Submitted to *Operations Research*
manuscript (Please, provide the manuscript number!)

Online Bayesian Recommendation with No Regret

Yiding Feng
University of Chicago, yidingfeng@uchicago.edu

Wei Tang
Columbia University, wt2359@columbia.edu

Haifeng Xu
University of Chicago, haifengxu@uchicago.edu

We introduce and study the online Bayesian recommendation problem for a recommender system platform. The platform has the privilege to privately observe a utility-relevant *state* of a product at each round and uses this information to make online recommendation to a stream of myopic users. This paradigm is common in a wide range of scenarios in the current Internet economy. The platform commits to an online recommendation policy that utilizes her information advantage on the product state to persuade self-interested user to follow the recommendation. Since the platform does not know users' preferences neither beliefs in advance, we study the platform's online learning problem of designing adaptive recommendation policy to persuade users while gradually learning users' preferences and beliefs en route.

Specifically, we aim to design online learning policies with no *Stackelberg regret* for the platform, i.e., against the optimal benchmark policy in hindsight under the assumption that users will correspondingly adapt their responses to the benchmark policy. Our first result is an online policy that achieves double logarithm regret dependence on the number of rounds. We then present an information theoretic lower bound showing that no adaptive online policy can achieve regret with better dependency on the number of rounds. Finally, by formulating the platform's problem as optimizing a linear program with membership oracle access, we present our second online recommendation policy that achieves regret with polynomial dependence on the number of states but logarithm dependence on the number of rounds.*

Key words: Online learning, regret minimization, linear program

1. Introduction

Thanks to the rapid growth of modern technology, online platforms have become a major component of today's economy. By the end of 2021, there are at least 30 social platforms with at least 100 million monthly active users, and seven of them have more than 1 billion users. Based on a recent report (Knowledge Sourcing Intelligence LLP 2021), the global networking platforms market is evaluated at

* A preliminary one-page extended abstract of this work has appeared in the proceeding of the Twenty-Third ACM Conference on Economics and Computation (EC'22) (Feng et al. 2022).

192 billion U.S. dollar for the year 2019 and is projected to reach a market size of 940 billion U.S. dollar by the year 2026. Numerous important algorithmic problems rise in this expanding industry.

Within these platforms, the recommended items are often associated with different labels that are used to signal varying degrees of relevance or quality of the items to the user. For example, music streaming platforms like Spotify use curated playlists with titles like "Discover Weekly" or "Release Radar" to signal the novelty or relevance of the recommended tracks. E-commerce platforms like Amazon use labels such as "Top Picks", "Best Sellers" and "Customers Also Bought" to categorize its product recommendations, each conveying the popularity level of the items. From platforms' perspective, these labels can serve as informational cues that guide user behavior and decision-making. This enables us to study the recommendation system in the language of the information design.

A prominent example of the recommendation in social platforms, which is a major motivating application of this work, is the *video recommendation* in short-video platforms such as TikTok, Instagram Reels and YouTube Shorts. Taking TikTok as an example, it has a trademark feature – "For You". It is a feed of videos that are recommended to a user in real time. In particular, there will be one video displayed per time, and the user can decide either to watch it or skip (i.e., not watch) it. If the user starts watching the video, the profit is generated for the platform (e.g., through in-stream ads, sponsored videos). We highlight two features in this application. First, this system not only recommends videos that the user is familiar with, but also intersperses diverse types of videos which may be potentially interesting to the user. Second, the recommendation decision is adaptively formed based on the user's interaction history (which reveals information about the user's personal preference and belief[1]) and the information of the video (e.g., captions, sounds, hashtags). Other real-time recommendation applications with similar features also appear in various key e-commerce platforms such as Amazon Live and Taobao Live .

Motivated by the above applications, we introduce and study the *online Bayesian recommendation* problem. Here we describe the problem in the context of video recommendation. Consider a sequential interaction between a video platform and a population of users with the same private preference and belief.[2] At each time, there is a video displayed by the platform to an incoming user.[3] To capture the uncertain characteristics of the video, we study a *Bayesian* model, in which the payoff-relevant

[1] In lots of applications such as TikTok, there is no interface for the user to directly report his preference, belief or manually customize his recommendation policy.

[2] Or equivalently, a repeated interaction between the platform and a myopic user. Taking Tiktok as an example, it has been reported that the average TikTok user spends 52 minutes on the app each day and a quarter of the highest performing videos on TikTok are in between 21 and 34 seconds, i.e., roughly $92 \sim 150$ videos to cover an average user's stay (see Flixier 2022); meanwhile, it is best to recommend ads videos with a length of 9 to 15 seconds i.e., roughly $208 \sim 350$ videos to cover an average user's stay (see Geyser 2022).

[3] In practice, platforms make joint decisions on which video to display and how to recommend. Here we decouple them and focus on the recommendation problem, by assuming that the displaying decision is made exogenously.

characteristics of the video is captured by a (random) *state* of the video. The platform and user each have their own preferences over the video states, which are captured by their utility functions respectively. We assume a natural *information asymmetry* between the platform and users — only the platform can privately observe the realized state of each video, whereas all users only have a prior belief about the video state. Notably, the platform also has its own prior belief over the video state, which is allowed to be different from the users' belief (after all, they form such beliefs from completely different sources). The platform designs and commits to a recommendation policy which makes different "levels" of recommendation (e.g., "not recommended", "standard", "recommended", "highly recommended") based on her private information about the video, i.e., its realized state. After observing the recommendation level, together with his initial belief, the user forms a posterior belief about the video and decides either to watch this video or skip it.

In the idealized situation when the platform knew both the user's preferences and prior beliefs, this sequential Bayesian recommendation problem reduces to be a standard Bayesian persuasion problem and thus can be solved by linear programming (Kamenica and Gentzkow 2011, Alonso and Camara 2016, Dughmi and Xu 2019). This paper, however, addresses the more realistic yet challenging situation in which the platform does not know user's preferences neither user's prior beliefs. Therefore, the platform has to adaptively update her recommendation strategy based on the past users' behaviors, so as to maximize its own accumulated utility. The goal of this paper is to design online learning policies with no *Stackelberg regret* for the platform. Notably, the Stackelberg regret is a new regret notion recently developed for strategic settings (Dong et al. 2018, Chen et al. 2020), which compares to the optimal policy in hindsight, assuming users will correspondingly adapt their behaviors to the benchmark policy (thus the "Stackelberg" in its name). While previous works demonstrated the difficulty of obtaining sublinear Stackelberg regret in online classification problems Chen et al. (2020), we surprisingly show that our problem admits efficient online learning algorithms with Stackelberg regret that only has logarithmic dependence on the number of rounds T.

1.1. Our Contributions and Techniques

In this work, we introduce a novel online Bayesian recommendation framework that addresses the challenge of making the recommendations to arriving users with unknown preferences, given the information asymmetry between the online platform and the users. In below, we focus on summarizing our contributions and results when the users share the same prior belief with platform, namely, only the users' preferences are unknown to the platform.[4]

When the platform has the complete knowledge of users' preferences, the optimal signaling schemes in all rounds are identical and can be solved separately as a classic Bayesian persuasion problem

[4] The results for the users with different prior belief are provided in Section A.

(Kamenica and Gentzkow 2011, Alonso and Camara 2016, Dughmi and Xu 2019). By the revelation principle, this optimal signaling scheme in hindsight is a ***direct*** signaling scheme which has binary recommendation level. In particular, the optimal signaling scheme will correspond each state with the users' ***preference difference***, which represents how much the user prefers watching the video over not watching the video given this particular state. Then, the optimal signaling scheme specifies an ***order*** over all states based on the users' preference differences as well as a ***threshold state*** such that it recommends every state above the threshold state in this order. The threshold state is selected such that whenever the signaling scheme recommends a video, the user is indifferent between watching and skipping it.

When the platform has no knowledge of users' preferences, the platform has to use adaptive signaling schemes to learn the correct order of the states, as well as the threshold state, to achieve the optimal long-term revenue. To understand and solve the platform's problem, we focus on two natural scenarios: (1) known ordinal preference – the order of the users' preference difference is known to the platform, but the exact differences are unknown to the platform;[5] (2) unknown ordinal preference: the order of the users' preference difference is unknown to the platform. We summarize our results in Table 1.

		Upper bound	Lower bound
Known ordinal preference		$O(\log\log T)^*$ [Theorem 1]	$\Omega(\log\log T)^*$ [Theorem 4]
Unknown ordinal preference	Affine preference	$O(\log\log T)^*$ [Proposition 1]	
	Arbitrary preference	$O(m2^{m-1}\log\log T \wedge m^6 \log^{O(1)}(mT))$ [Proposition 2, Theorem 2]	

Table 1 Regret bounds of the online Bayesian recommendation problem. Here m denotes the number of states, and T denotes the time horizon. (*): These regret bounds have no dependence on the the number of states m.

