_{\text{cond}}(\bar{\pi}_{\ell,\theta} \mid \mathbf{x}_\ell) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h]. \end{aligned} \tag{118}$$

This completes the proof. \square

L.2. Guarantee of FitValue for MTSS

Lemma L.4. *Let $\beta, \delta, \varepsilon \in (0, 1)$ and π_{ref} be given and suppose that [Assumption I.1](#) holds with $B > 0$. Consider a call to $\text{MTSS}(\beta, \delta, \varepsilon, \pi_{\text{ref}})$ ([Algorithm 4](#)) and let (λ, ν) and T_{prompt} be as in MTSS . Then, there is an event \mathcal{E}^{reg} of probability at least $1 - \delta/4$ under which for all $t \in [T_{\text{prompt}}]$ and $h \in [H]$, the variables in [Algorithm 4](#) satisfy:*

$$\begin{aligned}
& \|\theta_h^t - \theta_{h,\beta}^*\|_{\Sigma_h^t}^2 \\
& \leq 4\lambda B^2 + \frac{C_1}{N_{\text{reg}}} + C_2 \delta_{\text{rej}} + C_3 \sum_{(x_h, a_h) \in \mathcal{C}_h^t} \sum_{\ell=h+1}^H \mathbb{P}_{\hat{\pi}^t} [M_{\text{rej}} < 16C_{\text{cond}}(\bar{\pi}_\ell^t | \mathbf{x}_\ell)^2 | \mathbf{x}_h = x_h, \mathbf{a}_h = a_h] \\
& \quad + 2304HB\beta \sum_{(x_h, a_h) \in \mathcal{C}_h^t} \sum_{\ell=h+1}^H \mathbb{E}_{\hat{\pi}^t} \left[\min \left(1, C_{\text{cond}}(\bar{\pi}_\ell^t | \mathbf{x}_\ell) \cdot \sqrt{\frac{2}{M_{\text{rej}}}} \right) | \mathbf{x}_h = x_h, \mathbf{a}_h = a_h \right] \\
& \quad + 19200H\beta^2 \sum_{(x_h, a_h) \in \mathcal{C}_h^t} \sum_{\ell=h+1}^H \mathbb{E}_{\hat{\pi}^t} \left[D_{\text{KL}}(\bar{\pi}_\ell^t(\cdot | \mathbf{x}_\ell) \| \bar{\pi}_{\ell,\beta}^*(\cdot | \mathbf{x}_\ell))^2 | \mathbf{x}_h = x_h, \mathbf{a}_h = a_h \right] \\
& \quad + 7680HR_{\text{max}}^2 \sum_{(x_h, a_h) \in \mathcal{C}_h} \sum_{\ell=h+1}^H \mathbb{E}_{\hat{\pi}^t} [B_\ell^t(\mathbf{x}_\ell)^2 | \mathbf{x}_h = x_h, \mathbf{a}_h = a_h], \tag{119}
\end{aligned}$$

where $\hat{\pi}^t$ is as in [Algorithm 4](#);

$$\bar{\pi}_h^t(\cdot | x) \propto \pi_{h,\text{ref}}(\cdot | x) \cdot e^{\bar{\varphi}_h^t(x, \cdot)^\top \theta_h^t / \beta}; \tag{120}$$

$$\bar{\pi}_{h,\beta}^{t,*}(\cdot | x) \propto \pi_{h,\text{ref}}(\cdot | x) \cdot e^{\bar{\varphi}_h^t(x, \cdot)^\top \theta_{h,\beta}^* / \beta}; \tag{121}$$

$$\begin{aligned}
\bar{\varphi}_h^t(x, \cdot) &:= \varphi_h(x, \cdot) \cdot \mathbb{I} \left\{ \|\varphi_h(x, \cdot)\|_{(\Sigma_h^t)^{-1}}^2 \leq \nu^2 \right\}; \\
\varphi_h(\cdot, \cdot) &:= \phi_h(\cdot, \cdot) - \phi_h(\cdot, \mathbf{a}), \tag{122}
\end{aligned}$$

with \mathbf{a} as in [Algorithm 4](#); and

$$\begin{aligned}
C_1 &:= 6400B^2H^2d \log(3N/\delta') + 3840H^2B^2T_{\text{prompt}}, \\
C_2 &:= 19200T_{\text{prompt}}(12R_{\text{max}}^2 + 48\beta^2 \log(4M_{\text{rej}} \log(4\delta_{\text{rej}}^{-1})))^2 + 85H^2B^2) \\
& \quad + 3072T_{\text{prompt}}H^2B\beta \log(4M_{\text{rej}} \log(4\delta_{\text{rej}}^{-1})), \\
C_3 &:= 16H \cdot (8R_{\text{max}}^2 \log(4M_{\text{rej}} \log(4\delta_{\text{rej}}^{-1})))^2 + 200H^2B^2) + 3072HB^2,
\end{aligned}$$

and for $x \in \mathcal{X}$:

$$B_\ell^t(x) := \min \left(1, \max_{\pi \in \{\bar{\pi}_\ell^t, \bar{\pi}_{\ell,\beta}^{t,*}, \pi_{\ell,\beta}^*\}} C_{\text{cond}}(\pi | x) \cdot \mathbb{P}_{\mathbf{a} \sim \pi_{\ell,\text{ref}}(\cdot | x)} \left[\|\varphi_\ell(x, \mathbf{a})\|_{(\Sigma_\ell^t)^{-1}}^2 > \nu^2 \right] \right).$$

Proof of Lemma L.4. Note that from [Line 10](#) of [Algorithm 4](#), the set \mathcal{C}_h^t is a multiset of the form $\mathcal{C} = \bigcup_{i \in n} \{(x_i, a_i), (x_i, \mathbf{a})\}$, and thus satisfies the precondition of [Lemma L.1](#). The result of the lemma thus follows from [Lemma L.1](#) with

$$(\theta_{h+1:H}, \mathcal{C}_h, \Sigma_{h+1:H}, \tilde{\delta}, M, \delta', N) = (\theta_{1:H}^t, \mathcal{C}_h^t, \Sigma_{h+1:H}^t, \delta_{\text{rej}}, M_{\text{rej}}, \delta/(4HT_{\text{prompt}}), N_{\text{reg}}),$$

and [Lemma D.5](#) (essentially the union bound over $t \in [T_{\text{prompt}}]$ and $h \in [H]$). \square

Appendix M. Proof of Theorem J.1

In this section, we provide the proof of the main guarantee for MTSS. Before presenting the proof, we refine the guarantees of `UncertainStateAction` and `FitValue` by incorporating the parameter choices from MTSS and combining the guarantees across layers $h \in [H]$. The final guarantees for `UncertainStateAction` and `FitValue` are stated in [Lemma M.1](#) and [Lemma M.2](#), respectively, after which we proceed to prove [Theorem J.1](#).

Lemma M.1. *Let $\beta, \varepsilon, \delta \in (0, 1)$, $B > 0$, and π_{ref} be given and consider a call to $\text{MTSS}(\beta, \delta, \varepsilon, B, \pi_{\text{ref}})$ ([Algorithm 4](#)). Let (ν, T_{prompt}) and $(\varepsilon_{\text{span}}, \underline{\varepsilon}_{\text{span}}, \mathcal{E}^{\text{span}})$ be as in [Algorithm 4](#) and [Lemma K.2](#), respectively. Finally, let $\mathcal{J}^{\text{span}} := \{t \in [T_{\text{prompt}}] : \|\varphi_h(x_h^t, a_h^t)\|_{(\Sigma_h^t)^{-1}}^2 \leq \nu^2/4, \forall h \in [H]\}$, where $(x_h^t, a_h^t, \Sigma_h^t, \varphi_h)$ are as in [Algorithm 4](#) when the algorithm terminates. Then we have $\mathcal{J}^{\text{span}} \neq \emptyset$, and under the event $\mathcal{E}^{\text{span}}$, we have that for all $j \in \mathcal{J}^{\text{span}}$, the variables in [Algorithm 4](#) satisfy:*

- For all $h \in [H]$, $\ell \in [0 \dots h-1]$, and $(x_\ell, a_\ell) \in \mathcal{C}_\ell^j$,

$$\mathbb{P}_{\hat{\pi}_{\ell+1:h}^j} \left[\|\varphi_h(\mathbf{x}_h, \mathbf{a}_h)\|_{(\Sigma_h^j)^{-1}}^2 > \nu^2 \mid \mathbf{x}_\ell = x_\ell, \mathbf{a}_\ell = a_\ell \right] \leq \varepsilon_{\text{span}}; \quad (123)$$

- There exists $\mathcal{X}_{h,\text{span}}^j \subseteq \mathcal{X}$ such that for all $\ell \in [0 \dots h-1]$ and $(x_\ell, a_\ell) \in \mathcal{C}_\ell^j$, $\mathbb{P}_{\hat{\pi}^j}[\mathbf{x}_h \in \mathcal{X}_{h,\text{span}}^j \mid \mathbf{x}_\ell = x_\ell, \mathbf{a}_\ell = a_\ell] \geq 1 - \varepsilon_{\text{span}}$, and for all $x_h \in \mathcal{X}_{h,\text{span}}^j$:

$$\mathbb{P}_{\mathbf{a} \sim \pi_{h,\text{ref}}(\cdot \mid x_h)} \left[\|\varphi_h(x_h, \mathbf{a})\|_{(\Sigma_h^j)^{-1}}^2 > \nu^2 \right] \leq \underline{\varepsilon}_{\text{span}}. \quad (124)$$

Proof of Lemma M.1. We start by proving that $\mathcal{J}^{\text{span}} \neq \emptyset$. Let (x_h^t, a_h^t) be as in [Algorithm 4](#) and define

$$\forall s \in [T_{\text{prompt}}], \quad u_h^s := \varphi_h(x_h^s, a_h^s) \quad \text{and} \quad \forall t \in [T_{\text{prompt}}], \forall h \in [H], \quad U_h^t := \sum_{s=1}^{t-1} u_h^s (u_h^s)^\top.$$

Note that $\Sigma_h^t = \lambda I + U_h^t$, for all $h \in [H]$ and $t \in [T_{\text{prompt}}]$. By [Lemma D.4](#), we have that:

$$\sum_{t \in [T_{\text{prompt}}]} \sum_{h \in [H]} 1 \wedge \|u_h^t\|_{(\lambda I + U_h^t)^{-1}} \leq H \sqrt{2T_{\text{prompt}} d \log(1 + T_{\text{prompt}}/\lambda)}. \quad (125)$$

Therefore, there exists an $j \in [T_{\text{prompt}}]$ such that for all $h \in [H]$:

$$\begin{aligned} 1 \wedge \|u_h^j\|_{(\lambda I + U_h^j)^{-1}} &\leq \frac{1}{T_{\text{prompt}}} \sum_{t \in [T_{\text{prompt}}]} \sum_{h' \in [H]} \|u_{h'}^t\|_{(\lambda I + U_{h'}^t)^{-1}}, \\ &\leq \frac{H \sqrt{2T_{\text{prompt}} d \log(1 + T_{\text{prompt}}/\lambda)}}{T_{\text{prompt}}}, \\ &\leq \frac{\nu}{2}, \end{aligned} \quad (126)$$

where the last inequality follows from the fact that $T_{\text{prompt}} \geq 8\nu^{-4}dH^2 \log(1 + T_{\text{prompt}}/\lambda)$ ([Section J.1.4](#)). Since $\nu < 1$ ([Section J.1.4](#)), (126) implies that:

$$\forall h \in [H], \quad \|u_h^j\|_{(\lambda I + U_h^j)^{-1}} \leq \frac{\nu}{2}. \quad (127)$$

Using the definitions of u_h^j and U_h^j , this shows that $\|\varphi_h(x_h^j, a_h^j)\|_{(\Sigma_h^j)^{-1}}^2 \leq \nu^2/4$, for all $h \in [H]$, which implies that $\mathcal{J}^{\text{span}} \neq \emptyset$.

We now prove (123) and (124) under $\mathcal{E}^{\text{span}}$. Fix $j \in \mathcal{J}^{\text{span}}$ and condition on $\mathcal{E}^{\text{span}}$. Using Lemma K.2 (and the conditioning on $\mathcal{E}^{\text{span}}$) and the definition of $\mathcal{J}^{\text{span}}$, we have that for all $h \in [H]$, $\ell \in [0..h-1]$, and $(x_\ell, a_\ell) \in \mathcal{C}_\ell^j$:

$$\begin{aligned} & \mathbb{P}_{\hat{\pi}_{\ell+1:h}^j} \left[\|\varphi_h(\mathbf{x}_h, \mathbf{a}_h)\|_{(\Sigma_h^j)^{-1}}^2 > \nu^2 \mid \mathbf{x}_\ell = x_\ell, \mathbf{a}_\ell = a_\ell \right] \\ &= \mathbb{P}_{\hat{\pi}_{\ell+1:h}^j} \left[\|\varphi_h(\mathbf{x}_h, \mathbf{a}_h)\|_{(\Sigma_h^j)^{-1}}^2 > \nu^2 \vee \left(2\|u_h^j\|_{(\lambda I + U_h^j)^{-1}}^2 \right) \mid \mathbf{x}_\ell = x_\ell, \mathbf{a}_\ell = a_\ell \right], \\ &= \mathbb{P}_{\hat{\pi}_{\ell+1:h}^j} \left[\|\varphi_h(\mathbf{x}_h, \mathbf{a}_h)\|_{(\Sigma_h^j)^{-1}}^2 > \nu^2 \vee \left(2\|\varphi_h(x_h^j, a_h^j)\|_{(\Sigma_h^j)^{-1}}^2 \right) \mid \mathbf{x}_\ell = x_\ell, \mathbf{a}_\ell = a_\ell \right], \\ &\leq \varepsilon_{\text{span}}. \end{aligned}$$

Similarity, using Lemma K.2 and the definition of $\mathcal{J}^{\text{span}}$ once more, we have that there exists $\mathcal{X}_{h,\text{span}}^j \subseteq \mathcal{X}$ such that for all $\ell \in [0..h-1]$ and $(x_\ell, a_\ell) \in \mathcal{C}_\ell^j$, $\mathbb{P}_{\hat{\pi}^j}[\mathbf{x}_h \in \mathcal{X}_{h,\text{span}}^j \mid \mathbf{x}_\ell = x_\ell, \mathbf{a}_\ell = a_\ell] \geq 1 - \varepsilon_{\text{span}}$, and for all $x_h \in \mathcal{X}_{h,\text{span}}^j$:

$$\begin{aligned} & \mathbb{P}_{\mathbf{a} \sim \pi_{h,\text{ref}}(\cdot \mid x_h)} \left[\|\varphi_h(x_h, \mathbf{a})\|_{(\Sigma_h^j)^{-1}}^2 > \nu^2 \right] \\ &\leq \mathbb{P}_{\mathbf{a} \sim \pi_{h,\text{ref}}(\cdot \mid x_h)} \left[\|\varphi_h(x_h, \mathbf{a})\|_{(\Sigma_h^j)^{-1}}^2 > \nu^2 \vee \left(2\|\varphi_h(x_h^j, a_h^j)\|_{(\Sigma_h^j)^{-1}}^2 \right) \right], \\ &\leq \varepsilon_{\text{span}}. \end{aligned}$$

This completes the proof. \square

Lemma M.2 (Estimation error). *Let $\beta, \varepsilon, \delta \in (0, 1)$, $B > 0$, and π_{ref} be such that $\varepsilon \leq \beta^2/4$ and suppose that Assumption I.1 holds. Consider a call to MTSS($\beta, \delta, \varepsilon, \pi_{\text{ref}}$) (Algorithm 4), and let \mathcal{E}^{reg} , $\mathcal{E}^{\text{span}}$, and $\mathcal{J}^{\text{span}}$ be as in Lemma L.4, Lemma K.2, and Lemma M.1, respectively. Then, under the event $\mathcal{E}^{\text{reg}} \cap \mathcal{E}^{\text{span}}$, we have that for all $j \in \mathcal{J}^{\text{span}}$ and $h \in [H]$, the parameter vector θ_h^j in Algorithm 4 satisfies*

$$\|\theta_h^j - \theta_{h,\beta}^*\|_{\Sigma_h^j}^2 \leq \varepsilon_{\text{reg}}^2,$$

where Σ_h^j is as in Algorithm 4 at iteration j and $\varepsilon_{\text{reg}}^2 := \varepsilon$.

As we will see shortly in the proof of Lemma M.2, the reason the result holds and why we do not encounter compounding errors from future layers $\ell > h$ is due to the use of KL-regularization. The regularization ensures that errors from subsequent layers are raised to the fourth power, resulting in favorable error propagation across layers.

Proof of Lemma M.2. Let $(\varepsilon_{\text{span}}, \underline{\varepsilon}_{\text{span}})$ be as in Lemma K.2. In this proof, we condition on $\mathcal{E}^{\text{span}} \cap \mathcal{E}^{\text{reg}}$. and fix $j \in \mathcal{J}^{\text{span}}$. We proceed via backward induction over $\ell = H+1, \dots, 1$ to show that

$$\|\theta_\ell^j - \theta_{\ell,\beta}^*\|_{\Sigma_\ell^j}^2 \leq \varepsilon_{\text{reg}}^2. \quad (128)$$

The base case holds trivially by the convention that $\theta_{H+1}^j = \theta_{H+1,\beta}^* = 0$. Let $h \in [H]$ and suppose that (128) holds for $\ell = h+1$. We show that it holds for $\ell = h$.

By Lemma L.4 and the conditioning on \mathcal{E}^{reg} , we have that (with $\bar{\pi}_\ell^j, \bar{\pi}_{\ell,\beta}^{j,*}, C_1, C_2, C_3$, and B_ℓ^j as in Lemma L.4)

$$\begin{aligned}
 & \|\theta_h^j - \theta_{h,\beta}^*\|_{\Sigma_h^j}^2 \\
 & \leq 4\lambda B^2 + \frac{C_1}{N_{\text{reg}}} + C_2 \delta_{\text{rej}} + C_3 \sum_{(x_h, a_h) \in \mathcal{C}_h^j} \sum_{\ell=h+1}^H \mathbb{P}_{\hat{\pi}^j} [M_{\text{rej}} < 16C_{\text{cond}}(\bar{\pi}_\ell^j | \mathbf{x}_\ell)^2 | \mathbf{x}_h = x_h, \mathbf{a}_h = a_h] \\
 & \quad + 2304HB\beta \sum_{(x_h, a_h) \in \mathcal{C}_h^j} \sum_{\ell=h+1}^H \mathbb{E}_{\hat{\pi}^j} \left[\min \left(1, C_{\text{cond}}(\bar{\pi}_\ell^j | \mathbf{x}_\ell) \cdot \sqrt{\frac{2}{M_{\text{rej}}}} \right) | \mathbf{x}_h = x_h, \mathbf{a}_h = a_h \right] \\
 & \quad + 19200H\beta^2 \sum_{(x_h, a_h) \in \mathcal{C}_h^j} \sum_{\ell=h+1}^H \mathbb{E}_{\hat{\pi}^j} \left[D_{\text{KL}}(\bar{\pi}_\ell^j(\cdot | \mathbf{x}_\ell) \| \bar{\pi}_{\ell,\beta}^{j,*}(\cdot | \mathbf{x}_\ell))^2 | \mathbf{x}_h = x_h, \mathbf{a}_h = a_h \right] \\
 & \quad + 7680HR_{\text{max}}^2 \sum_{(x_h, a_h) \in \mathcal{C}_h^j} \sum_{\ell=h+1}^H \mathbb{E}_{\hat{\pi}^j} [B_\ell^j(\mathbf{x}_\ell)^2 | \mathbf{x}_h = x_h, \mathbf{a}_h = a_h]. \tag{129}
 \end{aligned}$$

We start by bounding the KL term on the right-hand side of (129), then bound the terms involving $(B_\ell^j(\mathbf{x}_\ell))$ and C_{cond} , which correspond to distribution shift.

$$B_\ell^j(x) := \max_{\pi \in \{\bar{\pi}_\ell^j, \bar{\pi}_{\ell,\beta}^{j,*}, \pi_{\ell,\beta}^*\}} \min \left(1, C_{\text{cond}}(\pi | x) \cdot \mathbb{P}_{\mathbf{a} \sim \pi_{\ell,\text{ref}}(\cdot | x)} \left[\|\varphi_\ell(x, \mathbf{a})\|_{(\Sigma_\ell^j)^{-1}}^2 > \nu^2 \right] \right),$$

for all $x \in \mathcal{X}$. We call (B_ℓ^j) *distribution shift* terms because they reflect the event in which the algorithm is surprised by a new direction in feature space when the rollout policy changes.

Bounding the KL term. By the induction hypothesis, Lemma N.1 applied to each $\ell > h$ (the precondition $\|\theta_{\ell,\beta}^* - \theta_\ell^j\| \leq \beta/\nu$ of Lemma N.1 is satisfied thanks to the induction hypothesis and $\varepsilon_{\text{reg}} = \varepsilon^{1/2} \leq \beta/2$), and Jensen's inequality, we have that for all $(x_h, a_h) \in \mathcal{C}_h^j$ and $\ell \in [h+1 .. H]$:

$$\begin{aligned}
 & \mathbb{E}_{\hat{\pi}^j} \left[D_{\text{KL}}(\bar{\pi}_\ell^j(\cdot | \mathbf{x}_\ell) \| \bar{\pi}_{\ell,\beta}^{j,*}(\cdot | \mathbf{x}_\ell))^2 | \mathbf{x}_h = x_h, \mathbf{a}_h = a_h \right] \\
 & \leq \mathbb{E}_{\hat{\pi}^j} \left[\left(\beta^{-1} \sum_{a \in \mathcal{A}} \bar{\pi}_\ell^j(a | \mathbf{x}_\ell) \cdot \|\bar{\varphi}_\ell^j(\mathbf{x}_\ell, a)\|_{(\Sigma_\ell^j)^{-1}}^2 \right)^2 | \mathbf{x}_h = x_h, \mathbf{a}_h = a_h \right] \|\theta_{\ell,\beta}^* - \theta_\ell^j\|_{\Sigma_\ell^j}^4, \tag{130}
 \end{aligned}$$

where $\bar{\varphi}^j$ is as in Lemma L.4. Now, by Lemma F.3, we have that for any $x \in \mathcal{X}$ and $\ell \in [h+1 .. H]$:

$$\begin{aligned}
 & \left| \sum_{a \in \mathcal{A}} \bar{\pi}_\ell^j(a | x) \cdot \|\bar{\varphi}_\ell^j(x, a)\|_{(\Sigma_\ell^j)^{-1}}^2 - \sum_{a \in \mathcal{A}} \hat{\pi}_\ell^j(a | x) \cdot \|\bar{\varphi}_\ell^j(x, a)\|_{(\Sigma_\ell^j)^{-1}}^2 \right| \\
 & \leq \frac{4}{\lambda} \delta_{\text{rej}} + \frac{4}{\lambda} \cdot \mathbb{I} \{M_{\text{rej}} < 4C_{\text{cond}}(\bar{\pi}_\ell^j | x)\}. \tag{131}
 \end{aligned}$$

Plugging this into (130) and using Jensen's inequality, we get that for all $(x_h, a_h) \in \mathcal{C}_h^j$ and $\ell \in [h+1 .. H]$:

$$\mathbb{E}_{\hat{\pi}^j} \left[D_{\text{KL}}(\bar{\pi}_\ell^j(\cdot | \mathbf{x}_\ell) \| \bar{\pi}_{\ell,\beta}^{j,*}(\cdot | \mathbf{x}_\ell))^2 | \mathbf{x}_h = x_h, \mathbf{a}_h = a_h \right]$$

$$\begin{aligned}
&\leq \frac{4}{\beta^2} \cdot \mathbb{E}_{\hat{\pi}^j} \left[\|\bar{\varphi}_\ell^j(\mathbf{x}_\ell, \mathbf{a}_\ell)\|_{(\Sigma_\ell^j)^{-1}}^4 \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h \right] \cdot \|\theta_{\ell,\beta}^* - \theta_\ell^j\|_{\Sigma_\ell^j}^4 \\
&\quad + \left(\frac{32\delta_{\text{rej}}^2}{\lambda^2\beta^2} + \frac{32}{\beta^2\lambda^2} \mathbb{P}_{\hat{\pi}^j} [M_{\text{rej}} < 4C_{\text{cond}}(\bar{\pi}_\ell^j \mid \mathbf{x}_\ell) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h] \right) \cdot \|\theta_{\ell,\beta}^* - \theta_\ell^j\|_{\Sigma_\ell^j}^4, \\
&\leq \frac{\nu^4}{\beta^2} + \frac{16}{\lambda^2\beta^2} \cdot \mathbb{P}_{\hat{\pi}^j} \left[\|\varphi_\ell(\mathbf{x}_\ell, \mathbf{a}_\ell)\|_{(\Sigma_\ell^j)^{-1}}^2 > \nu^2 \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h \right] \cdot \|\theta_\ell^j - \theta_{\ell,\beta}^*\|_{\Sigma_\ell^j}^4 \\
&\quad + \left(\frac{32\delta_{\text{rej}}^2}{\lambda^2\beta^2} + \frac{32}{\beta^2\lambda^2} \mathbb{P}_{\hat{\pi}^j} [M_{\text{rej}} < 4C_{\text{cond}}(\bar{\pi}_\ell^j \mid \mathbf{x}_\ell) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h] \right) \cdot \|\theta_{\ell,\beta}^* - \theta_\ell^j\|_{\Sigma_\ell^j}^4,
\end{aligned} \tag{132}$$

where the last inequality follows by the fact that $\sigma_{\min}(\Sigma_\ell^j) \geq \lambda$ and $\|\varphi_\ell(x, a)\| \leq 2$, for all $\ell \in [H]$, $(x, a) \in \mathcal{X} \times \mathcal{A}$ (follows by [Assumption I.1](#)). And so, by [Lemma M.1](#) (in particular (123)) and the induction hypothesis again (to bound $\|\theta_{\ell,\beta}^* - \theta_\ell^j\|_{\Sigma_\ell^j}^4$), we have that for all $(x_h, a_h) \in \mathcal{C}_h^j$ and $\ell \in [h+1..H]$:

$$\begin{aligned}
&\mathbb{E}_{\hat{\pi}^j} \left[D_{\text{KL}} \left(\bar{\pi}_\ell^j(\cdot \mid \mathbf{x}_\ell) \parallel \bar{\pi}_{\ell,\beta}^{j,*}(\cdot \mid \mathbf{x}_\ell) \right)^2 \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h \right] \\
&\leq (\lambda^2\nu^4 + 16\varepsilon_{\text{span}} + 32\delta_{\text{rej}}^2 + 32\mathbb{P}_{\hat{\pi}^j} [M_{\text{rej}} < 4C_{\text{cond}}(\bar{\pi}_\ell^j \mid \mathbf{x}_\ell) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h]) \cdot \frac{\varepsilon_{\text{reg}}^4}{\beta^2\lambda^2}.
\end{aligned} \tag{133}$$

We now bound the “distribution shift” terms (B_ℓ^j) appearing in (133) and on the right-hand side of (129).

Bounding the distribution shift terms. Let $\mathcal{X}_{\ell,\text{span}}^j$ and $(\varepsilon_{\text{span}}, \underline{\varepsilon}_{\text{span}})$ be as in [Lemma M.1](#) and [Lemma K.2](#), respectively. By [Lemma M.1](#), we have that for all $\ell \in [h+1..H]$ and $(x_h, a_h) \in \mathcal{C}_h^j$,

$$\mathbb{P}_{\hat{\pi}^j} \left[\mathbf{x}_\ell \in \mathcal{X}_{\ell,\text{span}}^j \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h \right] \geq 1 - \varepsilon_{\text{span}}, \tag{134}$$

and for all $x \in \mathcal{X}_{\ell,\text{span}}^j$:

$$\mathbb{P}_{\mathbf{a} \sim \pi_{\ell,\text{ref}}(\cdot \mid x)} \left[\|\varphi_\ell(x, \mathbf{a})\|_{(\Sigma_\ell^j)^{-1}}^2 > \nu^2 \right] \leq \underline{\varepsilon}_{\text{span}} \leq \frac{1}{4C_{\text{cond}}(\pi_{\ell,\beta}^*)}, \tag{135}$$

where the last inequality follows by the fact that $\bar{N}_{\text{span}} \geq 4C_{\text{cond}}(\pi_{\ell,\beta}^*)$ (see parameter choices in [Algorithm 4](#)). On the other hand, by [Lemma M.3](#) (stated and proven in the sequel), we have that for all $\ell \in [h+1..H]$ and $(x, a) \in \mathcal{X}_{\ell,\text{span}}^j \times \mathcal{A}$:

$$\begin{aligned}
\frac{\bar{\pi}_\ell^j(a \mid x)}{\pi_{\ell,\text{ref}}(a \mid x)} \vee \frac{\bar{\pi}_{\ell,\beta}^{j,*}(a \mid x)}{\pi_{\ell,\text{ref}}(a \mid x)} &\leq 2e^{\frac{4\nu}{\beta}\|\theta_\ell^j - \theta_{\ell,\beta}^*\|_{\Sigma_\ell^j}} \cdot \frac{\pi_{\ell,\beta}^*(a \mid x)}{\pi_{\ell,\text{ref}}(a \mid x)}, \\
&\leq 2e^{\frac{\pi_{\ell,\beta}^*(a \mid x)}{\pi_{\ell,\text{ref}}(a \mid x)}},
\end{aligned} \tag{136}$$

where the last step follows by the induction hypothesis and the fact that $\varepsilon_{\text{reg}} \leq \beta/(4\nu)$. Therefore, we have that for all $\ell \in [h+1..H]$, $x \in \mathcal{X}_{\ell,\text{span}}^j$, and $\pi \in \{\bar{\pi}_\ell^j, \bar{\pi}_{\ell,\beta}^{j,*}\}$:

$$C_{\text{cond}}(\pi \mid x) \leq 2eC_{\text{cond}}(\pi_{\ell,\beta}^*). \tag{137}$$