Before diving into the detail discussion of our results, we first highlight one crucial feature in our model – the feedback to the the platform is ***limited*** and ***probabilistic***. This feedback structure is one of the major issues that algorithms with low regret have to overcome or bypass, which distinguishes our model from other classic models in the online learning literature. Specifically, in our model, a recommendation strategy (aka., a signaling scheme) maps each video state to a possibly random recommendation level (aka., a signal); therefore, the platform only observes the user's response to this realized recommendation level but nothing about other recommendation levels. The feedback

[5] In practice, the online platform may infer the user's ordinal preferences – that is, the relative ranking of videos based on perceived interest or relevance, from offline data like past interactions. However, the platform may find it hard to learn the degree to which a user enjoys a particular video – that is, how much the user prefers watching this video than not watching the video.

is also probabilistic, since the realized recommendation level depends on the realized (video) state, which is drawn from an exogenous prior distribution. As a consequence, the platform may incur a large regret in order to learn the user's response to a specific signal realization, or his preference for a specific state.

Known ordinal preference. For the known ordinal preference scenario, it suffices for the platform to learn which state is the threshold state, and how to recommend when this threshold state is realized. For this scenario, we show that there exists a Conservative Recommendation Policy, henceforth ConRP (Algorithm 2), such that its regret has the double logarithm dependence on the number of rounds T, i.e., $O(\log\log T)$. An informal statement of our first main result is as follows (see Theorem 1 for the formal result).

THEOREM (INFORMAL) *ConRP achieves $O(\log\log T)$ regret.*

The key intuition behind ConRP is that the platform's expected payoff of a given signaling scheme exhibits a "asymmetric" structure: the user will not chose to watch the video when the issued signaling scheme is overoptimistic (e.g., a signaling scheme that always recommend the user to watch the video) no matter what signal is realized, and thus the platform's expected payoff is zero; on the other hand, when the issued signaling scheme is not overoptimistic, then there always exists positive probability such that the user will chose to watch the video, and thus the platform has non-zero expected payoff. Thus, one can use a "conservative" binary search to identify the threshold state and determine how to recommend given this threshold state.

However, due to the above mentioned limited and probabilistic feedback, additional careful treatments are needed to ensure the low regret. Specifically, since the feedback is limited, instead of learning the user's preference for each state separately, ConRP essentially batches states and learn the aggregated preference for the whole batch. To bypass the challenge due to the probabilistic feedback, ConRP utilizes a preprocessing step to pin down a rough range of the optimal payoff. By restricting to signaling schemes with payoff in this range, we ensure that the expected regret to learn the user's response to a specific signal realization is constant.

Unknown ordinal preference. For the unknown ordinal preference scenario, the order as well as the threshold state specified in the optimum signaling scheme in hindsight remains unknown. Due to the limited and probabilistic feedback feature, designing an online policy to pin down this order with logarithm regret may appear impossible at the first glance. However, we show that when the users' preferences are affine dependent over the states,[6] one can still achieve regret with double logarithm

[6] Formally speaking, a user has affine state-dependent preference if her expected utility can be uniquely determined by the mean of her posterior belief (Candogan and Strack 2021).

dependence on number of rounds T, and moreover, this regret is independent of the number of states, and also holds even for the continuous state space. For arbitrary users' preferences, a modified ConRP, which enumerates over all possible orders over all states and prunes out the bad orders in the process, can lead to a regret with still double logarithm dependence on number of rounds T but with an exponential dependence on the number of states m. The informal statement of these results is stated as follows (see Proposition 1 and Proposition 2 for the formal results).

PROPOSITION (INFORMAL) *For a ne state-dependent preferences, a modified ConRP has expected regret $O(\log\log T)$; for arbitrary preferences, a modified ConRP has expected regret $O(m2^{m-1} \cdot \log\log T)$.*

A caveat of the above results is that the regret dependence on the number of states m is exponential for arbitrary users' preferences, though we argue that for a wide range of applications, it is reasonable to focus on problem instances with small m.[7] Nonetheless, to also shed lights for problem instances with large m, we introduce another policy, a Linear Program-based Recommendation Policy, henceforth LP-RP (Algorithm 3), whose regret dependence is polynomial on m and logarithm on T, i.e., $O(\mathsf{poly}(m\log T))$ for arbitrary users' preferences. We obtain LP-RP by formulating the problem as optimizing a linear program with membership oracle access. In particular, the optimal signaling scheme in hindsight can be formulated as the optimal solution to a linear program as follows. Every feasible solution corresponds to a signaling scheme. The objective is the platform's utility; the constraints are the feasibility constraint and the obedience constraint. Here the feasibility constraint ensures that every feasible solution of the linear program is indeed a signaling scheme, and the obedience constraint ensures that the user prefers to follow the recommendation. When the platform has no knowledge of users' preferences, the obedience constraint becomes unknown. Nonetheless, the platform may check the obedience of a given signaling scheme by deploying this signaling scheme to users. In this sense, the platform obtains a membership oracle for the aforementioned linear program. This leads to our following guarantee about LP-RP.

THEOREM (INFORMAL) *LP-RP achieves $O(\mathsf{poly}(m\log T))$ regret.*

We note that similar ideas of formulating the learning problems as optimizing linear programs have also been applied to other online learning problem such as contextual dynamic pricing (e.g., Leme and Schneider 2018) and security game (e.g., Blum et al. 2014). However, our LP-RP requires additional special treatment to overcome the issue of probabilistic signals. Moreover, comparing with using the separation oracle as in (Leme and Schneider 2018, Blum et al. 2014), our problem of linear

[7] In our recommendation problem, two videos should be considered as having different states if (a) the platform has enough information to distinguish them, and (b) the user's utility for watching them are different.

optimization with membership oracle access is considerably harder. For instance, one key technical hurdle, which does not appear in previous works but our LP-RP has to overcome, is to construct an *interior point* inside the feasible region.

Lower bound. Similar to the optimal policy in hindsight, ConRP and LP-RP only use direct signaling schemes with *binary* recommendation levels. Such direct signaling schemes are prevalent in many real-world applications such as "For You" in TikTok. However, when the platform does not know and has to learn users' preferences, the revelation principle does not necessarily hold — i.e., it is unclear whether restricting to direct signaling schemes with binary recommendation levels is still without loss of generality during learning. Our third main result provides an affirmative answer, showing that introducing more recommendation levels cannot improve the regret dependence on T.[8]

THEOREM (INFORMAL) *No online policy can achieve a regret better than* $\Omega(\log\log T)$ *even for problem instances with binary state.*

We note that the above lower bound holds for all scenarios we mentioned before. To show this impossibility result, we first construct a reduction from the single-item dynamic pricing problem (cf. Kleinberg and Leighton 2003) to a special case of our online Bayesian recommendation problem, where the state space is restricted to be binary, and the signaling schemes are restricted to have binary signal space.[9] Then, we argue that in the online Bayesian recommendation problem, when the state space is binary, every online policy can be converted into an online policy which only uses direct signaling scheme with the same regret.

Simulations. We also provide numerical experiments to evaluate the empirical performance of our proposed algorithm and highlight some of its salient features. In particular, in our simulations, we evaluate the performance of our proposed algorithm ConRP, and compare its performance with several benchmarks, including the benchmarks that use simple searching strategies to find out the optimal signaling schemes without considering the unique structure of our problem, and also the benchmarks that use simple signaling schemes. We observe that our algorithm significantly outperforms all of these benchmarks. The results not only demonstrate the benefits of using partial information revealing in platform's problem, but also show the efficiency of our algorithm.

Organization. Our paper is organized as follows. After discussing the related works at the end of this section, we formulate our problem in Section 2. In Section 3, we present our algorithm ConRP and the regret analysis of the proposed algorithm when the platform knows the users' ordinal preference.

[8] Though we remark that studying the revelation principle for repeated learning tasks is a very intriguing but generally quite challenging task since the regret analysis is usually order-wise analysis while not exact calculation.

[9] Loosely speaking, this reduction suggests that our problem is harder than the dynamic pricing problem.

In Section 4, we present algorithms and also the regret analysis when the users' ordinal preference is unknown to the platform. In Section 5, we present another algorithm `LP-RP` and its regret analysis for arbitrary users' preference. We provide simulations in Section 6 and provide the lower bound analysis in Section 7. We conclude the paper in Section 8. In appendix, we provide the extensions of our results in Section A and all the missing proofs.