Thus, combining (137) and (135), we get that for all $x \in \mathcal{X}_{\ell, \text{span}}^j$:

$$\begin{aligned} B_\ell^j(x) &= \max_{\pi \in \{\bar{\pi}_\ell^j, \bar{\pi}_{\ell, \beta}^{j, \star}, \pi_{\ell, \beta}^{\star}\}} \min \left(1, C_{\text{cond}}(\pi \mid x) \cdot \mathbb{E}_{\mathbf{a} \sim \pi_{\ell, \text{ref}}(\cdot \mid x)} \left[\|\varphi_\ell(x, \mathbf{a})\|_{(\Sigma_\ell^j)^{-1}}^2 > \nu \right] \right), \\ &\leq \min \left(1, 2eC_{\text{cond}}(\pi_{\ell, \beta}^{\star} \mid x) \cdot \mathbb{E}_{\mathbf{a} \sim \pi_{\ell, \text{ref}}(\cdot \mid x)} \left[\|\varphi_\ell(x, \mathbf{a})\|_{(\Sigma_\ell^j)^{-1}}^2 > \nu \right] \right), \\ &\leq 2eC_{\text{cond}}(\pi_{\ell, \beta}^{\star}) \cdot \varepsilon_{\text{span}}, \end{aligned}$$

where the last step follows by (135) and that $\varepsilon_{\text{span}} \leq 1$. Thus, using (134), we have that for all $\ell \in [h+1 .. H]$ and $(x_h, a_h) \in \mathcal{C}_h^j$:

$$\begin{aligned} &\mathbb{E}_{\hat{\pi}^j} [B_\ell^j(\mathbf{x}_\ell) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h] \\ &\leq \mathbb{E}_{\hat{\pi}^j} [\mathbb{I}\{\mathbf{x}_\ell \in \mathcal{X}_{\ell, \text{span}}^j\} \cdot B_\ell^j(\mathbf{x}_\ell) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h] \\ &\quad + \mathbb{E}_{\hat{\pi}^j} [\mathbb{I}\{\mathbf{x}_\ell \notin \mathcal{X}_{\ell, \text{span}}^j\} \cdot B_\ell^j(\mathbf{x}_\ell) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h], \\ &\leq (1 + 2eC_{\text{cond}}(\pi_{\ell, \beta}^{\star})) \cdot \varepsilon_{\text{span}}. \end{aligned} \tag{138}$$

Now, by (136) and the fact that $M_{\text{rej}} \geq 32e(1 \vee C_{\text{cond}}(\pi_{\ell, \beta}^{\star}))^2$ (see parameter choice in Section J.1.4), we have that for all $\ell \in [h+1 .. H]$: if $x \in \mathcal{X}_{\ell, \text{span}}^j$ then

$$M_{\text{rej}} \geq (4C_{\text{cond}}(\bar{\pi}_\ell^j \mid x)) \vee (16C_{\text{cond}}(\bar{\pi}_\ell^j \mid x)^2)$$

and so by (135), we have for all $(x_h, a_h) \in \mathcal{C}_h^j$ and $\ell \in [h+1 .. H]$:

$$\begin{aligned} 1 - \varepsilon_{\text{span}} &\leq \mathbb{P}_{\hat{\pi}^j} [\mathbf{x}_\ell \in \mathcal{X}_{\ell, \text{span}}^j \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h], \\ &\leq \mathbb{P}_{\hat{\pi}^j} [M_{\text{rej}} \geq (4C_{\text{cond}}(\bar{\pi}_\ell^j \mid \mathbf{x}_\ell)) \vee (16C_{\text{cond}}(\bar{\pi}_\ell^j \mid \mathbf{x}_\ell)^2) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h]. \end{aligned}$$

This implies that for all $\ell \in [h+1 .. H]$ and $(x_h, a_h) \in \mathcal{C}_h^j$:

$$\mathbb{P}_{\hat{\pi}^j} [M_{\text{rej}} < 4C_{\text{cond}}(\bar{\pi}_\ell^j \mid \mathbf{x}_\ell) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h] \leq \varepsilon_{\text{span}}, \tag{139}$$

and

$$\mathbb{P}_{\hat{\pi}^j} [M_{\text{rej}} < 16C_{\text{cond}}(\bar{\pi}_\ell^j \mid \mathbf{x}_\ell)^2 \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h] \leq \varepsilon_{\text{span}}. \tag{140}$$

Finally, we have that for all $\ell \in [h+1 .. H]$ and $(x_h, a_h) \in \mathcal{C}_h^j$:

$$\begin{aligned} &\mathbb{E}_{\hat{\pi}^j} \left[\min \left(1, C_{\text{cond}}(\bar{\pi}_\ell^j \mid \mathbf{x}_\ell) \cdot \sqrt{\frac{2}{M_{\text{rej}}}} \right) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h \right] \\ &\leq \mathbb{E}_{\hat{\pi}^j} \left[\mathbb{I}\{\mathbf{x}_\ell \in \mathcal{X}_{\ell, \text{span}}^j\} \cdot \min \left(1, C_{\text{cond}}(\bar{\pi}_\ell^j \mid \mathbf{x}_\ell) \cdot \sqrt{\frac{2}{M_{\text{rej}}}} \right) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h \right] \\ &\quad + \mathbb{E}_{\hat{\pi}^j} \left[\mathbb{I}\{\mathbf{x}_\ell \notin \mathcal{X}_{\ell, \text{span}}^j\} \cdot \min \left(1, C_{\text{cond}}(\bar{\pi}_\ell^j \mid \mathbf{x}_\ell) \cdot \sqrt{\frac{2}{M_{\text{rej}}}} \right) \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h \right], \end{aligned}$$

and so by (136) and (134):

$$\leq 2eC_{\text{cond}}(\pi_{\ell, \beta}^{\star}) \cdot \sqrt{\frac{2}{M_{\text{rej}}}} + \mathbb{P}_{\hat{\pi}^j} [\mathbf{x}_\ell \notin \mathcal{X}_{\ell, \text{span}}^j \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h],$$

$$\leq 2eC_{\text{cond}}(\pi_{\ell,\beta}^*) \cdot \sqrt{\frac{2}{M_{\text{rej}}}} + \varepsilon_{\text{span}}. \quad (141)$$

Putting it all together. Combining (141), (139), (140), and (138) with (133) and (129) and using that $|\mathcal{C}_h^j| \leq 2T_{\text{prompt}}$, we get

$$\begin{aligned} & \|\theta_h^j - \theta_{h,\beta}^*\|_{\Sigma_h^j}^2 \\ & \leq 4\lambda B^2 + \frac{C_1}{N_{\text{reg}}} + C_2\delta_{\text{rej}} + C_3HT_{\text{prompt}}\varepsilon_{\text{span}} \\ & \quad + 1536eH^2B\beta T_{\text{prompt}} \left(\max_{\ell \in [H]} C_{\text{cond}}(\pi_{\ell,\beta}^*) \cdot \sqrt{\frac{2}{M_{\text{rej}}}} + \varepsilon_{\text{span}} \right) \\ & \quad + 3200H^2\beta^2 T_{\text{prompt}} (\lambda^2\nu^4 + 16\varepsilon_{\text{span}} + 32\delta_{\text{rej}}^2 + 32\varepsilon_{\text{span}}) \cdot \frac{\varepsilon_{\text{reg}}^4}{\beta^2\lambda^2} \\ & \quad + 7680H^2R_{\text{max}}^2 T_{\text{prompt}} (1 + 2eC_{\text{cond}}(\pi_{\ell,\beta}^*))^2 \cdot \varepsilon_{\text{span}}^2, \\ & \leq \varepsilon_{\text{reg}}^2, \end{aligned}$$

where the last inequality follows by the parameter choices in Algorithm 4. This completes the induction and implies the desired result. \square

Lemma M.3 (Helper lemma for estimation error). *Let $h \in [0 .. H]$ be given. Consider the setting of Lemma M.2 and let $j \in \mathcal{J}^{\text{span}}$ with $\mathcal{J}^{\text{span}}$ as in Lemma M.1. Further, let $\mathcal{X}_{\ell,\text{span}}^j$ and $(\varepsilon_{\text{span}}, \underline{\varepsilon}_{\text{span}})$ be as in Lemma M.1 and Lemma K.2, respectively. Then, we have that for all $\ell \in [h+1 .. H]$ and $(x, a) \in \mathcal{X}_{\ell,\text{span}}^j \times \mathcal{A}$:*

$$\frac{\bar{\pi}_{\ell}^j(a | x)}{\pi_{\ell,\text{ref}}(a | x)} \vee \frac{\bar{\pi}_{\ell,\beta}^{j,*}(a | x)}{\pi_{\ell,\text{ref}}(a | x)} \leq 2e^{\frac{4\nu}{\beta} \|\theta_{\ell}^j - \theta_{\ell,\beta}^*\|_{\Sigma_{\ell}^j}} \cdot \frac{\pi_{\ell,\beta}^*(a | x)}{\pi_{\ell,\text{ref}}(a | x)}.$$

Proof of Lemma M.3. By Lemma M.1, we have that for all $\ell \in [h+1 .. H]$ and $(x_h, a_h) \in \mathcal{C}_h^j$,

$$\mathbb{P}_{\hat{\pi}^j} \left[\mathbf{x}_{\ell} \in \mathcal{X}_{\ell,\text{span}}^j \mid \mathbf{x}_h = x_h, \mathbf{a}_h = a_h \right] \geq 1 - \varepsilon_{\text{span}}, \quad (142)$$

and for all $x \in \mathcal{X}_{\ell,\text{span}}^j$:

$$\mathbb{P}_{\mathbf{a} \sim \pi_{\ell,\text{ref}}(\cdot | x)} \left[\|\varphi_{\ell}(x, \mathbf{a})\|_{(\Sigma_{\ell}^j)^{-1}}^2 > \nu^2 \right] \leq \underline{\varepsilon}_{\text{span}} \leq \frac{1}{4C_{\text{cond}}(\pi_{\ell,\beta}^*)}, \quad (143)$$

where the last inequality follows by the fact that $\bar{N}_{\text{span}} \geq 4C_{\text{cond}}(\pi_{\beta}^*)$ (see parameter choices in Algorithm 4). Now, by Lemma N.2 and Lemma M.1, we have that for all $\ell \in [h+1 .. H]$ and $(x, a) \in \mathcal{X}_{\ell,\text{span}}^j \times \mathcal{A}$:

$$\frac{\bar{\pi}_{\ell}^j(a | x)}{\pi_{\ell,\text{ref}}(a | x)} \vee \frac{\bar{\pi}_{\ell,\beta}^{j,*}(a | x)}{\pi_{\ell,\text{ref}}(a | x)}$$

$$\begin{aligned}
 &\leq \min \left\{ \frac{e^{\frac{4\nu}{\beta} \|\theta_\ell^j - \theta_{\ell,\beta}^*\|_{\Sigma_\ell^j}}}{1 - C_{\text{cond}}(\pi_{\ell,\beta}^*) \cdot \mathbb{P}_{\mathbf{a} \sim \pi_{\ell,\text{ref}}(\cdot | x)} \left[\|\varphi_\ell(x, \mathbf{a})\|_{(\Sigma_\ell^j)^{-1}}^2 > \nu^2 \right]}, \frac{\pi_{\ell,\beta}^*(a | x)}{\pi_{\ell,\text{ref}}(a | x)}, e^{2B/\beta} \right\}, \\
 &\leq 2e^{\frac{4\nu}{\beta} \|\theta_\ell^j - \theta_{\ell,\beta}^*\|_{\Sigma_\ell^j}} \cdot \frac{\pi_{\ell,\beta}^*(a | x)}{\pi_{\ell,\text{ref}}(a | x)},
 \end{aligned} \tag{144}$$

where the last step follows by (143). \square

Proof of Theorem J.1. In this proof, we let \mathcal{E}^{reg} , $\mathcal{E}^{\text{span}}$, and $\mathcal{J}^{\text{span}}$ be as in Lemma L.4, Lemma K.2, and Lemma M.1, respectively, and we condition on $\mathcal{E}^{\text{reg}} \cap \mathcal{E}^{\text{span}}$. Further, let $j \in [T]$ be the index Algorithm 4 for which the algorithm returns $\hat{\pi}^j$, and note that $j \in \mathcal{J}^{\text{span}}$. By the performance difference lemma (Lemma N.5) and Lemma N.4, the policy $\hat{\pi}^j$ satisfies

$$\begin{aligned}
 &J_\beta(\pi_\beta^*) - J_\beta(\hat{\pi}_{1:H}^j) \\
 &= \sum_{h=1}^H \mathbb{E}_{\hat{\pi}^j} \left[\sum_{a \in \mathcal{A}} \pi_{h,\beta}^*(a | \mathbf{x}_h) \cdot \left(Q_{h,\beta}^*(\mathbf{x}_h, a) - \beta \cdot \log \frac{\pi_{h,\beta}^*(a | \mathbf{x}_h)}{\pi_{h,\text{ref}}(a | \mathbf{x}_h)} \right) \right] \\
 &\quad - \sum_{h=1}^H \mathbb{E}_{\hat{\pi}^j} \left[\sum_{a \in \mathcal{A}} \hat{\pi}_h^j(a | \mathbf{x}_h) \cdot \left(Q_{h,\beta}^*(\mathbf{x}_h, a) - \beta \cdot \log \frac{\hat{\pi}_h^j(a | \mathbf{x}_h)}{\pi_{h,\text{ref}}(a | \mathbf{x}_h)} \right) \right], \\
 &= \sum_{h=1}^H \mathbb{E}_{\hat{\pi}^j} \left[\sum_{a \in \mathcal{A}} \pi_{h,\beta}^*(a | \mathbf{x}_h) \cdot \left(Q_{h,\beta}^*(\mathbf{x}_h, a) - \beta \cdot \log \frac{\pi_{h,\beta}^*(a | \mathbf{x}_h)}{\pi_{h,\text{ref}}(a | \mathbf{x}_h)} \right) \right] \\
 &\quad - \sum_{h=1}^H \mathbb{E}_{\hat{\pi}^j} \left[\sum_{a \in \mathcal{A}} \hat{\pi}_h^j(a | \mathbf{x}_h) \cdot \left(Q_{h,\beta}^*(\mathbf{x}_h, a) - \beta \cdot \log \frac{\hat{\pi}_h^j(a | \mathbf{x}_h)}{\pi_{h,\text{ref}}(a | \mathbf{x}_h)} \right) \right] \\
 &\quad + \sum_{\ell=1}^H \beta \cdot \mathbb{E}_{\hat{\pi}^j} [D_{\text{KL}}(\hat{\pi}_\ell^j(\cdot | \mathbf{x}_\ell) \| \bar{\pi}_\ell^j(\cdot | \mathbf{x}_\ell))],
 \end{aligned} \tag{145}$$

where $\bar{\pi}^j$ is as in Lemma L.4. Now by Lemma L.3 (instantiated with $h = 0$, $\hat{\pi}_\theta = \hat{\pi}^j$, and $\bar{\pi}_\theta = \bar{\pi}^j$), we can bound the KL term in (145) as follows: for all $\ell \in [H]$,

$$\begin{aligned}
 &\mathbb{E}_{\hat{\pi}^j} [D_{\text{KL}}(\hat{\pi}_\ell^j(\cdot | \mathbf{x}_\ell) \| \bar{\pi}_\ell^j(\cdot | \mathbf{x}_\ell))] \\
 &\leq 4 \left(\frac{R_{\max}}{\beta} + \log(4M_{\text{rej}} \log(4\delta_{\text{rej}}^{-1})) \right) \delta_{\text{rej}} \\
 &\quad + \frac{R_{\max}}{\beta} \log(4M_{\text{rej}} \log(4\delta_{\text{rej}}^{-1})) \cdot \mathbb{P}_{\hat{\pi}^j} [M_{\text{rej}} < 4C_{\text{cond}}(\bar{\pi}_\ell^j | \mathbf{x}_\ell)].
 \end{aligned} \tag{146}$$