1.2. Related Work

Our work connects to several strands of existing literature. First, when user's preference is known and shares the same prior belief with the platform, the one-shot instantiation of our problem exactly follows the formulation of the canonical Bayesian persuasion problem (Kamenica and Gentzkow 2011). Bayesian persuasion concerns the problem that an informed sender (i.e., platform) designs an information structure (i.e., signaling scheme) to influence the behavior of a receiver (i.e., user). There is a growing literature, including our work, on studying the relaxation of one fundamental assumption in the Bayesian persuasion model – The sender perfectly knows receiver's preference and his prior belief. There are generally two approaches to deal with such sender's uncertainty: the robust approach (Dworczak and Pavan 2020, Babichenko et al. 2021, Kosterina 2018, Hu and Weng 2021) which tries to design signaling schemes that perform robustly well for all possible receiver's utilities; the online learning approach (Castiglioni et al. 2020, 2021, Zu et al. 2021) which studies the regret minimization when the sender repeatedly interacts with receivers. [10] Our work falls into the second approach. In particular, Castiglioni et al. (2020) concerns the sender interacting with receivers who have the unknown type. They provide an algorithm with regret guarantee $O(T^{4/5})$ but has exponential running-time over the number of states. Zu et al. (2021) studies a setting where the sender has unknown prior distribution, and they require sender to make obedient signaling schemes at each round. They provide an algorithm with $O(\sqrt{T})$ regret bound, and also demonstrate that it is tight whenever the receiver has five (or more) actions. Our work differs from the above works in many ways. First, instead of assuming unknown types, our setting directly relaxes the knowledge on user's utilities. Second, we do not require platform's singling scheme to be obedient at each round. Third, we achieve logarithmic regrets over the time horizon and this is possible due to the special structure of the Bayesian recommendation problem.

Second, our work also relates to research on Bayesian exploration in multi-armed bandit (Kremer et al. 2014, Mansour et al. 2015, 2021). In both our Bayesian recommendation and their Bayesian exploration, the platform utilizes her information advantage to persuade the user to take the desired action, and the user observes the platform's message and forms his posterior which will be used to

[10] We refer the reader to the work by Dworczak and Pavan (2020), Babichenko et al. (2021) for a comprehensive overview on different methods in the robust approach.

pick his optimal action. However, in Bayesian exploration, the platform is learning the true state of the nature, which is realized at the very beginning and never changes afterwards, and is required to make incentive-compatible action recommendation at each time round. While in our setup, the platform is learning the users' preferences and beliefs, and the state is realized independently across the time horizon. Additionally, our problem do not require the platform to make incentive-compatible recommendations. Thus, the analysis and the technique of this work are quite different from theirs.

Third, our setting with homogeneous users shares the similarity to the fixed valuation in (contextual) dynamic pricing literature (Lobel et al. 2018, Kleinberg and Leighton 2003, Leme and Schneider 2018, Liu et al. 2021), where the logarithm regrets are also achievable. Part of our analysis is also related to this line of literature. In particular, we prove our lower bound via a non-trivial reduction to the single-item dynamic pricing problem. Though it is seemingly that our problem for multiple states shares the similarity to the contextual dynamic pricing (e.g., we both need to learn an unknown vector: in our setting, it is the user's preference of product state, and in contextual pricing, it is buyer's preference of product features), we note that there are significant differences in our problem structure like the platform's actions, and the probabilistic feedback from users (see the end of Section 7 for the detailed comparisons).

Our work is also conceptually similar to the multinomial logit (MNL) bandit with applications to online assortment (e.g., Rusmevichientong et al. 2010, Sauré and Zeevi 2013, Agrawal et al. 2019, Chen et al. 2021). In this model, a seller with m products sequentially interacts with a population of consumer with the same private preference $\{v_i\}_{i\in[m]}$ in T rounds. Similar to our model, in each round the seller needs to make a high-dimensional decision (i.e., display a subset of products, i.e., an assortment) to the arriving consumer, and then observes a probabilistic feedback (i.e., the purchasing decision of the consumer – the probability of purchasing each product is proportional to the private preference). The optimal regret achievable in this model is $\tilde{O}(\sqrt{mT})$ (Agrawal et al. 2019).

2. Preliminary

2.1. Basic Setup

Motivated by the applications of short-video platforms, this paper introduces and studies the *Bayesian recommendation* problem. We begin with describing a static model and then introduces the online setup studied in this work.

In the static model, there are two players: a platform and a user.[11] The platform wants to recommend a video to the user. The video is associated with a private state θ drawn from a finite set $[m] \triangleq \{1,\dots,m\}$ according to a prior distribution $\lambda \in \Delta([m])$, which is common knowledge among

[11] In this paper, we use "she" to denote the platform and "he" to denote the user.

both players. We use notation θ to denote the state as a random variable, and $i, j \in [m]$ as its possible realizations. The user has a binary-action set $\mathcal{A} = \{0, 1\}$ (i.e., not watch or watch), and a utility function $\rho : [m] \times \mathcal{A} \to \mathbb{R}$ mapping from the state of the video and his action to his utility. The platform has a state-independent utility function $\xi : \mathcal{A} \to \mathbb{R}$ that only depends on the user's action a. For ease of presentation, our main context will focus on a stylized setup where (i) the platform and the user share the same prior belief λ over $[m]$; (ii) the platform has state-independent utility function and only benefits from the user's action 1 (i.e., click). In Section A, we illustrate how our algorithms and results can be easily extended to general settings where the users might have different prior beliefs and the platform has arbitrary utility functions. Without loss of generality, we normalize $\xi(a) = a$.[12]

The platform has the ability to recommend the video in different levels based on its private state. In particular, the platform can design a finite[13] signal space Σ where each signal $\sigma \in \Sigma$ represents a recommendation level for the video (e.g., "recommended", "highly recommended", "best of today"). A signaling scheme $\pi : [m] \to \Delta(\Sigma)$ is a mapping from video (based on its state) into probability distributions over signals. We denote by $\pi(i, \sigma)$ the probability of sending signal $\sigma \in \Sigma$ at state i.

In this work, we consider the following repeated interaction between the platform and a population of users. All users share the same utility function $\rho(\cdot, \cdot)$ which is ***unknown*** to the platform. All players (including the platform) have the same prior λ. The setting proceeds for T rounds. For each round $t = 1, \ldots, T$:

1. The platform commits to a signal space Σ_t and a signaling scheme $\pi_t : [m] \to \Delta(\Sigma_t)$.
2. A video with state $\theta_t \sim \lambda$ is realized according to the prior λ and a signal σ_t is realized according to $\{\pi_t(\theta_t, \sigma)\}_{\sigma \in \Sigma_t}$.
3. Upon seeing the signal σ_t, user t updates his belief given prior λ and signaling scheme π_t. In particular, he forms a posterior distribution $\mu_t : \Sigma_t \to \Delta([m])$ that maps realized signal σ_t into probability distribution over state space $[m]$. We assume that users are ***Bayesian***, i.e., $\mu_t(\sigma_t, i) \triangleq \frac{\lambda(i)\pi_t(i,\sigma_t)}{\sum_{j \in [m]} \lambda(j)\pi_t(j,\sigma_t)}$.
4. With the posterior μ_t, user t chooses an action a_t that maximizes his expected utility, i.e., $a_t = \arg\max_{a \in \mathcal{A}} \mathbb{E}_{\theta \sim \mu_t}[\rho(\theta, a)]$.
5. The platform then derives the utility a_t.

Remark 1 (Bayesian-rational behavior). Upon seeing a signal realization, the user is able to form his Bayesian posterior belief, and then make his decision by maximizing the expected utility

[12] Namely, the platform gains 1 unit of profit if the user watches the video.

[13] For ease of presentation, our main context restricts the signal space to be finite. Our main result can be extended to continuous signal space.

based on the current belief. This Bayesian-rational behavior is following the common assumption adopted in Bayesian persuasion literature, and also other literature that includes signaling as a way to reveal product quality/characteristics information.[14]

REMARK 2 (COMMITMENT POWER). In the above interaction, the platform is assumed to have the commitment power and the designed signaling scheme is known to the user. In practice, the platform might lack this commitment power and may not be able to change her signaling scheme daily. However, we note that user may engage with the platform over a specific duration (i.e., a time cycle), e.g., staying on the platform for a while or coming back to the platform several times. Throughout this duration, the platform may stick to the same signaling scheme, then the user may be able to discern the underlying signaling scheme as well as his best response to this signaling scheme. Thus, a single round in above theoretical model can be equated to one practical time cycle.

Given a signaling scheme π, let $U(\pi)$ denote the platform's expected payoff. The goal of the platform is to design an online policy which constructs signaling schemes $\{\pi_t\}_{t\in[T]}$ to maximize her long-term expected utility $\sum_{t\in[T]} U(\pi_t)$.

2.2. Stackelberg Regret and Benchmark

We evaluate the performance of an online policy by its ***Stackelberg regret*** (Chen et al. 2020) against the ***optimal policy in hindsight***. The optimal policy in hindsight knows users' utility function $\rho(\cdot,\cdot)$, and maximizes the platform's long-term expected utility. Since users are all identical, the optimal policy in hindsight commits to the same signaling scheme π^* (see its characterization in program $\mathcal{P}^{\text{opt}}$ and Lemma 1) for every round $t\in[T]$.

DEFINITION 1. Given user's utility function ρ, let π^* be the optimal signaling scheme. The ***Stackelberg regret*** of online policy ALG is

$$\mathbf{REG}[\mathsf{ALG}] \triangleq \sum_{t\in[T]} U(\pi^*) - \mathbb{E}_{\pi_1,\dots,\pi_T}\left[\sum_{t\in[T]} U(\pi_t)\right]$$

where π_t is the signaling scheme committed by ALG in each round $t\in[T]$.

[14] For example, in the literature of Bayesian social learning with pricing, the customers are usually assumed to be able to update their beliefs in a Bayesian manner upon seeing certain new information about the product quality (Ifrach et al. 2019, Shin et al. 2023), and then make the purchase decision by maximizing the corresponding expected utility (Ifrach et al. 2019). Other literature includes signaling in queues (Debo et al. 2012, Lingenbrink and Iyer 2019).

We believe that the inclusion of Bayesian rational users inherently adds complexity to the problem, yielding rich insights and results. We view incorporating realistic and relevant behavioral biases as a next step in this research direction.

Different from the regret notation (e.g., external regret) in classic single-agent no regret learning literature (cf. Blum and Mansour 2007), the Stackelberg regret compares to the optimal policy in hindsight, where users have the opportunity to re-generate a different history by best-responding to the new signaling scheme π^*. In the remaining of the paper, we simplify the terminology Stackelberg regret as regret.

When users' utility function $\rho(\cdot,\cdot)$ is known to the platform, the optimal signaling scheme in hindsight. By the revelation principle (Kamenica and Gentzkow 2011), there always exists an optimal signaling scheme with binary signal space $\Sigma = \{0,1\} \equiv \mathcal{A}$ that corresponds to action recommendations. In particular, it can be solved by a linear program as follows,

$$\begin{aligned} \pi^* = \arg\max_{\pi} \ & \sum_{i\in[m]} \lambda(i)\pi(i,1) && \text{s.t.} \\ \text{(IC)} \quad & \sum_{i\in[m]} \left(\rho(i,1)-\rho(i,0)\right)\lambda(i)\pi(i,1) \geq 0 && \\ & \pi(i,1)+\pi(i,0) = 1 && i\in[m] \\ & \pi(i,1)\geq 0, \pi(i,0)\geq 0 && i\in[m] \end{aligned} \tag{$\mathcal{P}^{\mathtt{opt}}$}$$

Here constraint (IC) ensures obedience of the signaling schemes, i.e., taking action 1 is indeed user's optimal action given his posterior when action 1 is recommended.[15] For ease of presentation, with slight abuse of notation, we use $\pi^*(i) \triangleq \pi^*(i,1)$ and thus $\pi^*(i,0) \equiv 1-\pi^*(i)$. Additionally, we introduce one auxiliary variable that will be helpful for our analysis: $\delta(i) \triangleq \rho(i,1)-\rho(i,0)$ which represents how much the user prefers action 1 over action 0 given state i. To make the problem non-trivial, we make the following two assumptions on user's utility function throughout this paper.[16]

ASSUMPTION 1. *For user's utility function , there exists at least one state $i \in [m]$ such that $\delta(i)\lambda(i) > 0$.*

ASSUMPTION 2. *For user's utility function, $\sum_{i\in[m]} \delta(i)\lambda(i) < 0$.*

Program $\mathcal{P}^{\mathtt{opt}}$ can be interpreted as a fractional knapsack problem, where the budget is zero, and each state i corresponds to an item with value $\lambda(i)$ and (possibly negative) cost $\delta(i)\lambda(i)$. Thus, its optimal solution π^* has the following characterization.

LEMMA 1 **(See for example Renault et al. 2017)**. *The optimal signaling scheme π^* in hindsight is the optimal solution of linear program $\mathcal{P}^{opt}$. There exists a threshold state $i^\dagger \in [m]$ such that (a) for every state $i \neq i^\dagger$, $\pi^*(i) = \mathbb{1}[\delta(i) \geq \delta(i^\dagger)]$, and (b) $\pi^*(i^\dagger) = -\frac{\sum_{i\neq i^\dagger} \delta(i)\lambda(i)\pi^*(i)}{\delta(i^\dagger)\lambda(i^\dagger)}$.*

[15] In program $\mathcal{P}^{\mathtt{opt}}$, the obedient constraint for action 0 is omitted, since two programs have the same optimal solution. This is due to our assumption that the platform's utility of action 1 is higher than her utility of action 0.

[16] When Assumption 1 is violated, the problem becomes trivial since user takes action 0 regardless of the signaling scheme and thus any online policy achieves zero regret. Assumption 2 can be verified in round 1 by committing to a no-information-revealing signaling scheme, which induces regret at most 1. If $\sum_{i\in[m]} \delta(i)\lambda(i) \geq 0$, then committing to no-information-revealing signaling scheme in the remaining $T-1$ rounds attains optimal utility for the platform.

In words, Lemma 1 states that the signaling scheme π^* reveals whether the state is above or below[17] a threshold state $i^\dagger$, with possibly randomization at state $i^\dagger$.

The above Lemma 1 highlights the significance of both the cardinal value and the order of user preference differences $\{\delta(i)\}$ to characterize the optimal signaling scheme in hindsight. As we mentioned earlier, the user preference $\rho(\cdot,\cdot)$ is unknown to the platform. Consequently, the platform is unaware of both the cardinal value and the order of $\{\delta(i)\}$, and thus needs to learn these quantities by adaptively changing her signaling schemes. To address this intricate learning challenge, we initially study the scenario where only the cardinal value of $\{\delta(i)\}$ is unknown to the platform, while its order is known to the platform (see Section 3). Building on the insights from this scenario, we then discuss the more complex situation where the platform lacks knowledge of both the cardinal value and the order of $\{\delta(i)\}$ (see Section 4).

2.3. A Useful Subroutine for Checking Obedience

When user's utility function $\rho(\cdot,\cdot)$ is unknown, the standard revelation principle fails. As a consequence, restricting to binary signal space (e.g., {"recommended", "not recommended"}) is ***not*** without loss of generality. Nonetheless, as we formally show later, restricting to the subclass of signaling schemes with binary signal space does not hurt the optimal regret. We now formally define such signaling schemes as follows.

DEFINITION 2. A ***direct signaling scheme*** $\pi : [m] \to \Delta(\mathcal{A})$ is a mapping from state into probability distributions over action recommended to user.

With slight abuse of notation, for every direct signaling scheme π, we use $\pi(i) \triangleq \pi(i,1)$ and thus $\pi(i,0) \equiv 1 - \pi(i)$. When facing a direct signaling scheme π, user takes the action that maximizes his expected utility given his posterior. We say π is ***obedient*** if the user takes action 1 as long as action 1 is recommended by signaling scheme π. The proofs of Lemma 2, Lemma 3 and Lemma 4 are deferred to Section B.

LEMMA 2. *A direct signaling scheme π is obedient if and only if $\sum_{i\in[m]} \delta(i)\lambda(i)\pi(i) \geq 0$.*

Before we finish the preliminary section, we provide Procedure 1 as a useful subroutine which will be used in our online policies. Procedure 1 takes a direct signaling scheme as input, and determines whether this direct signaling scheme is obedient. Its correctness guarantee is given in Lemma 3 and the regret guarantee is given in Lemma 4.

[17] Throughout this paper, we say a state i is above (resp. below) state j if it satisfies that $\delta(i) \geq \delta(j)$ (resp. $\delta(i) < \delta(j)$).