Now, note that for all $\ell \in [H]$,

$$\left| Q_{\ell,\beta}^*(\cdot, \cdot) - \beta \cdot \log \frac{\bar{\pi}_\ell^j(\cdot | \cdot)}{\pi_{\ell,\text{ref}}(\cdot | \cdot)} \right| \leq BH + 2H\beta \sup_{(x,a) \in \mathcal{X} \times \mathcal{A}} \left| \log \frac{\bar{\pi}_{\ell,\theta}(a | x)}{\pi_{\ell,\text{ref}}(a | x)} \right| \leq 5BH,$$

since $e^{-2B/\beta} \leq \frac{\pi_{\ell,\theta}(\cdot|\cdot)}{\pi_{\ell,\text{ref}}(\cdot|\cdot)} \leq e^{2B/\beta}$ for all $\theta_\ell \in \mathbb{B}(B)$. Thus, by [Lemma F.3](#), we have that for all $\ell \in [H]$:

$$\begin{aligned} & \left| \mathbb{E}_{\hat{\pi}^j} \left[\sum_{a \in \mathcal{A}} \hat{\pi}_\ell^j(a | \mathbf{x}_\ell) \cdot \left(Q_{\ell,\beta}^*(\mathbf{x}_\ell, a) - \beta \cdot \log \frac{\pi_\ell^j(a | \mathbf{x}_\ell)}{\pi_{\ell,\text{ref}}(a | \mathbf{x}_\ell)} \right) \right] \right. \\ & \quad \left. - \mathbb{E}_{\hat{\pi}^j} \left[\sum_{a \in \mathcal{A}} \bar{\pi}_\ell^j(a | \mathbf{x}_\ell) \cdot \left(Q_{\ell,\beta}^*(\mathbf{x}_\ell, a) - \beta \cdot \log \frac{\pi_\ell^j(a | \mathbf{x}_\ell)}{\pi_{\ell,\text{ref}}(a | \mathbf{x}_\ell)} \right) \right] \right| \\ & \leq 5BH\delta_{\text{rej}} + 5BH \cdot \mathbb{P}_{\hat{\pi}^j} [M_{\text{rej}} < 4C_{\text{cond}}(\bar{\pi}_\ell^j | \mathbf{x})]. \end{aligned} \quad (147)$$

Combining this with [\(145\)](#) and [\(146\)](#), we get that

$$\begin{aligned} & J_\beta(\pi_\beta^*) - J_\beta(\hat{\pi}_{1:H}^j) \\ & \leq \sum_{h=1}^H \mathbb{E}_{\hat{\pi}^j} \left[\sum_{a \in \mathcal{A}} \pi_{h,\beta}^*(a | \mathbf{x}_h) \cdot \left(Q_{h,\beta}^*(\mathbf{x}_h, a) - \beta \cdot \log \frac{\pi_{h,\beta}^*(a | \mathbf{x}_h)}{\pi_{h,\text{ref}}(a | \mathbf{x}_h)} \right) \right] \\ & \quad - \sum_{h=1}^H \mathbb{E}_{\hat{\pi}^j} \left[\sum_{a \in \mathcal{A}} \bar{\pi}_h^j(a | \mathbf{x}_h) \cdot \left(Q_{h,\beta}^*(\mathbf{x}_h, a) - \beta \cdot \log \frac{\pi_h^j(a | \mathbf{x}_h)}{\pi_{h,\text{ref}}(a | \mathbf{x}_h)} \right) \right] \\ & \quad + 4H \left(R_{\max} + 5HB/4 + \beta \log(4M_{\text{rej}} \log(4\delta_{\text{rej}}^{-1})) \right) \cdot \delta_{\text{rej}} \\ & \quad + B \left(\log(4M_{\text{rej}} \log(4\delta_{\text{rej}}^{-1})) + 5H \right) \sum_{\ell=1}^H \mathbb{P}_{\hat{\pi}^j} [M_{\text{rej}} < 4C_{\text{cond}}(\bar{\pi}_\ell^j | \mathbf{x}_\ell)], \end{aligned} \quad (148)$$

and so by [Lemma N.3](#), we have

$$\begin{aligned} & \leq \beta \sum_{\ell=1}^H \mathbb{E}_{\hat{\pi}^j} \left[D_{\text{KL}} \left(\bar{\pi}_\ell^j(\cdot | \mathbf{x}_\ell) \parallel \bar{\pi}_{\ell,\beta}^{j,*}(\cdot | \mathbf{x}_\ell) \right) \right] \\ & \quad + 4H \left(R_{\max} + 5HB/4 + \beta \log(4M_{\text{rej}} \log(4\delta_{\text{rej}}^{-1})) \right) \cdot \delta_{\text{rej}} \\ & \quad + B \left(\log(4M_{\text{rej}} \log(4\delta_{\text{rej}}^{-1})) + 5H \right) \sum_{\ell=1}^H \mathbb{P}_{\hat{\pi}^j} [M_{\text{rej}} < 4C_{\text{cond}}(\bar{\pi}_\ell^j | \mathbf{x}_\ell)], \end{aligned} \quad (149)$$

where $\bar{\pi}_{\ell,\beta}^{j,*}$ is as in [Lemma L.4](#). We start with bounding the KL terms in [Eq. \(149\)](#).

Bounding the KL term. We now bound the KL term in [\(149\)](#). The argument closely follows the proof of [Lemma M.2](#), where we bounded the expectation of the squared KL term. The key difference here is that [\(149\)](#) does not include a squared term. By [Lemma N.1](#) and [Lemma M.2](#) (which implies that $\|\theta_{\ell,\beta}^* - \theta_\ell^j\|_{\Sigma_\ell^j} \leq \beta/\nu$ —the precondition of [Lemma N.1](#)), we have that for all $\ell \in [H]$:

$$\begin{aligned} & \mathbb{E}_{\hat{\pi}^j} \left[D_{\text{KL}} \left(\bar{\pi}_\ell^j(\cdot | \mathbf{x}_\ell) \parallel \bar{\pi}_{\ell,\beta}^{j,*}(\cdot | \mathbf{x}_\ell) \right) \right] \\ & \leq \mathbb{E}_{\hat{\pi}^j} \left[\beta^{-1} \sum_{a \in \mathcal{A}} \bar{\pi}_\ell^j(a | \mathbf{x}_\ell) \cdot \|\bar{\varphi}_\ell^j(\mathbf{x}_\ell, a)\|_{(\Sigma_\ell^j)^{-1}}^2 \right] \cdot \|\theta_{\ell,\beta}^* - \theta_\ell^j\|_{\Sigma_\ell^j}^2, \end{aligned} \quad (150)$$

where $\bar{\varphi}^j$ is as in [Lemma L.4](#). Now by [Lemma F.3](#), we have that for any $x \in \mathcal{X}$ and $\ell \in [H]$:

$$\begin{aligned} & \left| \sum_{a \in \mathcal{A}} \bar{\pi}_\ell^j(a | x) \cdot \|\bar{\varphi}_\ell^j(x, a)\|_{(\Sigma_\ell^j)^{-1}}^2 - \sum_{a \in \mathcal{A}} \hat{\pi}_\ell^j(a | x) \cdot \|\bar{\varphi}_\ell^j(x, a)\|_{(\Sigma_\ell^j)^{-1}}^2 \right| \\ & \leq \frac{4}{\lambda} \delta_{\text{rej}} + \frac{4}{\lambda} \cdot \mathbb{I} \{M_{\text{rej}} < 4C_{\text{cond}}(\bar{\pi}_\ell^j | x)\}. \end{aligned}$$

Plugging this into [\(150\)](#) and using Jensen inequality, we get that for all $\ell \in [H]$:

$$\begin{aligned} & \mathbb{E}_{\hat{\pi}^j} \left[D_{\text{KL}} \left(\bar{\pi}_\ell^j(\cdot | \mathbf{x}_\ell) \parallel \bar{\pi}_{\ell, \beta}^{j, \star}(\cdot | \mathbf{x}_\ell) \right) \right] \\ & \leq \frac{1}{\beta} \cdot \mathbb{E}_{\hat{\pi}^j} \left[\|\bar{\varphi}_\ell^j(\mathbf{x}_\ell, \mathbf{a}_\ell)\|_{(\Sigma_\ell^j)^{-1}}^2 \cdot \|\theta_{\ell, \beta}^* - \theta_\ell^j\|_{\Sigma_\ell^j}^2 \right. \\ & \quad \left. + \left(\frac{4\delta_{\text{rej}}}{\lambda\beta} + \frac{4}{\beta\lambda} \mathbb{P}_{\hat{\pi}^j} [M_{\text{rej}} < 4C_{\text{cond}}(\bar{\pi}_\ell^j | \mathbf{x}_\ell)] \right) \cdot \|\theta_{\ell, \beta}^* - \theta_\ell^j\|_{\Sigma_\ell^j}^2 \right], \\ & \leq \frac{\nu^2}{\beta} + \frac{4}{\lambda\beta} \cdot \mathbb{P}_{\hat{\pi}^j} \left[\|\varphi_\ell(\mathbf{x}_\ell, \mathbf{a}_\ell)\|_{(\Sigma_\ell^j)^{-1}}^2 > \nu^2 \right] \cdot \|\theta_\ell^j - \theta_{\ell, \beta}^*\|_{\Sigma_\ell^j}^2 \\ & \quad + \left(\frac{4\delta_{\text{rej}}}{\lambda\beta} + \frac{4}{\beta\lambda} \mathbb{P}_{\hat{\pi}^j} [M_{\text{rej}} < 4C_{\text{cond}}(\bar{\pi}_\ell^j | \mathbf{x}_\ell)] \right) \cdot \|\theta_{\ell, \beta}^* - \theta_\ell^j\|_{\Sigma_\ell^j}^2, \end{aligned}$$

where the last inequality follows by the fact that $\sigma_{\min}(\Sigma_\ell^j) \geq \lambda$, and $\|\varphi_\ell(x, a)\| \leq 2$, for all $\ell \in [H]$, $(x, a) \in \mathcal{X} \times \mathcal{A}$ (follows by [Assumption I.1](#)). And so, by [Lemma M.1](#) and [Lemma M.2](#), we have that for all $\ell \in [H]$:

$$\begin{aligned} & \mathbb{E}_{\bar{\pi}^j} \left[D_{\text{KL}} \left(\bar{\pi}_\ell^j(\cdot | \mathbf{x}_\ell) \parallel \bar{\pi}_{\ell, \beta}^{j, \star}(\cdot | \mathbf{x}_\ell) \right) \right] \\ & \leq (\lambda\nu^2 + 4\varepsilon_{\text{span}} + 4\delta_{\text{rej}} + 4\mathbb{P}_{\hat{\pi}^j} [M_{\text{rej}} < 4C_{\text{cond}}(\bar{\pi}_\ell^j | \mathbf{x}_\ell)]) \cdot \frac{\varepsilon_{\text{reg}}^2}{\beta\lambda}. \end{aligned} \quad (151)$$

We now bound the distribution shift terms $(\mathbb{P}_{\hat{\pi}^j} [M_{\text{rej}} < 4C_{\text{cond}}(\bar{\pi}_\ell^j | \cdot)])_\ell$ in [\(151\)](#) and on the right-hand side of [\(149\)](#).

Bounding the distribution shift terms. Let $\mathcal{X}_{\ell, \text{span}}^j$ and $(\varepsilon_{\text{span}}, \underline{\varepsilon}_{\text{span}})$ be as in [Lemma M.1](#) and [Lemma K.2](#), respectively. By [Lemma M.1](#), we have that for all $\ell \in [H]$ and $(x_0, a_0) \in \mathcal{X} \times \mathcal{A}$,

$$\mathbb{P}_{\hat{\pi}^j} [\mathbf{x}_\ell \in \mathcal{X}_{\ell, \text{span}}^j] = \mathbb{P}_{\hat{\pi}^j} [\mathbf{x}_\ell \in \mathcal{X}_{\ell, \text{span}}^j | \mathbf{x}_0 = x_0, \mathbf{a}_0 = a_0] \geq 1 - \varepsilon_{\text{span}}, \quad (152)$$

where $\underline{\varepsilon}_{\text{span}}$ is as in [Lemma K.2](#), and for all $x \in \mathcal{X}_{\ell, \text{span}}^j$:

$$\mathbb{P}_{\mathbf{a} \sim \pi_{\ell, \text{ref}}(\cdot | x)} \left[\|\varphi_\ell(x, \mathbf{a})\|_{(\Sigma_\ell^j)^{-1}}^2 > \nu^2 \right] \leq \underline{\varepsilon}_{\text{span}} \leq \frac{1}{4C_{\text{cond}}(\pi_{\ell, \beta}^*)}, \quad (153)$$

where the last inequality follows by the fact that $\bar{N}_{\text{span}} \geq 4C_{\text{cond}}(\pi_{\beta}^*)$ (see parameter choices in [Algorithm 4](#)). On the other hand, by [Lemma M.3](#), we have that for all $\ell \in [H]$ and $(x, a) \in \mathcal{X}_{\ell, \text{span}}^j \times \mathcal{A}$:

$$\frac{\bar{\pi}_\ell^j(a | x)}{\pi_{\ell, \text{ref}}(a | x)} \vee \frac{\bar{\pi}_{\ell, \beta}^{j, \star}(a | x)}{\pi_{\ell, \text{ref}}(a | x)} \leq 2e^{\frac{4\nu}{\beta} \|\theta_\ell^j - \theta_{\ell, \beta}^*\|_{\Sigma_\ell^j}} \cdot \frac{\pi_{\ell, \beta}^*(a | x)}{\pi_{\ell, \text{ref}}(a | x)},$$

$$\leq 2e \frac{\pi_{\ell,\beta}^*(a \mid x)}{\pi_{\ell,\text{ref}}(a \mid x)}, \quad (154)$$

where the last step follows by [Lemma M.2](#) and the fact that $\varepsilon_{\text{reg}} \leq \beta/(4\nu)$ (see choice of ν in [Algorithm 4](#)). Therefore, we have that for all $\ell \in [H]$, $x \in \mathcal{X}_{\ell,\text{span}}^j$, and $\pi \in \{\bar{\pi}_\ell^j, \bar{\pi}_{\ell,\beta}^{j,*}\}$:

$$C_{\text{cond}}(\pi \mid x) \leq 2e C_{\text{cond}}(\pi_{\ell,\beta}^*). \quad (155)$$

Now, by (154) and the fact that $M_{\text{rej}} \geq 8e C_{\text{cond}}(\pi_{\ell,\beta}^*)$ (see [Section J.1.4](#)), we have that for all $\ell \in [H]$: $x \in \mathcal{X}_{\ell,\text{span}}^j$ only if $M_{\text{rej}} \geq 4C_{\text{cond}}(\bar{\pi}_\ell^j \mid x)$ and so by (135), we have for all $\ell \in [H]$:

$$1 - \varepsilon_{\text{span}} \leq \mathbb{P}_{\hat{\pi}^j} [\mathbf{x}_\ell \in \mathcal{X}_{\ell,\text{span}}^j] \leq \mathbb{P}_{\hat{\pi}^j} [M_{\text{rej}} \geq 4C_{\text{cond}}(\bar{\pi}_\ell^j \mid \mathbf{x}_\ell)].$$

This implies that for all $\ell \in [H]$:

$$\mathbb{P}_{\hat{\pi}^j} [M_{\text{rej}} < 4C_{\text{cond}}(\bar{\pi}_\ell^j \mid \mathbf{x}_\ell)] \leq \varepsilon_{\text{span}}. \quad (156)$$

Putting it all together. Combining (156) with (151) and (149), we get that

$$\begin{aligned} & J_\beta(\pi_\beta^*) - J_\beta(\hat{\pi}_{1:H}^j) \\ & \leq \sum_{\ell=1}^H (\lambda\nu^2 + 4\varepsilon_{\text{span}} + 4\delta_{\text{rej}} + 4\mathbb{P}_{\hat{\pi}^j} [M_{\text{rej}} < 4C_{\text{cond}}(\bar{\pi}_\ell^j \mid \mathbf{x}_\ell)]) \cdot \frac{\varepsilon_{\text{reg}}^2}{\lambda} \\ & \quad + 4H \left(R_{\max} + 5HB/4 + \beta \log(4M_{\text{rej}} \log(4\delta_{\text{rej}}^{-1})) \right) \cdot \delta_{\text{rej}} \\ & \quad + B \left(\log(4M_{\text{rej}} \log(4\delta_{\text{rej}}^{-1})) + 5H \right) \sum_{\ell=1}^H \mathbb{P}_{\hat{\pi}^j} [M_{\text{rej}} < 4C_{\text{cond}}(\bar{\pi}_\ell^j \mid \mathbf{x}_\ell)], \\ & \leq H \cdot (\lambda\nu^2 + 4\varepsilon_{\text{span}} + 4\delta_{\text{rej}} + 4\varepsilon_{\text{span}}) \cdot \frac{\varepsilon_{\text{reg}}^2}{\lambda} \\ & \quad + 4H \left(R_{\max} + 5HB/4 + \beta \log(4M_{\text{rej}} \log(4\delta_{\text{rej}}^{-1})) \right) \cdot \delta_{\text{rej}} \\ & \quad + B \left(\log(4M_{\text{rej}} \log(4\delta_{\text{rej}}^{-1})) + 5H \right) H \varepsilon_{\text{span}}, \\ & \leq \varepsilon, \end{aligned} \quad (157)$$

where the last inequality follows by the choice of parameters in [Algorithm 4](#). Combining this with the fact that $\mathbb{P}[\mathcal{E}^{\text{span}} \cap \mathcal{E}^{\text{reg}}] \geq 1 - \delta$ (by [Lemma L.4](#) and [Lemma M.1](#) and the union bound) completes the proof. \square

Appendix N. Technical Lemmas for Multi-Turn Exploration

In this section, we present and prove the technical results required for the proofs in the Multi-turn setting. Some of the statements provided here are generalizations of those in [Appendix G](#), originally formulated for the contextual bandit setting.

Lemma N.1 (KL bound for truncated softmax policies). *Let $h \in [H]$, $B, \nu > 0$, $(x, \mathbf{a}) \in \mathcal{X} \times \mathcal{A}$, $\theta_h \in \mathbb{B}(B)$, and $\Sigma_h \in \mathbb{R}^{d \times d}$ be given, and let $\varphi_h(x, \cdot) := \phi_h(x, \cdot) - \phi_h(x, \mathbf{a})$. Further, define $\bar{\pi}_{h,\theta}(\cdot | x) \propto \pi_{h,\text{ref}}(\cdot | x) \cdot e^{\beta^{-1} \bar{\varphi}_h(x, \cdot)^\top \theta}$ and $\bar{\pi}_{h,\beta}^*(\cdot | x) \propto \pi_{h,\text{ref}}(\cdot | x) \cdot e^{\beta^{-1} \bar{\varphi}_h(x, \cdot)^\top \theta_{h,\beta}^*}$, where $\bar{\varphi}_h(x, \cdot) := \varphi_h(x, \cdot) \cdot \mathbb{I} \left\{ \|\varphi_h(x, \cdot)\|_{\Sigma_h^{-1}}^2 \leq \nu^2 \right\}$. If $\|\theta_{h,\beta}^* - \theta_h\|_{\Sigma_h} \leq \beta/\nu$, then we have that*

$$D_{\text{KL}}(\bar{\pi}_{h,\theta}(\cdot | x) \| \bar{\pi}_{h,\beta}^*(\cdot | x)) \leq \beta^{-1} \mathbb{E}_{\mathbf{a} \sim \bar{\pi}_{h,\theta}(\cdot | x)} \left[\|\bar{\varphi}_h(x, \mathbf{a})\|_{\Sigma_h^{-1}}^2 \right] \cdot \|\theta_{h,\beta}^* - \theta_h\|_{\Sigma_h}^2.$$

Proof of Lemma N.1. Let $h \in [H]$, $B, \nu > 0$, $(x, \mathbf{a}) \in \mathcal{X} \times \mathcal{A}$, $\theta_h \in \mathbb{B}(B)$, and $\Sigma_h \in \mathbb{R}^{d \times d}$ be as in the lemma statement. We have for $\bar{Z}_{h,\theta}(x) := \mathbb{E}_{\mathbf{a} \sim \pi_{h,\text{ref}}(\cdot | x)} [\exp(\beta^{-1} \bar{\varphi}_h(x, \mathbf{a})^\top \theta)]$:

$$\begin{aligned} D_{\text{KL}}(\bar{\pi}_{h,\theta}(\cdot | x) \| \bar{\pi}_{h,\beta}^*(\cdot | x)) &= \beta \log \frac{\bar{Z}_{h,\theta_{h,\beta}^*}(x)}{\bar{Z}_{h,\theta_h}(x)} + \mathbb{E}_{\mathbf{a} \sim \bar{\pi}_{h,\theta}(\cdot | x)} \left[\bar{\varphi}_h(x, \mathbf{a})^\top (\theta_h - \theta_{h,\beta}^*) \right], \\ &= \beta \log \left(\mathbb{E}_{\mathbf{a} \sim \bar{\pi}_{h,\theta}(\cdot | x)} \left[\exp(\beta^{-1} \bar{\varphi}_h(x, \mathbf{a})^\top (\theta_{h,\beta}^* - \theta_h)) \right] \right) + \mathbb{E}_{\mathbf{a} \sim \bar{\pi}_{h,\theta}(\cdot | x)} \left[\bar{\varphi}_h(x, \mathbf{a})^\top (\theta_h - \theta_{h,\beta}^*) \right]. \end{aligned} \quad (158)$$

Now, by Hölder's inequality, we have that

$$\begin{aligned} |\langle \bar{\varphi}_h(x, \mathbf{a}), \theta_{h,\beta}^* - \theta_h \rangle| &\leq \|\bar{\varphi}_h(x, \mathbf{a})\|_{\Sigma_h^{-1}} \cdot \|\theta_{h,\beta}^* - \theta_h\|_{\Sigma_h}, \\ &\leq \nu \cdot \|\theta_{h,\beta}^* - \theta_h\|_{\Sigma_h}, \\ &\leq \beta, \end{aligned} \quad (159)$$

where the last inequality follows by the assumption made on $\|\theta_{h,\beta}^* - \theta_h\|_{\Sigma_h}$. Combining (159) with the fact that $e^x \leq 1 + x + x^2$, for all $x \leq 1$, we get that

$$\begin{aligned} &\beta \log \left(\mathbb{E}_{\mathbf{a} \sim \bar{\pi}_{h,\theta}(\cdot | x)} \left[\exp(\beta^{-1} \bar{\varphi}_h(x, \mathbf{a})^\top (\theta_{h,\beta}^* - \theta_h)) \right] \right) \\ &\leq \mathbb{E}_{\mathbf{a} \sim \bar{\pi}_{h,\theta}(\cdot | x)} [\langle \bar{\varphi}_h(x, \mathbf{a}), \theta_{h,\beta}^* - \theta_h \rangle] + \beta^{-1} \mathbb{E}_{\mathbf{a} \sim \bar{\pi}_{h,\theta}(\cdot | x)} [\langle \bar{\varphi}_h(x, \mathbf{a}), \theta_{h,\beta}^* - \theta_h \rangle^2], \\ &\leq \mathbb{E}_{\mathbf{a} \sim \bar{\pi}_{h,\theta}(\cdot | x)} [\langle \bar{\varphi}_h(x, \mathbf{a}), \theta_{h,\beta}^* - \theta_h \rangle] + \beta^{-1} \mathbb{E}_{\mathbf{a} \sim \bar{\pi}_{h,\theta}(\cdot | x)} \left[\|\bar{\varphi}_h(x, \mathbf{a})\|_{\Sigma_h^{-1}}^2 \right] \cdot \|\theta_{h,\beta}^* - \theta_h\|_{\Sigma_h}^2. \end{aligned}$$

Plugging this in to [Eq. \(158\)](#) gives the desired result. \square

Lemma N.2 (Density ratio bound). *Suppose [Assumption 1.1](#) holds and let $h \in [H]$, $B, \nu > 0$, $(x, \mathbf{a}) \in \mathcal{X} \times \mathcal{A}$, $\theta_h \in \mathbb{B}(B)$, and $\Sigma_h \in \mathbb{R}^{d \times d}$ be given and let $\varphi_h(x, \cdot) := \phi_h(x, \cdot) - \phi_h(x, \mathbf{a})$. Further, define $\bar{\pi}_{h,\theta}(\cdot | x) \propto \pi_{h,\text{ref}}(\cdot | x) \cdot e^{\bar{\varphi}_h(x, \cdot)^\top \theta/\beta}$, where $\bar{\varphi}_h(x, \cdot) := \varphi_h(x, \cdot) \cdot \mathbb{I} \left\{ \|\varphi_h(x, \cdot)\|_{\Sigma_h^{-1}}^2 \leq \nu^2 \right\}$. Then,*

$$\forall a \in \mathcal{A}, \quad \frac{\bar{\pi}_{h,\theta}(a | x)}{\pi_{h,\text{ref}}(a | x)} \leq \min \left(\frac{\pi_{h,\beta}^*(a | x)}{\pi_{h,\text{ref}}(a | x)} \cdot \frac{e^{\frac{4\nu}{\beta} \|\theta_h - \theta_{h,\beta}^*\|_{\Sigma_h}}}{1 - C_{\text{cond}}(\pi_{h,\beta}^* | x) \cdot p_{\text{surprise}}(x)}, e^{2B/\beta} \right),$$

where $\pi_{h,\beta}^*$ and $\theta_{h,\beta}^*$ are as in [Definition I.2](#) and [Assumption I.1](#), respectively, and

$$p_{\text{surprise}}(x) := \mathbb{P}_{\mathbf{a} \sim \pi_{h,\text{ref}}(\cdot|x)} \left[\|\bar{\varphi}_h(x, \mathbf{a})\|_{\Sigma_h^{-1}}^2 > \nu^2 \right].$$

Proof of Lemma N.2. Let $h \in [H]$, $B, \nu > 0$, $(x, \mathbf{a}) \in \mathcal{X} \times \mathcal{A}$, $\theta_h \in \mathbb{B}(B)$, and $\Sigma_h \in \mathbb{R}^{d \times d}$ be as in the lemma statement, and fix $a \in \mathcal{A}$. We have

$$\frac{\bar{\pi}_{h,\theta}(a | x)}{\pi_{h,\text{ref}}(a | x)} = \frac{\exp(\beta^{-1} \bar{\varphi}_h(x, a)^\top \theta_h)}{Z_{\theta_h}},$$

where $Z_\theta := \mathbb{E}_{\mathbf{a} \sim \pi_{h,\text{ref}}(\cdot|x)} [\exp(\beta^{-1} \bar{\varphi}_h(x, \mathbf{a})^\top \theta)]$, for $\theta \in \mathbb{R}^d$. Therefore, we have that

$$\frac{\bar{\pi}_{h,\theta}(a | x)}{\pi_{h,\text{ref}}(a | x)} = \frac{1}{\mathbb{E}_{\mathbf{a} \sim \pi_{h,\text{ref}}(\cdot|x)} [\exp(\beta^{-1} \langle \bar{\varphi}_h(x, \mathbf{a}) - \bar{\varphi}_h(x, a), \theta_h \rangle)]}. \quad (160)$$

On the other hand, we have

$$\begin{aligned} & \mathbb{E}_{\mathbf{a} \sim \pi_{h,\text{ref}}(\cdot|x)} [\exp(\beta^{-1} \langle \bar{\varphi}_h(x, \mathbf{a}) - \bar{\varphi}_h(x, a), \theta_h \rangle)] \\ & \geq \mathbb{E}_{\mathbf{a} \sim \pi_{h,\text{ref}}(\cdot|x)} \left[\exp(\beta^{-1} \langle \bar{\varphi}_h(x, \mathbf{a}) - \bar{\varphi}_h(x, a), \theta_h \rangle) \cdot \mathbb{I} \left\{ \|\bar{\varphi}_h(x, \mathbf{a})\|_{\Sigma_h^{-1}}^2 \leq \nu^2 \right\} \right], \end{aligned}$$

and so by Hölder's inequality, we have

$$\begin{aligned} & \geq \mathbb{E}_{\mathbf{a} \sim \pi_{h,\text{ref}}(\cdot|x)} \left[e^{-\frac{2}{\beta} \|\theta_h - \theta_{h,\beta}^*\|_{\Sigma_h} \cdot \|\bar{\varphi}_h(x, \mathbf{a}) - \bar{\varphi}_h(x, a)\|_{\Sigma_h^{-1}}} e^{\beta^{-1} \langle \bar{\varphi}_h(x, \mathbf{a}) - \bar{\varphi}_h(x, a), \theta_{h,\beta}^* \rangle} \cdot \mathbb{I} \left\{ \|\bar{\varphi}_h(x, \mathbf{a})\|_{\Sigma_h^{-1}}^2 \leq \nu^2 \right\} \right], \\ & \geq \mathbb{E}_{\mathbf{a} \sim \pi_{h,\text{ref}}(\cdot|x)} \left[e^{-\frac{2}{\beta} \|\theta_h - \theta_{h,\beta}^*\|_{\Sigma_h} \cdot \left(\|\bar{\varphi}_h(x, \mathbf{a})\|_{\Sigma_h^{-1}} + \|\bar{\varphi}_h(x, a)\|_{\Sigma_h^{-1}} \right)} e^{\beta^{-1} \langle \bar{\varphi}_h(x, \mathbf{a}) - \bar{\varphi}_h(x, a), \theta_{h,\beta}^* \rangle} \cdot \mathbb{I} \left\{ \|\bar{\varphi}_h(x, \mathbf{a})\|_{\Sigma_h^{-1}}^2 \leq \nu^2 \right\} \right], \\ & \geq \frac{e^{-\frac{4\nu}{\beta} \cdot \|\theta_h - \theta_{h,\beta}^*\|_{\Sigma_h}}}{e^{\beta^{-1} \bar{\varphi}_h(x, a)^\top \theta_{h,\beta}^*}} \left(\mathbb{E}_{\mathbf{a} \sim \pi_{h,\text{ref}}(\cdot|x)} \left[e^{\beta^{-1} \langle \bar{\varphi}_h(x, \mathbf{a}) - \bar{\varphi}_h(x, a), \theta_{h,\beta}^* \rangle} \cdot \mathbb{I} \left\{ \|\bar{\varphi}_h(x, \mathbf{a})\|_{\Sigma_h^{-1}}^2 \leq \nu^2 \right\} \right] \right), \\ & = \frac{e^{-\frac{4\nu}{\beta} \cdot \|\theta_h - \theta_{h,\beta}^*\|_{\Sigma_h}}}{e^{\beta^{-1} \bar{\varphi}_h(x, a)^\top \theta_{h,\beta}^*}} \left(Z_{\theta_{h,\beta}^*} - \mathbb{E}_{\mathbf{a} \sim \pi_{h,\text{ref}}(\cdot|x)} \left[e^{\beta^{-1} \langle \bar{\varphi}_h(x, \mathbf{a}) - \bar{\varphi}_h(x, a), \theta_{h,\beta}^* \rangle} \cdot \mathbb{I} \left\{ \|\bar{\varphi}_h(x, \mathbf{a})\|_{\Sigma_h^{-1}}^2 > \nu^2 \right\} \right] \right), \\ & = \frac{Z_{\theta_{h,\beta}^*} \cdot e^{-\frac{4\nu}{\beta} \cdot \|\theta_h - \theta_{h,\beta}^*\|_{\Sigma_h}}}{e^{\beta^{-1} \bar{\varphi}_h(x, a)^\top \theta_{h,\beta}^*}} \left(1 - \mathbb{E}_{\mathbf{a} \sim \pi_{h,\text{ref}}(\cdot|x)} \left[\frac{e^{\beta^{-1} \langle \bar{\varphi}_h(x, \mathbf{a}) - \bar{\varphi}_h(x, a), \theta_{h,\beta}^* \rangle}}{Z_{\theta_{h,\beta}^*}} \cdot \mathbb{I} \left\{ \|\bar{\varphi}_h(x, \mathbf{a})\|_{\Sigma_h^{-1}}^2 > \nu^2 \right\} \right] \right), \\ & = \frac{Z_{\theta_{h,\beta}^*} \cdot e^{-\frac{4\nu}{\beta} \cdot \|\theta_h - \theta_{h,\beta}^*\|_{\Sigma_h}}}{e^{\beta^{-1} \bar{\varphi}_h(x, a)^\top \theta_{h,\beta}^*}} \left(1 - \mathbb{P}_{\mathbf{a} \sim \pi_{h,\text{ref}}(\cdot|x)} \left[\|\bar{\varphi}_h(x, \mathbf{a})\|_{\Sigma_h^{-1}}^2 > \nu^2 \right] \right), \end{aligned}$$