Procedure 1: `CheckObed`(π)

Input: a direct signaling scheme π

Output: True/False – whether π is obedient; or `round-exhausted` if there is no round left

while *there are rounds remaining* **do**
  /* suppose now is round t */
  Commit to signaling scheme π towards user t.
  if $\sigma_t = 1$ *and* $a_t = 1$ **then**
    return True
  end
  else if $\sigma_t = 1$ *and* $a_t = 0$ **then**
    return False
  else if $\sigma_t = 0$ *and* $a_t = 1$ **then**
    return False
  move to next round, i.e., $t \leftarrow t+1$
end
return `round-exhausted`

LEMMA 3. *Given a direct signaling scheme π, Procedure 1 returns* True *only if π is obedient, and returns* False *only if π is not obedient.*

Note that Procedure 1 does not include the case for $\sigma_t = 0$ and $a_t = 0$ as it does not convey any information about whether the signaling scheme is obedient or not (see Lemma 2). Hence, Procedure 1 will keep running until not seeing the case $\sigma_t = 0$ and $a_t = 0$ or time rounds are exhausted. As long as Procedure 1 returns True\False, the platform knows for sure whether the signaling scheme is obedient or not.

LEMMA 4. *Given a direct signaling scheme π, the expected regret of Procedure 1 is at most* $\frac{U(\pi^*)}{\sum_i \lambda(i)\pi(i)} - \mathbb{1}[\pi \text{ is obedient}]$.

The intuition behind the Lemma 4 is as follows. Given a direct signaling scheme π, as long as its probability (i.e., the value $\sum_i \lambda(i)\pi(i)$) for recommending action 1 is a constant approximation to the optimal payoff $U(\pi^*)$, then the expected regret of Procedure 1 for checking its obedience can be upper bounded by this constant. However, if this probability is small compared to $U(\pi^*)$, then the incurred expected regret can be very large, regardless of the obedience of π or the value of $U(\pi^*)$.

3. Algorithm with Known Ordinal Preference

In this section, we provide our first result – an online policy with $O(\log\log T)$ regret when the order of $\{\delta(i)\}$ is known to the platform. The results of this section will be served as a building block for the algorithm design when the order of $\{\delta(i)\}$ is not known in advance.

Before diving into our result, let us highlight one of the main challenges in the design of a good online policy. In our problem, the platform's feedback is *limited* and *probabilistic*. Specifically, when signaling scheme π_t is used in round t and signal $\sigma_t \sim \pi_t(\theta_t)$ is realized, the platform only observes user's action under signal σ_t and learns her corresponding payoff, but nothing about her payoff under other signals. Meanwhile, this feedback is also probabilistic, since the realized signal σ_t depends on the realized state θ_t. Because of these two features of the feedback, some natural tasks towards learning user's utility may not be completed easily. Here are two illustrative examples.

Identifying the signs of $\{\delta(i)\}$. Recall that the optimal signaling scheme in hindsight π^* follows from a threshold signaling scheme – it recommends action 1 deterministically for all states above a threshold state $i^\dagger$, recommends action 1 randomly at threshold state, and recommends action 0 deterministically for all states below $i^\dagger$. Following the same logic, a natural attempt to design a good online policy is trying to identify the threshold state and the states that are above the threshold state. However, it is unclear on how to identify the threshold state. In fact, it is even challenging to identify the sign of $\delta(i)$ of a state i. To see this, ideally, identifying the sign of $\delta(i)$ needs to solicit the user's action when the user's posterior belief is concentrated on state i when a particular signal is realized. A signaling scheme π with $\pi(j) = \mathbb{1}[j = i], \forall j \in [m]$ can shape user's posterior belief to be concentrated on state i when a signal 1 is realized. However, since the feedback is limited, such signaling scheme cannot collect useful information whenever other signal is realized. Consequently, it bears a large regret if it happens to be the case when $\lambda(i)$ is small.

Determining if $U(\pi^*) \geq C$. Consider a problem instance with $m = 2$ states. Suppose the platform knows that $\delta(1) > 0$, and $\delta(2) < 0$. This implies that the threshold state $i^\dagger = 2$, and state 1 is above threshold state 2. Note that a good online policy should be able to approximately identify the value of $U(\pi^*)$. Now, suppose the platform only wants to determine whether $U(\pi^*) \geq C$. If platform can determine whether this following natural signaling scheme $\pi(1) = 1$ and $\pi(2) = {}^{(C - \lambda(1))}/_{\lambda(2)}$, is obedient or not, then the platform can determine whether $U(\pi^*) \geq C$.[18] However, since the feedback is probabilistic, it takes ${}^1/_C$ rounds (in expectation) to learn the obedience of π. Thus, even if $U(\pi^*)$ is small (i.e., $U(\pi^*) = o(1)$), by Lemma 4, the aforementioned attempt bears a superconstant regret as long as $C = o(U(\pi^*))$.

3.1. Towards $(\log\log T)$ Regret

Despite the aforementioned challenges, in this subsection, we present an algorithm that can have $O(\log\log T)$ regret guarantee. The details of our proposed algorithm `ConRP` are in Algorithm 2.

[18] Under this signaling scheme π, if π is obedient, then we have $U(\pi^*) \geq U(\pi) \geq C$, otherwise $U(\pi^*) < U(\pi) < C$.

Overview of the algorithm. In our online policy, the whole T rounds are divided into the *exploring* phase and the *exploiting* phase. The exploring phase has two subphases. The first subphase (i.e., *exploring phase I*) identifies a lower bound and an upper bound of $U(\pi^*)$, i.e., it identifies an obedient signaling scheme $\underline{\pi}$ with $\underline{\mathrm{U}} := U(\underline{\pi})$ such that $\underline{\mathrm{U}} \le U(\pi^*) \le 2\underline{\mathrm{U}}$. Note that once we narrow down the value of $U(\pi^*)$ to be in the interval $[\underline{\mathrm{U}}, 2\underline{\mathrm{U}}]$, with the obedient signaling scheme $\underline{\pi}$, we can ensure that expected regret to check the obedience of the signaling schemes in the later rounds is at most a constant, which addresses the second challenge (i.e., determining if $U(\pi^*) \ge C$) we just mentioned before. We will show that the expected cumulative regret in exploring phase I is $O(1)$. The second subphase (i.e., *exploring phase II*) identifies a signaling scheme $\pi^\dagger$ whose per-round expected regret is $1/T$, and we will show that its expected cumulative regret is at most $O(\log\log T)$. The identified signaling scheme $\pi^\dagger$ from the exploring phase II is used in the remaining rounds considered as the exploiting phase, which induces $O(1)$ expected cumulative regret. See ConRP for a formal description.

A subclass of direct signaling schemes $\{\pi^{(u)}\}$. Our online policy will repeatedly consider a subclass of direct signaling schemes. Recall program $\mathcal{P}^{\mathsf{opt}}$ indicates that the optimal signaling scheme in hindsight π^* can be thought as the optimal solution of a fractional knapsack problem, where each state i corresponds to an item with value $\lambda(i)$ and cost $\omega(i)$. This observation implies that there must exist a total order r^* over all states with respect to their true bang-per-buck $\delta(i) = \lambda(i)/\omega(i)$. Given an arbitrary number $u \in [0,1]$, we define $\pi^{(u)}$ to be the direct signaling scheme as follows: there exists a threshold state $i^\dagger$ such that (a) for every state $i \neq i^\dagger$, $\pi^{(u)}(i) = \mathbb{1}[\delta(i) > \delta(i^\dagger)]$ and (b) $\pi^{(u)}(i^\dagger) = \frac{u - \sum_{i \neq i^\dagger} \lambda(i)\pi^{(u)}(i)}{\lambda(i^\dagger)}$ (recall that since the platform knows the order of user's preference difference $\{\delta(i0\}$, this signaling scheme is well-defined). As a sanity check, observe that the signaling scheme $\pi^{(U(\pi^*))}$ is exactly the optimal signaling scheme in hindsight π^*. By construction, it is also guaranteed that $\sum_{i\in[m]} \lambda(i)\pi^{(u)}(i) = u$. We note that by focusing the signaling scheme $\pi^{(u)}$, we bypass the challenge on identifying the value, order or even signs of $\{\delta(i)\}$. Indeed, for general problem instances, our ConRP does not explicitly learn those quantities, nor they can be inferred from the outcome of ConRP.

Remark 3. In the above ConRP, the first subphase (i.e., exploring phase I) is used to identify a lower bound and an upper bound of the optimal payoff $U(\pi^*)$ of the platform. This step is crucial for us to establish the $O(\log\log T)$ regret. In Section 6, we present simulation studies showing that without this step, the algorithm may perform very bad. Moreover, this subphase is also necessary and used to identify an interior point in designing our second main algorithm, i.e., Algorithm 3.

We are now ready to describe the main result of this section.

Theorem 1. *The expected regret of ConRP is at most $O(\log\log T)$.*

Algorithm 2: Conservative Recommendation Policy (ConRP)

Input: number of rounds T, number of states m, prior distribution λ

/* exploring phase I - identify $\underline{U}$ such that $\underline{U} \le U(\pi^*) \le 2\underline{U}$ */

Initialize $\underline{U} \leftarrow \frac{1}{2}$

while $\mathit{CheckObed}\left(\pi^{(\underline{U})}\right) = \mathsf{False}$ **do**

  $\underline{U} \leftarrow \frac{\underline{U}}{2}$

end

/* exploring phase II - identify a signaling scheme $\pi^\dagger$ such that $U(\pi^\dagger) \ge U(\pi^*) - \frac{1}{T}$ */

Initialize $R \leftarrow 2\underline{U}$, $L \leftarrow \underline{U}$, $\delta \leftarrow 1$

while $R - L \ge \frac{1}{T}$ **do**

  $\varepsilon \leftarrow \frac{\delta}{2}$, $S \leftarrow \lfloor \frac{R-L}{\varepsilon L} \rfloor$, $\ell \leftarrow 1$.

  while $\mathit{CheckObed}\left(\pi^{(L+\ell\varepsilon L)}\right) = \mathsf{False}$ **do**

    $R \leftarrow L + \ell\varepsilon L$, $L \leftarrow L + (\ell-1)\varepsilon L$, $\delta \leftarrow \varepsilon^2$, $\ell \leftarrow \ell + 1$.

  end

end

Set $\pi^\dagger \leftarrow \pi^{(L)}$.

/* exploiting phase */

Use signaling scheme $\pi^\dagger$ for all remaining rounds.

Proof. We analyze the expected regret in exploring phase I, exploring phase II, and exploiting phase separately. We first assume that ConRP finishes exploring phase I and II before T rounds are exhausted. Similar argument follows for the other case where exploring phase I or exploring phase II is completed due to the exhaustion of rounds.

Exploring phase I. Let $K = -\lceil \log(U(\pi^*)) \rceil$. By definition, $\mathsf{CheckObed}(\pi^{(2^{-k})}) = \mathsf{False}$ for $k \in [K-1]$, and $\mathsf{CheckObed}(\pi^{(2^{-K})}) = \mathsf{True}$. Thus, in the end of exploring phase I, $\underline{U}$ is 2^{-K}, and there are K iterations in the while loop. For each iteration $k \in [K]$, $\mathsf{CheckObed}(\pi^{(2^{-k})})$ is called once. By Lemma 4, the total expected regret is

$$\sum_{k\in[K]} \frac{U(\pi^*)}{\sum_{i\in[m]} \lambda(i)\pi^{(2^{-k})}(i)} \overset{(a)}{\le} \sum_{k\in[K]} \frac{2^{-(K-1)}}{2^{-k}} = \sum_{k\in[K]} 2^{-(K-k-1)} = O(1)$$

where the denominator in the right-hand side of inequality (a) is due to the construction of $\pi^{(2^{-k})}$.

Exploring phase II. By construction, there are $O(\log\log T)$ iterations in the while loop. Thus, it is sufficient to show the expected regret in each iteration is $O(1)$.

In each iteration k, let $\ell^\dagger \in [S]$ be the smallest index that the signaling scheme $\pi^{(L+\ell^\dagger\varepsilon L)}$ is not obedient. The expected regret in iteration k is at most

$$\sum_{\ell=1}^{\ell^\dagger-1} \left(\frac{U(\pi^*)}{\sum_{i\in[m]} \lambda(i)\pi^{(L+\ell\varepsilon L)}(i)} - 1 \right) + \frac{U(\pi^*)}{\sum_{i\in[m]} \lambda(i)\pi^{(L+\ell^\dagger\varepsilon L)}(i)}$$

$$\overset{(a)}{=}\sum_{\ell=1}^{\ell^\dagger-1}\left(\frac{U(\pi^*)}{L+\ell\varepsilon L}-1\right)+\frac{U(\pi^*)}{L+\ell^\dagger\varepsilon L}\overset{(b)}{\leq}\sum_{\ell=1}^{\ell^\dagger-1}\left(\frac{R}{L}-1\right)+\frac{R}{L}\overset{(c)}{\leq}(S-1)\frac{R-L}{L}+2\overset{(d)}{\leq}\frac{(R-L)^2}{\varepsilon L^2}+2$$

where equality (a) holds due to the construction of $\pi^{(L+\ell\varepsilon L)}$ and $\pi^{(L+\ell^\dagger\varepsilon L)}$; inequality (b) holds since $U(\pi^*)\leq R$; inequality (c) holds since $\ell^\dagger\leq S$ and $R\leq 2L$; and inequality (d) holds since $S=\lfloor\frac{R-L}{\varepsilon L}\rfloor$.

We finish this part by showing $R-L\leq\sqrt{2\varepsilon}L$ by induction. Let $L^{(k)},R^{(k)},\delta^{(k)}$ and $\varepsilon^{(k)}$ be the value of L,R,δ,ε in each iteration k. The claim is satisfied for iteration $k=1$, since $R^{(1)}-L^{(1)}=2\underline{\mathrm{U}}-\underline{\mathrm{U}}=L^{(1)}$ and $\varepsilon^{(1)}=1/2$. Suppose the claim holds for iteration $k-1$. Now, for iteration k, we know that $R^{(k)}-L^{(k)}=\varepsilon^{(k-1)}L^{(k-1)}\leq\varepsilon^{(k-1)}L^{(k)}=\sqrt{\delta^{(k)}}L^{(k)}=\sqrt{2\varepsilon^{(k)}}L^{(k)}$, which finishes the induction.

Exploiting phase. In this phase, we know that $\pi^\dagger$ is obedient and $U(\pi^\dagger)\geq U(\pi^*)-1/T$, which concludes the proof. □

4. Algorithm with Unknown Ordinal Preference

In previous section, we have discussed how to design an algorithm with $O(\log\log T)$ regret when the user's ordinal preference, i.e., the order of $\{\delta(i)\}$ is known to the platform, but the cardinal values of the preference are unknown to the platform. In this section, we relax platform's knowledge of the order of $\{\delta(i)\}$. In particular, we first present the algorithm design for a general class of user's utilities, we then discuss a more challenging setting where platform has no knowledge about user's preference. In both scenarios, we show that there exists an algorithm with $O(\log\log T)$ regret.

4.1. Application: Posterior-mean-dependent User Preference

In this subsection, we show that for a general class of user's preference which is affine with respect to the state, `ConRP` is able to achieve $O(\log\log T)$ regret, and this regret bound does not depend on the number of possible states.

Notice that an affine state-dependent preference could be represented as $\rho(\theta,a)=\rho_1(a)\theta+\rho_2(a)$ where functions $\rho_1,\rho_2:\mathcal{A}\to\mathbb{R}$ are unknown to the platform. With such affine state-dependent preference, it is easy to see that the user's optimal action, i.e., whether to watch the video or not, depends only on the expected state of the user's posterior belief over the underlying video states. This posterior-mean dependency of the optimal action is a fundamental setting in Bayesian persuasion studied in many works (Candogan 2022, Arieli et al. 2023, Kolotilin 2018, Gentzkow and Kamenica 2016, Candogan and Strack 2021, Kolotilin et al. 2017).

To see how `ConRP` could be adapted to solve the this setting, notice that under affine state-dependent preference, the user's preference difference is essentially $\delta(i)=(\rho_1(1)-\rho_1(0))i+\rho_2(1)-\rho_2(0)$. This shows that there exists at most two possible orders of the user's preference difference $\{\delta(i)\}$, depending on the sign of $\rho_1(1)-\rho_1(0)$. Thus, one can just run `ConRP` over these two possible orders of $\{\delta(i)\}$ in

a round-robin manner and use the payoff of any identified obedient signaling scheme to prune out the incorrect order. The main result in this subsection is summarized as follows:

PROPOSITION 1. *For the a ne state-dependent preference, the expected regret of* `ConRP` *is* $O(\log\log T)$*, this regret also holds even when the state space* Θ *is continuous.*

Lastly, we observe that the above results can also be extended to scenarios wherein the user's preferences are depending on a potentially non-linear transformation of the state, denoted as $f(\theta)$, in an arbitrary manner. Fundamental to our conclusions is the observation that the order of user's preference difference $\{\delta(i)\}$ is not changing even with such non-linear transformation.