and we can further lower bound by

$$\begin{aligned} & \geq \frac{Z_{\theta_{h,\beta}^*} \cdot e^{-\frac{4\nu}{\beta} \cdot \|\theta_h - \theta_{h,\beta}^*\|_{\Sigma_h}}}{e^{\beta^{-1} \bar{\varphi}_h(x, a)^\top \theta_{h,\beta}^*}} \left(1 - C_{\text{cond}}(\pi_{h,\beta}^* | x) \cdot \mathbb{P}_{\mathbf{a} \sim \pi_{h,\text{ref}}(\cdot|x)} \left[\|\bar{\varphi}_h(x, \mathbf{a})\|_{\Sigma_h^{-1}}^2 > \nu^2 \right] \right), \\ & = \frac{\pi_{h,\text{ref}}(a | x) \cdot e^{-\frac{4\nu}{\beta} \cdot \|\theta_h - \theta_{h,\beta}^*\|_{\Sigma_h}}}{\pi_{h,\beta}^*(a | x)} \left(1 - C_{\text{cond}}(\pi_{h,\beta}^* | x) \cdot \mathbb{P}_{\mathbf{a} \sim \pi_{h,\text{ref}}(\cdot|x)} \left[\|\bar{\varphi}_h(x, \mathbf{a})\|_{\Sigma_h^{-1}}^2 > \nu^2 \right] \right), \end{aligned}$$

where the last equality follows by [Assumption I.1](#) and [Lemma N.4](#). Using this with (160) and the fact that $\bar{\pi}_{h,\theta}(a | x) / \pi_{h,\text{ref}}(a | x)$ is at most $e^{2B/\beta}$, we get the desired result. \square

Lemma N.3 (Two-sided bound for truncated policies). *Suppose [Assumption I.1](#) holds and let $h \in [H]$, $\nu > 0$, $(x, \mathbf{a}) \in \mathcal{X} \times \mathcal{A}$, $\theta_h \in \mathbb{R}^d$, and $\Sigma_h \in \mathbb{R}^{d \times d}$ be given and let $\varphi_h(\cdot, \cdot) := \phi_h(\cdot, \cdot) - \phi_h(\cdot, \mathbf{a})$. Let $\pi_{h,\beta}^*$ and $\theta_{h,\beta}^*$ are as in [Definition I.2](#) and [Assumption I.1](#), respectively. Further, define*

$$\bar{\pi}_{h,\theta}(\cdot | x) \propto \pi_{h,\text{ref}}(\cdot | x) \cdot e^{\bar{\varphi}_h(x, \cdot)^\top \theta / \beta}, \quad \text{and} \quad \bar{\pi}_{h,\beta}^*(\cdot | x) \propto \pi_{h,\text{ref}}(\cdot | x) \cdot e^{\bar{\varphi}_h(x, \cdot)^\top \theta_{h,\beta}^* / \beta}, \quad (161)$$

where $\bar{\varphi}_h(x, \cdot) := \varphi_h(x, \cdot) \cdot \mathbb{I} \left\{ \|\varphi_h(x, \cdot)\|_{\Sigma_h^{-1}}^2 \leq \nu^2 \right\}$. Then, we have

$$\begin{aligned} & \left| \sum_{a \in \mathcal{A}} \pi_{h,\beta}^*(a | x) \cdot \left(Q_{h,\beta}^*(x, a) - \beta \cdot \log \frac{\pi_{h,\beta}^*(a | x)}{\pi_{h,\text{ref}}(a | x)} \right) - \sum_{a \in \mathcal{A}} \bar{\pi}_{h,\theta}(a | x) \cdot \left(Q_{h,\beta}^*(x, a) - \beta \cdot \log \frac{\bar{\pi}_{h,\theta}(a | x)}{\pi_{h,\text{ref}}(a | x)} \right) \right| \\ & \leq \beta D_{\text{KL}}(\bar{\pi}_{h,\theta}(\cdot | x) \| \pi_{h,\beta}^*(\cdot | x)) \\ & \quad + 2R_{\max} \max_{\pi \in \{\bar{\pi}_{h,\theta}, \bar{\pi}_{h,\beta}^*, \pi_{h,\beta}^*\}} \min \left(1, C_{\text{cond}}(\pi | x) \cdot \mathbb{P}_{\mathbf{a} \sim \pi_{h,\text{ref}}(\cdot | x)} \left[\|\varphi_h(x, \mathbf{a})\|_{\Sigma_h^{-1}}^2 > \nu^2 \right] \right). \end{aligned}$$

Proof of Lemma N.3. Let $h \in [H]$, $\nu > 0$, $(x, \mathbf{a}) \in \mathcal{X} \times \mathcal{A}$, $\theta_h \in \mathbb{R}^d$, and $\Sigma_h \in \mathbb{R}^{d \times d}$ be as in the lemma statement. We define $\tilde{Q}_{h,\beta}^*(\cdot, \cdot) := Q_{h,\beta}^*(\cdot, \cdot) - Q_{h,\beta}^*(\cdot, \mathbf{a})$. Note that by [Assumption I.1](#), we have that for all $(x', a') \in \mathcal{X} \times \mathcal{A}$:

$$\tilde{Q}_{h,\beta}^*(x', a') = \varphi_h(x', a')^\top \theta_{h,\beta}^*,$$

where φ_h is as in the lemma statement. Now, observe that by the definition of $\bar{\varphi}_h$, we have for any $\pi \in \Pi$,

$$\begin{aligned} & \left| \mathbb{E}_\pi \left[\varphi_h(\mathbf{x}_h, \mathbf{a}_h)^\top \theta_{h,\beta}^* \mid \mathbf{x}_h = x \right] - \mathbb{E}_\pi \left[\bar{\varphi}_h(\mathbf{x}_h, \mathbf{a}_h)^\top \theta_{h,\beta}^* \mid \mathbf{x}_h = x \right] \right| \\ & \leq R_{\max} \cdot \mathbb{P}_\pi \left[\|\varphi_h(\mathbf{x}_h, \mathbf{a}_h)\|_{\Sigma_h^{-1}}^2 > \nu^2 \mid \mathbf{x}_h = x \right]. \end{aligned} \quad (162)$$

Instantiating this with $\pi = \bar{\pi}_{h,\theta}$ and using [Assumption I.1](#), we get that

$$\begin{aligned} & \left| \sum_{a \in \mathcal{A}} \bar{\pi}_{h,\theta}(a | x) \cdot \tilde{Q}_{h,\beta}^*(x, a) - \mathbb{E}_{\bar{\pi}_{h,\theta}} \left[\bar{\varphi}_h(\mathbf{x}_h, \mathbf{a}_h)^\top \theta_{h,\beta}^* \mid \mathbf{x}_h = x \right] \right| \\ & \leq R_{\max} \cdot \mathbb{P}_{\bar{\pi}_{h,\theta}} \left[\|\varphi_h(\mathbf{x}_h, \mathbf{a}_h)\|_{\Sigma_h^{-1}}^2 > \nu^2 \mid \mathbf{x}_h = x \right], \\ & \leq R_{\max} \cdot \min \left(1, C_{\text{cond}}(\bar{\pi}_{h,\theta} | x) \cdot \mathbb{P}_{\pi_{\text{ref}}} \left[\|\varphi_h(\mathbf{x}_h, \mathbf{a}_h)\|_{\Sigma_h^{-1}}^2 > \nu^2 \mid \mathbf{x}_h = x \right] \right). \end{aligned} \quad (163)$$

Now, instantiating (162) with $\pi = \pi_{h,\beta}^*$ and using [Assumption I.1](#), we get that

$$\begin{aligned} & \sum_{a \in \mathcal{A}} \pi_{h,\beta}^*(a | x) \cdot \left(\tilde{Q}_{h,\beta}^*(x, a) - \beta \log \frac{\pi_{h,\beta}^*(a | x)}{\pi_{h,\text{ref}}(a | x)} \right) \\ & \leq \mathbb{E}_{\pi_{h,\beta}^*} \left[\bar{\varphi}_h(\mathbf{x}_h, \mathbf{a}_h)^\top \theta_{h,\beta}^* \mid \mathbf{x}_h = x \right] - \beta D_{\text{KL}}(\pi_{h,\beta}^*(\cdot | x) \| \pi_{h,\text{ref}}(\cdot | x)) \\ & \quad + R_{\max} \cdot \mathbb{P}_{\pi_{h,\beta}^*} \left[\|\varphi_h(\mathbf{x}_h, \mathbf{a}_h)\|_{\Sigma_h^{-1}}^2 > \nu^2 \mid \mathbf{x}_h = x \right], \end{aligned}$$

$$\begin{aligned}
&\leq \mathbb{E}_{\pi_{h,\beta}^*} \left[\bar{\varphi}_h(\mathbf{x}_h, \mathbf{a}_h)^\top \theta_{h,\beta}^* \mid \mathbf{x}_h = x \right] - \beta D_{\text{KL}}(\pi_{h,\beta}^*(\cdot \mid x) \parallel \pi_{h,\text{ref}}(\cdot \mid x)) \\
&\quad + R_{\max} \cdot \min \left(1, C_{\text{cond}}(\pi_{h,\beta}^* \mid x) \cdot \mathbb{P}_{\pi_{\text{ref}}} \left[\|\varphi_h(\mathbf{x}_h, \mathbf{a}_h)\|_{\Sigma_h^{-1}}^2 > \nu^2 \mid \mathbf{x}_h = x \right] \right), \\
&\leq \mathbb{E}_{\bar{\pi}_{h,\beta}^*} \left[\bar{\varphi}_h(\mathbf{x}_h, \mathbf{a}_h)^\top \theta_{h,\beta}^* \mid \mathbf{x}_h = x \right] - \beta D_{\text{KL}}(\bar{\pi}_{h,\beta}^*(\cdot \mid x) \parallel \pi_{h,\text{ref}}(\cdot \mid x)) \\
&\quad + R_{\max} \cdot \min \left(1, C_{\text{cond}}(\pi_{h,\beta}^* \mid x) \cdot \mathbb{P}_{\pi_{\text{ref}}} \left[\|\varphi_h(\mathbf{x}_h, \mathbf{a}_h)\|_{\Sigma_h^{-1}}^2 > \nu^2 \mid \mathbf{x}_h = x \right] \right), \quad (164)
\end{aligned}$$

where in the last step we used the fact that

$$\bar{\pi}_{h,\beta}^*(\cdot \mid x) \in \arg \max_{\pi \in \Delta(\mathcal{A})} \left\{ \sum_{a \in \mathcal{A}} \pi(a) \cdot \bar{\varphi}_h(x, a)^\top \theta_{h,\beta}^* - \beta D_{\text{KL}}(\pi(\cdot \mid x) \parallel \pi_{h,\text{ref}}(\cdot \mid x)) \right\}, \quad (165)$$

which follows from the definition of $\bar{\pi}_{h,\beta}^*$. Using (162) with Assumption I.1 again, we have that for all $\pi \in \Pi$:

$$\begin{aligned}
&\sum_{a \in \mathcal{A}} \pi(a \mid x) \cdot \left(\tilde{Q}_{h,\beta}^*(x, a) - \beta \log \frac{\pi(a \mid x)}{\pi_{h,\text{ref}}(a \mid x)} \right) \\
&\geq \mathbb{E}_{\pi} \left[\bar{\varphi}_h(\mathbf{x}_h, \mathbf{a}_h)^\top \theta_{h,\beta}^* \mid \mathbf{x}_h = x \right] - \beta D_{\text{KL}}(\pi(\cdot \mid x) \parallel \pi_{h,\text{ref}}(\cdot \mid x)) \\
&\quad - R_{\max} \cdot \mathbb{P}_{\pi} \left[\|\varphi_h(\mathbf{x}_h, \mathbf{a}_h)\|_{\Sigma_h^{-1}}^2 > \nu^2 \mid \mathbf{x}_h = x \right].
\end{aligned}$$

Thus, taking the maximum over π on both sides and using the definition of $\pi_{h,\beta}^*$ we get

$$\begin{aligned}
&\sum_{a \in \mathcal{A}} \pi_{h,\beta}^*(a \mid x) \cdot \left(\tilde{Q}_{h,\beta}^*(x, a) - \beta \log \frac{\pi_{h,\beta}^*(a \mid x)}{\pi_{h,\text{ref}}(a \mid x)} \right) \\
&\geq \max_{\pi} \left\{ \mathbb{E}_{\pi} \left[\bar{\varphi}_h(\mathbf{x}_h, \mathbf{a}_h)^\top \theta_{h,\beta}^* \mid \mathbf{x}_h = x \right] - \beta D_{\text{KL}}(\pi(\cdot \mid x) \parallel \pi_{h,\text{ref}}(\cdot \mid x)) \right. \\
&\quad \left. - R_{\max} \cdot \mathbb{P}_{\pi} \left[\|\varphi_h(\mathbf{x}_h, \mathbf{a}_h)\|_{\Sigma_h^{-1}}^2 > \nu^2 \mid \mathbf{x}_h = x \right] \right\}, \\
&\geq \mathbb{E}_{\bar{\pi}_{h,\beta}^*} \left[\bar{\varphi}_h(\mathbf{x}_h, \mathbf{a}_h)^\top \theta_{h,\beta}^* \mid \mathbf{x}_h = x \right] - \beta D_{\text{KL}}(\bar{\pi}_{h,\beta}^*(\cdot \mid x) \parallel \pi_{h,\text{ref}}(\cdot \mid x)) \\
&\quad - R_{\max} \cdot \mathbb{P}_{\bar{\pi}_{h,\beta}^*} \left[\|\varphi_h(\mathbf{x}_h, \mathbf{a}_h)\|_{\Sigma_h^{-1}}^2 > \nu^2 \mid \mathbf{x}_h = x \right], \\
&\geq \mathbb{E}_{\bar{\pi}_{h,\beta}^*} \left[\bar{\varphi}_h(\mathbf{x}_h, \mathbf{a}_h)^\top \theta_{h,\beta}^* \mid \mathbf{x}_h = x \right] - \beta D_{\text{KL}}(\bar{\pi}_{h,\beta}^*(\cdot \mid x) \parallel \pi_{h,\text{ref}}(\cdot \mid x)) \\
&\quad - R_{\max} \cdot \min \left(1, C_{\text{cond}}(\bar{\pi}_{h,\beta}^* \mid x) \cdot \mathbb{P}_{\pi_{\text{ref}}} \left[\|\varphi_h(\mathbf{x}_h, \mathbf{a}_h)\|_{\Sigma_h^{-1}}^2 > \nu^2 \mid \mathbf{x}_h = x \right] \right). \quad (166)
\end{aligned}$$

By combining (163), (164), and (166), we get that

$$\begin{aligned}
&\left| \sum_{a \in \mathcal{A}} \pi_{h,\beta}^*(a \mid x) \cdot \left(Q_{h,\beta}^*(x, a) - \beta \cdot \log \frac{\pi_{h,\beta}^*(a \mid x)}{\pi_{h,\text{ref}}(a \mid x)} \right) \right. \\
&\quad \left. - \sum_{a \in \mathcal{A}} \bar{\pi}_{h,\theta}(a \mid x) \cdot \left(Q_{h,\beta}^*(x, a) - \beta \cdot \log \frac{\bar{\pi}_{h,\theta}(a \mid x)}{\pi_{h,\text{ref}}(a \mid x)} \right) \right| \\
&= \left| \sum_{a \in \mathcal{A}} \pi_{h,\beta}^*(a \mid x) \cdot \left(\tilde{Q}_{h,\beta}^*(x, a) - \beta \cdot \log \frac{\pi_{h,\beta}^*(a \mid x)}{\pi_{h,\text{ref}}(a \mid x)} \right) \right. \\
&\quad \left. - \sum_{a \in \mathcal{A}} \bar{\pi}_{h,\theta}(a \mid x) \cdot \left(\tilde{Q}_{h,\beta}^*(x, a) - \beta \cdot \log \frac{\bar{\pi}_{h,\theta}(a \mid x)}{\pi_{h,\text{ref}}(a \mid x)} \right) \right|
\end{aligned}$$

$$\begin{aligned}
 &\leq \left| \mathbb{E}_{\bar{\pi}_{h,\beta}^*} \left[\bar{\varphi}_h(\mathbf{x}_h, \mathbf{a}_h)^\top \theta_{h,\beta}^* \mid \mathbf{x}_h = x \right] - \beta D_{\text{KL}}(\bar{\pi}_{h,\beta}^*(\cdot \mid x) \parallel \pi_{h,\text{ref}}(\cdot \mid x)) \right. \\
 &\quad \left. - \mathbb{E}_{\bar{\pi}_{h,\theta}} \left[\bar{\varphi}_h(\mathbf{x}_t, \mathbf{a}_h)^\top \theta_{h,\beta}^* \mid \mathbf{x}_h = x \right] + \beta D_{\text{KL}}(\bar{\pi}_{h,\theta}(\cdot \mid x) \parallel \pi_{h,\text{ref}}(\cdot \mid x)) \right| \\
 &\quad + 2 \max_{\pi \in \{\bar{\pi}_{h,\theta}, \bar{\pi}_{h,\beta}^*, \pi_{h,\beta}^*\}} R_{\max} \min \left(1, C_{\text{cond}}(\pi \mid x) \cdot \mathbb{P}_{\pi_{\text{ref}}} \left[\|\varphi_h(\mathbf{x}_h, \mathbf{a}_h)\|_{\Sigma_h^{-1}}^2 > \nu^2 \mid \mathbf{x}_h = x \right] \right), \\
 &= \mathbb{E}_{\bar{\pi}_{h,\beta}^*} \left[\bar{\varphi}_h(\mathbf{x}_h, \mathbf{a}_h)^\top \theta_{h,\beta}^* \mid \mathbf{x}_h = x \right] - \beta D_{\text{KL}}(\bar{\pi}_{h,\beta}^*(\cdot \mid x) \parallel \pi_{h,\text{ref}}(\cdot \mid x)) \\
 &\quad - \mathbb{E}_{\bar{\pi}_{h,\theta}} \left[\bar{\varphi}_h(\mathbf{x}_t, \mathbf{a}_h)^\top \theta_{h,\beta}^* \mid \mathbf{x}_h = x \right] + \beta D_{\text{KL}}(\bar{\pi}_{h,\theta}(\cdot \mid x) \parallel \pi_{h,\text{ref}}(\cdot \mid x)) \\
 &\quad + 2 \max_{\pi \in \{\bar{\pi}_{h,\theta}, \bar{\pi}_{h,\beta}^*, \pi_{h,\beta}^*\}} R_{\max} \min \left(1, C_{\text{cond}}(\pi \mid x) \cdot \mathbb{P}_{\pi_{\text{ref}}} \left[\|\varphi_h(\mathbf{x}_h, \mathbf{a}_h)\|_{\Sigma_h^{-1}}^2 > \nu^2 \mid \mathbf{x}_h = x \right] \right), \\
 &= \beta D_{\text{KL}}(\bar{\pi}_{h,\theta}(\cdot \mid x) \parallel \bar{\pi}_{h,\beta}^*(\cdot \mid x)) \\
 &\quad + 2 \max_{\pi \in \{\bar{\pi}_{h,\theta}, \bar{\pi}_{h,\beta}^*, \pi_{h,\beta}^*\}} R_{\max} \min \left(1, C_{\text{cond}}(\pi \mid x) \cdot \mathbb{P}_{\pi_{\text{ref}}} \left[\|\varphi_h(\mathbf{x}_h, \mathbf{a}_h)\|_{\Sigma_h^{-1}}^2 > \nu^2 \mid \mathbf{x}_h = x \right] \right),
 \end{aligned}$$

where the second-to-last step follows by (165), and the last step is a standard manipulation of KL-divergence. \square

Lemma N.4. *The state-action value functions $(Q_{h,\beta}^*)$ and policies $(\pi_{h,\beta}^*)$ in Definition 1.2 satisfy: for all $h \in [H]$ and $(x, a) \in \mathcal{X} \times \mathcal{A}$,*

$$\sup_{\pi_{h+1:H}: \mathcal{X} \rightarrow \Delta(\mathcal{A})} Q_{h,\beta}^\pi(x, a) = Q_{h,\beta}^{\pi_\beta^*}(x, a) = Q_{h,\beta}^*(x, a);$$

and

$$\pi_{h,\beta}^*(\cdot \mid x) \in \arg \max_{\pi \in \Delta(\mathcal{A})} \sum_{a' \in \mathcal{A}} \pi(a') \cdot \left(Q_{h,\beta}^*(x, a') - \beta \log \frac{\pi(a')}{\pi_{h,\text{ref}}(a' \mid x)} \right), \quad (167)$$

where Q_h^π is as in Definition 1.1.

Proof of Lemma N.4. We prove the result via backward induction over $\ell = H + 1, \dots, 1$ that and $(x, a) \in \mathcal{X} \times \mathcal{A}$,

$$\sup_{\pi_{\ell+1:H}: \mathcal{X} \rightarrow \Delta(\mathcal{A})} Q_{\ell,\beta}^\pi(x, a) = Q_{\ell,\beta}^{\pi_\beta^*}(x, a) = Q_{\ell,\beta}^*(x, a); \quad (168)$$

and

$$\pi_{\ell,\beta}^*(\cdot \mid x) \in \arg \max_{\pi \in \Delta(\mathcal{A})} \sum_{a' \in \mathcal{A}} \pi(a') \cdot \left(Q_{\ell,\beta}^*(x, a') - \beta \log \frac{\pi(a')}{\pi_{\ell,\text{ref}}(a' \mid x)} \right). \quad (169)$$

The base case $\ell = H + 1$ is immediate. Now, suppose that (168) and (169) hold for $\ell = h \in [2 \dots H + 1]$. We show that they hold for $\ell = h - 1$. Fix $(x, a) \in \mathcal{X} \times \mathcal{A}$. For any $\pi : \mathcal{X} \rightarrow \Delta(\mathcal{A})$, we have

$$\begin{aligned}
 &Q_{h-1,\beta}^\pi(x, a) \\
 &= r_{h-1}^*(x, a) + \mathbb{E}_\pi \left[\sum_{\ell=h}^H r_\ell^*(\mathbf{x}_\ell, \mathbf{a}_\ell) - \beta \sum_{\ell=h}^H \log \frac{\pi_\ell(\mathbf{a}_\ell \mid \mathbf{x}_\ell)}{\pi_{\ell,\text{ref}}(\mathbf{a}_\ell \mid \mathbf{x}_\ell)} \mid \mathbf{x}_{h-1} = x, \mathbf{a}_{h-1} = a \right],
 \end{aligned}$$

$$\begin{aligned}
&= r_{h-1}^*(x, a) + \mathbb{E}_\pi \left[Q_{h,\beta}^\pi(\mathbf{x}_h, \mathbf{a}_h) - \beta \log \frac{\pi_h(\mathbf{a}_h | \mathbf{x}_h)}{\pi_{h,\text{ref}}(\mathbf{a}_h | \mathbf{x}_h)} \mid \mathbf{x}_{h-1} = x, \mathbf{a}_{h-1} = a \right], \\
&= r_{h-1}^*(x, a) \\
&\quad + \mathbb{E} \left[\sum_{a' \in \mathcal{A}} \pi_h(a' | \mathbf{x}_h) \left(Q_{h,\beta}^\pi(\mathbf{x}_h, a') - \beta \log \frac{\pi_h(a' | \mathbf{x}_h)}{\pi_{h,\text{ref}}(a' | \mathbf{x}_h)} \right) \mid \mathbf{x}_{h-1} = x, \mathbf{a}_{h-1} = a \right].
\end{aligned}$$

Therefore, we have

$$\begin{aligned}
&\max_{\pi_{h:H}: \mathcal{X} \rightarrow \Delta(\mathcal{A})} Q_{h-1,\beta}^\pi(x, a) = r_{h-1}^*(x, a) \\
&\quad + \mathbb{E} \left[\max_{\pi_h: \mathcal{X} \rightarrow \Delta(\mathcal{A})} \sum_{a' \in \mathcal{A}} \pi_h(a' | \mathbf{x}_h) \left(\max_{\pi_{h+1:H}: \mathcal{X} \rightarrow \Delta(\mathcal{A})} Q_{h,\beta}^\pi(\mathbf{x}_h, a') - \beta \log \frac{\pi_h(a' | \mathbf{x}_h)}{\pi_{h,\text{ref}}(a' | \mathbf{x}_h)} \right) \mid \mathbf{x}_{h-1} = x, \mathbf{a}_{h-1} = a \right],
\end{aligned}$$

and so using the induction assumption (in particular (168) for $\ell = h$), we get

$$\begin{aligned}
&= r_{h-1}^*(x, a) \\
&\quad + \mathbb{E} \left[\max_{\pi_h: \mathcal{X} \rightarrow \Delta(\mathcal{A})} \sum_{a' \in \mathcal{A}} \pi_h(a' | \mathbf{x}_h) \left(Q_{h,\beta}^*(\mathbf{x}_h, a') - \beta \log \frac{\pi_h(a' | \mathbf{x}_h)}{\pi_{h,\text{ref}}(a' | \mathbf{x}_h)} \right) \mid \mathbf{x}_{h-1} = x, \mathbf{a}_{h-1} = a \right], \\
&= r_{h-1}^*(x, a) + \mathbb{E} \left[\sum_{a' \in \mathcal{A}} \pi_{h,\text{ref}}(a' | \mathbf{x}_h) \cdot e^{Q_{h,\beta}^*(\mathbf{x}_h, a')/\beta} \mid \mathbf{x}_{h-1} = x, \mathbf{a}_{h-1} = a \right], \\
&= r_{h-1}^*(x, a) + \mathcal{T}_{h,\beta}[Q_{h,\beta}^*](x, a), \\
&= Q_{h,\beta}^*(x, a). \tag{170}
\end{aligned}$$

This shows (168) for $\ell = h - 1$. Finally, (169) for $\ell = h - 1$ follows from a direct calculation. \square

Lemma N.5 (Performance difference lemma for KL-regularized regret). *For all $\pi_{1:H}, \pi'_{1:H} \subset \{\pi: \mathcal{X} \rightarrow \Delta(\mathcal{A})\}$, we have*

$$\begin{aligned}
J_\beta(\pi) - J_\beta(\pi') &= \sum_{h=1}^H \mathbb{E}_{\pi'} \left[\sum_{a \in \mathcal{A}} \pi_h(a | \mathbf{x}_h) \cdot \left(Q_{h,\beta}^\pi(\mathbf{x}_h, a) - \beta \cdot \log \frac{\pi_h(a | \mathbf{x}_h)}{\pi_{h,\text{ref}}(a | \mathbf{x}_h)} \right) \right] \\
&\quad - \sum_{h=1}^H \mathbb{E}_{\pi'} \left[\sum_{a \in \mathcal{A}} \pi'_h(a | \mathbf{x}_h) \cdot \left(Q_{h,\beta}^\pi(\mathbf{x}_h, a) - \beta \cdot \log \frac{\pi'_h(a | \mathbf{x}_h)}{\pi_{h,\text{ref}}(a | \mathbf{x}_h)} \right) \right], \tag{171}
\end{aligned}$$

where $J_\beta(\pi) := \sum_{h=1}^H \mathbb{E}_\pi [r_h^*(\mathbf{x}_h, \mathbf{a}_h)] - \beta D_{\text{KL}}(\pi \| \pi_{\text{ref}})$.