4.2. Application: User with Unknown Ordinal Preference

In this subsection, we show that when the true order of user's preference differences $\{\delta(i)\}$ is unknown to the platform, a regret $O(m2^{m-1}\cdot\log\log T)$ is achievable by numerating all possible orders.

PROPOSITION 2. *When the platform has no knowledge of user's preference, the expected regret of a modified version (see Algorithm 4 in Section C.1) of* `ConRP` *is* $O(m2^{m-1}\cdot\log\log T)$.

In below we briefly discuss how the `ConRP` could be adapted to obtain the above regret bound. The formal proof of Proposition 2 is provided in Section C.1. Similar to `ConRP`, the modified algorithm (Algorithm 4) also relies on a subclass of direct signaling schemes $\pi^{(r,u)}$, but with an additional parameter r to represent a possible (total) order of the user's preference differences $\{\delta(i)\}$. In other words, for every possible order r of the user's preference differences $\{\delta(i)\}$, one can identify a direct signaling scheme $\pi^{(r,u)}$ such that the expected payoff to the platform will exactly equal to u if this signaling scheme $\pi^{(r,u)}$ is *obedient*. In the modified algorithm, whenever the algorithm identifies an obedient signaling scheme $\pi^{(r,u)}$ (see Line 4 and 15 in Algorithm 4), it then naturally gives a lower bound of the platform's expected payoff of the optimal signaling scheme, namely, $U(\pi^*)\geq u$. Then one can use the payoff (i.e., u) of such obedient signaling scheme to further prune out other signaling schemes that are constructed with different order of $\{\delta(i)\}$ but are either non-obedient or have payoff less than u. We would like to note that without pruning, the regret of the algorithm might have linear dependency on the time horizon T.

In more detail, recall that when user's utility function is unknown, the bang-per-buck as well as the true total order r^* are unknown to the platform. In the exploring phase of Algorithm 4, our online policy maintains a subset $\mathcal{P}$ of total orders over $[m]$ that contains the optimal order r^*. In particular, the algorithm initializes the set $\mathcal{P}$ such that it contains all possible orders, and each order r in the set $\mathcal{P}$ specifies a state $i^\dagger\in[m]$, a subset of states that are below the state $i^\dagger$, and a subset of states that are above the state $i^\dagger$. Given an arbitrary order r, we define $\pi^{(r,u)}$ to be the direct signaling

scheme as follows: let state $i^\dagger$ be the state associated with this order r, then (a) for every state $i \neq i^\dagger$, $\pi^{(r,u)}(i) = \mathbb{1}[r(i) > r(i^\dagger)]$,[19] and (b) $\pi^{(r,u)}(i^\dagger) = \frac{u - \sum_{i \neq i^\dagger} \lambda(i)\pi^{(r,u)}(i)}{\lambda(i^\dagger)}$. As a sanity check, observe that the signaling scheme $\pi^{(r^*, U(\pi^*))}$ is exactly the optimal signaling scheme in hindsight π^*, and it is also guaranteed that $\sum_{i \in [m]} \lambda(i)\pi^{(r,u)}(i) = u$ by construction.

Note that even though the number of all possible total order could be as large as $O(m!)$, we can have a more succinct representation on user's ordinal preference. To see this, note that each possible order r can first specify a state $i^\dagger$, a subset of states that are below the state $i^\dagger$, and remaining states that are above the state $i^\dagger$. Then, in total, there are at most $O(m2^{m-2})$ such possible orders.

REMARK 4. In both exploring phase I and II, Algorithm 4 checks whether there exists $r \in \mathcal{P}$ such that $\mathsf{CheckObed}(\pi^{(r,u)}) = \mathsf{True}$ for some u. Our regret bound has an (2^m) dependence due to brute-force searching over r. We would like to note that (i) in many practical applications, the number of states m is small or even constant, and thus our main focus in this section is the optimal dependence on the number of rounds T, and (ii) when N $(\leq m2^{m-1})$ identical problem instances are allowed to run in parallel, the regret dependence on m becomes $m2^{m-1}/N$.

5. LP-based Algorithm

In this section, we provide our second result – a linear program-based recommendation policy (LP-RP) with $O(\mathsf{poly}(m \log T))$ regret when the user's preferences (including both the cardinal preference and ordinal preference) are unknown to the platform. The main result in this section is as follows:

THEOREM 2. *The expected regret of LP-RP is at most* $O\left(m^6 \log^{O(1)}(mT)\right)$.

The proposed LP-RP uses a subroutine MembershipLP – an algorithm (e.g., Lee et al. 2018) to solve linear program with membership oracle access.[20] We first formally introduce the linear program optimization with membership oracle access, and discuss its connection to our online Bayesian recommendation problem. Then we provide the formal description and the explanation of LP-RP where we also present the proof of Theorem 2.

Linear program optimization with membership oracle access. Optimizing a linear function $f(\cdot)$ within an unknown convex set H has been studied extensively in the literature. There are two standard oracle assumptions: *membership oracle* and *separation oracle*. A membership oracle returns whether a queried point y is contained in convex set H. In contrast, a separation oracle not only returns whether a queried point y is contained in convex set H, but also returns a hyperplane that separates y from H if $y \notin H$.

[19] Given an order r, we denote $r(i)$ by the rank of state i.

[20] LP-RP uses MembershipLP as a blackbox. Namely, it can be replaced by other algorithms for linear program with membership oracle access.

Recall that in our problem, the optimal signaling scheme in hindsight π^* is the optimal solution of linear program $\mathcal{P}^{\mathsf{opt}}$. From the platform's perspective, the only unknown component in this program is $\{\delta(i)\lambda(i)\}$ in the IC constraint. Nonetheless, using Procedure 1, the platform can determine the obedience (i.e., whether the IC constraint is satisfied) of any direct signaling scheme. In other words, Procedure 1 works like a membership oracle for the convex set which contains all obedient signaling schemes. Thus, finding the optimal signaling scheme π^* can be formulated as optimizing a linear program with membership oracle access. In particular, we leverage the algorithm introduced in Lee et al. (2018) with the following guarantee.

THEOREM 3 (**Lee et al. 2018**). *For any linear function $f(\cdot)$, and convex set $H \subseteq \mathbb{R}^m$, given an interior point $x^{(0)}$, a lower bound r, an upper bound R such that $\mathbf{B}_2(x^{(0)}, r) \subseteq H \subseteq \mathbf{B}_2(x^{(0)}, R)$,*[21] *and given a membership oracle, there exists an algorithm* MembershipLP *that finds an ϵ-approximate optimal solution for $f(\cdot)$ in H with probability $1-\delta$, using $O(m^2 \log^{O(1)}(mR/\epsilon\delta r))$ queries to the oracle.*

We note that when only membership oracle is given, the interior point $x^{(0)}$ as well as lower bound r, and upper bound R such that $\mathbf{B}_2(x^{(0)}, r) \subseteq H \subseteq \mathbf{B}_2(x^{(0)}, R)$ is necessary for any algorithms. Otherwise, there is an information-theoretic barrier (see Grötschel et al. 2012).

5.1. Towards $O(\mathsf{poly}(m \log T))$ Regret

Before we describe our algorithm, let us highlight two major hurdles in applying the membership oracle approach to solve our Bayesian recommendation problem.

1. Though Theorem 3 upper bounds the total number of queries to the membership oracle (a.k.a., Procedure 1), as illustrated in our second example presented in Section 3, the regret from one execution of Procedure 1 may be superconstant.
2. Theorem 3 requires an interior point x as well as a lower bound radius r, and an upper bound radius R such that $\mathbf{B}_2(x, r) \subseteq H \subseteq \mathbf{B}_2(x, R)$, and the number of queries depends on the value of r and R. However in our problem, the interior point is *not* given explicitly. How to find a proper interior point x with non-trivial lower bound radius r (without incurring too much regret) is not obvious in our problem.