Proof of Lemma N.5. Corollary of Lemma N.6 with $h = 1$, taking expectation over $\mathbf{x}_1 \sim \rho$. \square

Lemma N.6 (Performance difference lemma (generalized version)). *For all $h \in [H]$, $x \in \mathcal{X}$, and $\pi_{h:H}, \pi'_{h:H} \subset \{\pi: \mathcal{X} \rightarrow \Delta(\mathcal{A})\}$:*

$$\mathbb{E}_\pi \left[\sum_{\ell=h}^H r_\ell^*(\mathbf{x}_\ell, \mathbf{a}_\ell) - \beta \sum_{\ell=h}^H \log \frac{\pi_\ell(\mathbf{a}_\ell | \mathbf{x}_\ell)}{\pi_{\ell,\text{ref}}(\mathbf{a}_\ell | \mathbf{x}_\ell)} \mid \mathbf{x}_h = x \right]$$

$$\begin{aligned}
 & - \mathbb{E}_{\pi'} \left[\sum_{\ell=h}^H r_{\ell}^{\star}(\mathbf{x}_{\ell}, \mathbf{a}_{\ell}) - \beta \sum_{\ell=h}^H \log \frac{\pi'_{\ell}(\mathbf{a}_{\ell} | \mathbf{x}_{\ell})}{\pi_{\ell, \text{ref}}(\mathbf{a}_{\ell} | \mathbf{x}_{\ell})} \mid \mathbf{x}_h = x \right] \\
 & = \sum_{\ell=h}^H \mathbb{E}_{\pi'} \left[\sum_{a \in \mathcal{A}} \pi_{\ell}(a | \mathbf{x}_{\ell}) \cdot \left(Q_{\ell, \beta}^{\pi}(\mathbf{x}_{\ell}, a) - \beta \cdot \log \frac{\pi_{\ell}(a | \mathbf{x}_{\ell})}{\pi_{\ell, \text{ref}}(a | \mathbf{x}_{\ell})} \right) \mid \mathbf{x}_h = x \right] \\
 & - \sum_{\ell=h}^H \mathbb{E}_{\pi'} \left[\sum_{a \in \mathcal{A}} \pi'_{\ell}(a | \mathbf{x}_{\ell}) \cdot \left(Q_{\ell, \beta}^{\pi}(\mathbf{x}_{\ell}, a) - \beta \cdot \log \frac{\pi'_{\ell}(a | \mathbf{x}_{\ell})}{\pi_{\ell, \text{ref}}(a | \mathbf{x}_{\ell})} \right) \mid \mathbf{x}_h = x \right].
 \end{aligned}$$

Proof of Lemma N.6. First, for any $\pi_{1:H} \subset \{\pi: \mathcal{X} \rightarrow \Delta(\mathcal{A})\}$, $(x, a) \in \mathcal{X} \times \mathcal{A}$, $h \in [H]$, define

$$r_h^{\pi}(x, a) := r_h^{\star}(x, a) - \beta \log \frac{\pi_h(a | x)}{\pi_{h, \text{ref}}(a | x)}, \quad \text{and} \quad \bar{Q}_{h, \beta}^{\pi}(x, a) = \mathbb{E}_{\pi} \left[\sum_{\ell=h}^H r_{\ell}^{\pi}(\mathbf{x}_{\ell}, \mathbf{a}_{\ell}) \mid \mathbf{x}_h = x, \mathbf{a}_h = a \right]$$

and note that

$$Q_{h, \beta}^{\pi}(x, a) = \bar{Q}_{h, \beta}^{\pi}(x, a) + \beta \log \frac{\pi_h(a | x)}{\pi_{h, \text{ref}}(a | x)}. \quad (172)$$

We need to show that for all $h \in [H]$, $x \in \mathcal{X}$, and $\pi_{h:H}, \pi'_{h:H} \subset \{\pi: \mathcal{X} \rightarrow \Delta(\mathcal{A})\}$:

$$\begin{aligned}
 & \mathbb{E}_{\pi' \circ_h \pi} \left[\sum_{\ell=h}^H r_{\ell}^{\pi}(\mathbf{x}_{\ell}, \mathbf{a}_{\ell}) \mid \mathbf{x}_h = x \right] - \mathbb{E}_{\pi'} \left[\sum_{\ell=h}^H r_{\ell}^{\pi'}(\mathbf{x}_{\ell}, \mathbf{a}_{\ell}) \mid \mathbf{x}_h = x \right] \\
 & = \sum_{\ell=h}^H \mathbb{E}_{\pi'} \left[\sum_{a \in \mathcal{A}} \pi_{\ell}(a | \mathbf{x}_{\ell}) \cdot \left(Q_{\ell, \beta}^{\pi}(\mathbf{x}_{\ell}, a) - \beta \cdot \log \frac{\pi_{\ell}(a | \mathbf{x}_{\ell})}{\pi_{\ell, \text{ref}}(a | \mathbf{x}_{\ell})} \right) \mid \mathbf{x}_h = x \right] \\
 & - \sum_{\ell=h}^H \mathbb{E}_{\pi'} \left[\sum_{a \in \mathcal{A}} \pi'_{\ell}(a | \mathbf{x}_{\ell}) \cdot \left(Q_{\ell, \beta}^{\pi}(\mathbf{x}_{\ell}, a) - \beta \cdot \log \frac{\pi'_{\ell}(a | \mathbf{x}_{\ell})}{\pi_{\ell, \text{ref}}(a | \mathbf{x}_{\ell})} \right) \mid \mathbf{x}_h = x \right].
 \end{aligned}$$

Fix $h \in [H]$, $x \in \mathcal{X}$, and $\pi_{1:H}, \pi'_{1:H} \subset \{\pi: \mathcal{X} \rightarrow \Delta(\mathcal{A})\}$. We now show via induction over $j = h, \dots, H+1$ that

$$\begin{aligned}
 & \mathbb{E}_{\pi' \circ_h \pi} \left[\sum_{\ell=h}^H r_{\ell}^{\pi}(\mathbf{x}_{\ell}, \mathbf{a}_{\ell}) \mid \mathbf{x}_h = x \right] - \mathbb{E}_{\pi'} \left[\sum_{\ell=h}^H r_{\ell}^{\pi'}(\mathbf{x}_{\ell}, \mathbf{a}_{\ell}) \mid \mathbf{x}_h = x \right] \\
 & = \sum_{\ell=h}^{j-1} \mathbb{E}_{\pi'} \left[\sum_{a \in \mathcal{A}} \pi_{\ell}(a | \mathbf{x}_{\ell}) \cdot \bar{Q}_{\ell, \beta}^{\pi}(\mathbf{x}_{\ell}, a) - \sum_{a \in \mathcal{A}} \pi'_{\ell}(a | \mathbf{x}_{\ell}) \cdot \bar{Q}_{\ell, \beta}^{\pi}(\mathbf{x}_{\ell}, a) \mid \mathbf{x}_h = x \right] \\
 & + \beta \sum_{\ell=h}^{j-1} \mathbb{E}_{\pi'} \left[\log \frac{\pi'_{\ell}(\mathbf{a}_{\ell} | \mathbf{x}_{\ell})}{\pi_{\ell}(\mathbf{a}_{\ell} | \mathbf{x}_{\ell})} \mid \mathbf{x}_h = x \right] \\
 & + \mathbb{E}_{\pi' \circ_j \pi} \left[\sum_{\ell=j}^H r_{\ell}^{\pi}(\mathbf{x}_{\ell}, \mathbf{a}_{\ell}) \mid \mathbf{x}_h = x \right] - \mathbb{E}_{\pi'} \left[\sum_{\ell=j}^H r_{\ell}^{\pi'}(\mathbf{x}_{\ell}, \mathbf{a}_{\ell}) \mid \mathbf{x}_h = x \right]. \quad (173)
 \end{aligned}$$

The base case $j = h$ is trivial. Now assume that (173) holds for $j \in \{h, \dots, H+1\}$; we show it holds for $j+1$:

$$\mathbb{E}_{\pi' \circ_j \pi} \left[\sum_{\ell=j}^H r_{\ell}^{\pi}(\mathbf{x}_{\ell}, \mathbf{a}_{\ell}) \mid \mathbf{x}_h = x \right] - \mathbb{E}_{\pi'} \left[\sum_{\ell=j}^H r_{\ell}^{\pi'}(\mathbf{x}_{\ell}, \mathbf{a}_{\ell}) \mid \mathbf{x}_h = x \right]$$

$$\begin{aligned}
 &= \mathbb{E}_{\pi' \circ_j \pi} \left[\sum_{\ell=j}^H r_{\ell}^{\pi}(\mathbf{x}_{\ell}, \mathbf{a}_{\ell}) \mid \mathbf{x}_h = x \right] - \mathbb{E}_{\pi' \circ_{j+1} \pi} \left[r_{\ell}^{\pi'}(\mathbf{x}_{\ell}, \mathbf{a}_{\ell}) + \sum_{\ell=j+1}^H r_{\ell}^{\pi}(\mathbf{x}_{\ell}, \mathbf{a}_{\ell}) \mid \mathbf{x}_h = x \right] \\
 &\quad + \mathbb{E}_{\pi' \circ_{j+1} \pi} \left[r_{\ell}^{\pi'}(\mathbf{x}_{\ell}, \mathbf{a}_{\ell}) + \sum_{\ell=j+1}^H r_{\ell}^{\pi}(\mathbf{x}_{\ell}, \mathbf{a}_{\ell}) \mid \mathbf{x}_h = x \right] - \mathbb{E}_{\pi'} \left[\sum_{\ell=j}^H r_{\ell}^{\pi'}(\mathbf{x}_{\ell}, \mathbf{a}_{\ell}) \mid \mathbf{x}_h = x \right], \\
 &= \mathbb{E}_{\pi' \circ_j \pi} \left[\sum_{\ell=j}^H r_{\ell}^{\pi}(\mathbf{x}_{\ell}, \mathbf{a}_{\ell}) \mid \mathbf{x}_h = x \right] - \mathbb{E}_{\pi' \circ_{j+1} \pi} \left[r_{\ell}^{\pi'}(\mathbf{x}_{\ell}, \mathbf{a}_{\ell}) + \sum_{\ell=j+1}^H r_{\ell}^{\pi}(\mathbf{x}_{\ell}, \mathbf{a}_{\ell}) \mid \mathbf{x}_h = x \right] \\
 &\quad + \mathbb{E}_{\pi' \circ_j \pi} \left[\sum_{\ell=j+1}^H r_{\ell}^{\pi}(\mathbf{x}_{\ell}, \mathbf{a}_{\ell}) \mid \mathbf{x}_h = x \right] - \mathbb{E}_{\pi'} \left[\sum_{\ell=j+1}^H r_{\ell}^{\pi'}(\mathbf{x}_{\ell}, \mathbf{a}_{\ell}) \mid \mathbf{x}_h = x \right], \\
 &= \mathbb{E}_{\pi' \circ_j \pi} \left[\sum_{\ell=j}^H r_{\ell}^{\pi}(\mathbf{x}_{\ell}, \mathbf{a}_{\ell}) \mid \mathbf{x}_h = x \right] - \mathbb{E}_{\pi' \circ_{j+1} \pi} \left[\sum_{\ell=j}^H r_{\ell}^{\pi}(\mathbf{x}_{\ell}, \mathbf{a}_{\ell}) \mid \mathbf{x}_h = x \right] \\
 &\quad + \mathbb{E}_{\pi'} \left[r_{\ell}^{\pi}(\mathbf{x}_{\ell}, \mathbf{a}_{\ell}) - r_{\ell}^{\pi'}(\mathbf{x}_{\ell}, \mathbf{a}_{\ell}) \mid \mathbf{x}_h = x \right] \\
 &\quad + \mathbb{E}_{\pi' \circ_{j+1} \pi} \left[\sum_{\ell=j+1}^H r_{\ell}^{\pi}(\mathbf{x}_{\ell}, \mathbf{a}_{\ell}) \mid \mathbf{x}_h = x \right] - \mathbb{E}_{\pi'} \left[\sum_{\ell=j+1}^H r_{\ell}^{\pi'}(\mathbf{x}_{\ell}, \mathbf{a}_{\ell}) \mid \mathbf{x}_h = x \right], \\
 &= \mathbb{E}_{\pi' \circ_j \pi} \left[\sum_{a \in \mathcal{A}} \pi_j(a \mid \mathbf{x}_{\ell}) \cdot \bar{Q}_{j,\beta}^{\pi}(\mathbf{x}_j, a) - \sum_{a \in \mathcal{A}} \pi'_j(a \mid \mathbf{x}_j) \cdot \bar{Q}_{j,\beta}^{\pi}(\mathbf{x}_j, a) \mid \mathbf{x}_h = x \right] \\
 &\quad + \beta \mathbb{E}_{\pi'} \left[\log \frac{\pi'_j(\mathbf{a}_j \mid \mathbf{x}_j)}{\pi_j(\mathbf{a}_j \mid \mathbf{x}_j)} \mid \mathbf{x}_h = x \right] \\
 &\quad + \mathbb{E}_{\pi' \circ_j \pi} \left[\sum_{\ell=j+1}^H r_{\ell}^{\pi}(\mathbf{x}_{\ell}, \mathbf{a}_{\ell}) \mid \mathbf{x}_h = x \right] - \mathbb{E}_{\pi'} \left[\sum_{\ell=j+1}^H r_{\ell}^{\pi'}(\mathbf{x}_{\ell}, \mathbf{a}_{\ell}) \mid \mathbf{x}_h = x \right]. \tag{174}
 \end{aligned}$$

This shows that (173) holds with j replaced by $j + 1$ and completes the induction. Instantiating (173) with $\ell = H + 1$ shows that

$$\begin{aligned}
 &\mathbb{E}_{\pi' \circ_h \pi} \left[\sum_{\ell=h}^H r_{\ell}^{\pi}(\mathbf{x}_{\ell}, \mathbf{a}_{\ell}) \mid \mathbf{x}_h = x \right] - \mathbb{E}_{\pi'} \left[\sum_{\ell=h}^H r_{\ell}^{\pi'}(\mathbf{x}_{\ell}, \mathbf{a}_{\ell}) \mid \mathbf{x}_h = x \right] \\
 &= \sum_{\ell=h}^H \mathbb{E}_{\pi'} \left[\sum_{a \in \mathcal{A}} \pi_{\ell}(a \mid \mathbf{x}_{\ell}) \cdot \bar{Q}_{\ell,\beta}^{\pi}(\mathbf{x}_{\ell}, a) - \sum_{a \in \mathcal{A}} \pi'_{\ell}(a \mid \mathbf{x}_{\ell}) \cdot \bar{Q}_{\ell,\beta}^{\pi}(\mathbf{x}_{\ell}, a) + \beta \log \frac{\pi'_{\ell}(\mathbf{a}_{\ell} \mid \mathbf{x}_{\ell})}{\pi_{\ell}(\mathbf{a}_{\ell} \mid \mathbf{x}_{\ell})} \mid \mathbf{x}_h = x \right].
 \end{aligned}$$

Combining this with (172) implies the desired result. \square

Lemma N.7. Let $\mathcal{C} \subset \mathcal{X} \times \mathcal{A}$ be a multiset of the form

$$\mathcal{C} = \bigcup_{i \in [N]} \{(x_i, a_i), (x_i, \mathbf{a})\}, \tag{175}$$

for $N \geq 1$. Then, for any non-negative $f: \mathcal{X} \times \mathcal{A} \rightarrow \mathbb{R}$, we have

$$\sum_{(x,a) \in \mathcal{C}} f(x, a) + \sum_{(x,a) \in \mathcal{C}} f(x, \mathfrak{a}) \leq 3 \sum_{(x,a) \in \mathcal{C}} f(x, a). \quad (176)$$

Proof of Lemma N.7. Because \mathcal{C} is a multiset satisfying (175) and f is non-negative, we have

$$\begin{aligned} \sum_{(x,a) \in \mathcal{C}} f(x, a) &\geq \sum_{(x,a) \in \mathcal{C}: a \neq \mathfrak{a}} f(x, a) + \sum_{(x,a) \in \mathcal{C}: a \neq \mathfrak{a}} f(x, \mathfrak{a}), \\ &\geq \sum_{(x,a) \in \mathcal{C}: a \neq \mathfrak{a}} f(x, \mathfrak{a}). \end{aligned} \quad (177)$$

On the other hand, we also have that

$$\begin{aligned} \sum_{(x,a) \in \mathcal{C}} f(x, a) &= \sum_{(x,a) \in \mathcal{C}: a \neq \mathfrak{a}} f(x, a) + \sum_{(x,a) \in \mathcal{C}: a = \mathfrak{a}} f(x, \mathfrak{a}), \\ &\geq \sum_{(x,a) \in \mathcal{C}: a = \mathfrak{a}} f(x, \mathfrak{a}). \end{aligned} \quad (178)$$

Combining (1) and (2) implies that

$$\begin{aligned} \sum_{(x,a) \in \mathcal{C}} f(x, \mathfrak{a}) &= \sum_{(x,a) \in \mathcal{C}: a = \mathfrak{a}} f(x, \mathfrak{a}) + \sum_{(x,a) \in \mathcal{C}: a \neq \mathfrak{a}} f(x, \mathfrak{a}), \\ &\leq 2 \sum_{(x,a) \in \mathcal{C}} f(x, a), \end{aligned}$$

which implies (176) after adding $\sum_{(x,a) \in \mathcal{C}} f(x, a)$ on both sides. \square