Overview of the algorithm. We now sketch LP-RP. The detailed proof is provided in Section D in appendix. The high-level idea of this algorithm is to use MembershipLP as a subroutine to identify a signaling scheme $\pi^\dagger$ whose per-round expected regret is $1/T$. In more detail, LP-RP divides the whole T rounds into an *exploring phase* and an *exploiting phase*. In exploring phase, we use MembershipLP to identify a persuasive signaling scheme $\pi^\dagger$, and exploiting phase uses $\pi^\dagger$ until the rounds are

[21] $\mathbf{B}_2(x^{(0)}, r)$ is the ball of radius r centered at $x^{(0)}$.

exhausted. As mentioned in the above two hurdles, to use MembershipLP, we need to ensure that each query (i.e., a signaling scheme) to the MembershipLP cannot incur too much regret, i.e., Procedure 1 for checking the persuasiveness of a queried signaling scheme cannot be large; and we need to find a proper interior point with non-trivial lower bound radius. To achieve this, there are three subphases in the exploring phase:

Exploring phase I – Lowerbounding $U(\pi^*)$: Similar to ConRP, the first step of LP-RP is to identify a lower bound and an upper bound of $U(\pi^*)$. But different from ConRP, here, we identify a set $\hat{\mathcal{S}}$ of persuasive direct signaling schemes such that for every signaling scheme $\pi^I \in \hat{\mathcal{S}}$, it has following two properties: (i) it has the same payoff $\underline{\mathrm{U}}$ with other signaling schemes in set $\hat{\mathcal{S}}$, i.e., $\underline{\mathrm{U}} \equiv U(\pi^I), \forall \pi^I \in \hat{\mathcal{S}}$, and $\underline{\mathrm{U}}$ is relatively good, i.e., $\underline{\mathrm{U}} \geq \frac{U(\pi^*)}{m^2}$; (ii) signaling scheme π^I has a specific structure where it has non-zero probability for recommending action 1 on at most two states.

LEMMA 5 **(informal)**. *When exploring phase I terminates, $\underline{\mathrm{U}} \geq \frac{U(\pi^*)}{m^2}$ and $\hat{\mathcal{S}}$ is not empty.*

At a high level, the property (i) implies that $\underline{\mathrm{U}} \leq U(\pi^*) \leq m^2\,\underline{\mathrm{U}}$, which can guarantee us whenever we use Procedure 1 (as a membership oracle) to check the persuasiveness of a direct signaling scheme in the later rounds, the expected regret is at most $O(m^2)$. The property (ii) can guarantee us to find an interior point with non-trivial lower bound radius r in the later subphase.

Exploring phase II – Excluding Degenerated States: To find the interior point for the program ($\mathcal{P}^{\mathsf{opt}}$), however, we first note that it is possible the convex set in the program ($\mathcal{P}^{\mathsf{opt}}$) is degenerated and thus no interior point exists. Nonetheless, those degenerated dimensions (i.e., states) must contribute little to $U(\pi^*)$. Thus, in this exploring phase, we use the signaling schemes in $\hat{\mathcal{S}}$ obtained in Exploring phase I to exclude those states and obtain a set $\tilde{\Theta} \subseteq [m]$ that contains all relatively good states (i.e., the state whose ω cannot be too negative).

LEMMA 6. *When exploring phase II terminates,*

- *for each state $i \in \tilde{\Theta}$:* $\delta(i)\lambda(i) \geq -mT \cdot \max_{j\in[m]} \delta(j)\lambda(j)$;
- *for each state $i \notin \tilde{\Theta}$:* $\delta(i)\lambda(i) < -\frac{mT}{3} \cdot \max_{j\in[m]} \delta(j)\lambda(j)$.

With the obtained $\tilde{\Theta}$ at hand, we show that there exists a persuasive direct signaling scheme $\tilde{\pi}^{(0)}$ such that $\frac{1}{8m^2T} \leq \tilde{\pi}^{(0)}(i) \leq 1 - \frac{1}{8m^2T}$ for every $i \in \tilde{\Theta}$, and $\tilde{\pi}^{(0)}(i) = 0$ for every $i \notin \tilde{\Theta}$. Furthermore, signaling scheme $\tilde{\pi}^{(0)}$ is an interior point[22] of following linear program.

$$
\begin{aligned}
\max_{\pi:\pi(i)=0\ \forall i\notin\tilde{\Theta}} \quad & \sum_{i\in\tilde{\Theta}} \lambda(i)\pi(i) && \text{s.t.} \\
& \sum_{i\in\tilde{\Theta}} \delta(i)\lambda(i)\pi(i) \geq 0 \\
& \sum_{i\in\tilde{\Theta}} \lambda(i)\pi(i) \geq \frac{1}{16}\underline{\mathrm{U}} \\
& \pi(i) \in [0,1] && \forall i \in \tilde{\Theta}
\end{aligned}
\qquad (\mathcal{P}^{\mathsf{opt}}_{\tilde{\Theta}})
$$

[22] Here we mean $\tilde{\pi}^{(0)}$ is an interior point of convex set in program $\mathcal{P}^{\mathsf{opt}}_{\tilde{\Theta}}$ when we restrict to states in $\tilde{\Theta}$.

Because of Lemma 6, the optimal objective value of program $\mathcal{P}^{\mathtt{opt}}_{\tilde{\Theta}}$ is close to $U(\pi^*)$.

LEMMA 7. *Let $\pi^\ddagger$ be the optimal solution in program $\mathcal{P}^{\mathtt{opt}}_{\tilde{\Theta}}$, i.e., $\pi^\ddagger = \arg\max \mathcal{P}^{\mathtt{opt}}_{\tilde{\Theta}}$. Then $U(\pi^\ddagger) \geq U(\pi^*) - O(\frac{1}{T})$.*

Exploring phase III – Executing MembershipLP *with Interior Point Candidates*: In this phase, we identify a direct signaling scheme $\pi^\dagger$ whose per-round expected regret is $O(\frac{1}{T})$ (i.e., $U(\pi^\dagger) \geq U(\pi^*) - O(\frac{1}{T})$) with probability $1 - \frac{1}{T}$. To do this, we solve a $\frac{1}{T}$-approximate solution in program $\mathcal{P}^{\mathtt{opt}}_{\tilde{\Theta}}$ by using MembershipLP as a subroutine. However, we note that we cannot directly identify the interior point $\tilde{\pi}^{(0)}$ to the program $\mathcal{P}^{\mathtt{opt}}_{\tilde{\Theta}}$ mentioned in exploring phase II. Instead, we introduce a specific modification for signaling schemes in $\hat{\mathcal{S}}$ – for every signaling scheme $\pi^I \in \hat{\mathcal{S}}$, its modification $\pi^{(0)}$ is an interior point candidate. In particular, because of Lemma 6, there exists a signaling scheme $\pi^I \in \hat{\mathcal{S}}$ whose modification $\pi^{(0)}$ is indeed an interior point $\tilde{\pi}^{(0)}$.

LEMMA 8 **(informal)**. *There exists a signaling scheme $\pi^I \in \hat{\mathcal{S}}$ such that its modification $\pi^{(0)}$ is an interior point of program $\mathcal{P}^{\mathtt{opt}}_{\tilde{\Theta}}$. In particular, let $r = \frac{1}{16m^2T}$, then $\mathbf{B}_2(\pi^{(0)}, r) \subseteq H(\mathcal{P}^{\mathtt{opt}}_{\tilde{\Theta}})$.*

Finally, we run MembershipLP based on every interior point candidate $\pi^{(0)}$ (and in the end, we pick the best solution as $\pi^\dagger$), where we set the interior point $x^{(0)} \leftarrow \pi^{(0)}$, lower-bound radius $r \leftarrow \frac{1}{16m^2T}$, upper-bound radius $R \leftarrow \sqrt{m}$, precision $\epsilon \leftarrow \frac{1}{T}$ and success probability $\delta \leftarrow \frac{1}{T}$.[23]

We present formal description of LP-RP below.

In this algorithm, three specific subclasses of direct signaling schemes $\{\pi^I\}$, $\{\pi^{\mathtt{II}}\}$, and $\{\pi^{(0)}\}$ are used, whose constructions are as follows. We note that the aforementioned signaling scheme subset $\hat{\mathcal{S}}$ in the algorithm overview is not explicitly defined in LP-RP. Its formal definition is $\hat{\mathcal{S}} \triangleq \{\pi^I \text{ induced by} (i^\ddagger, j^\ddagger) \in \mathcal{S}\}$.

- Given $(i^\ddagger, j^\ddagger, \underline{\mathrm{U}})$, we let π^I denote a direct signaling scheme with $\pi^I(i^\ddagger) = 1$, $\pi^I(j^\ddagger) = \frac{\underline{\mathrm{U}}}{\lambda(j^\ddagger)}$ if $j^\ddagger \neq i^\ddagger$, and $\pi^I(i) = 0$ for every $i \notin \{i^\ddagger, j^\ddagger\}$.
- Given $(i^\ddagger, j^\ddagger, \underline{\mathrm{U}}, i)$, we let $\pi^{\mathtt{II}}$ denote a direct signaling scheme with $\pi^{\mathtt{II}}(i^\ddagger) = 1$, $\pi^{\mathtt{II}}(j^\ddagger) = \frac{\underline{\mathrm{U}}}{2\lambda(j^\ddagger)}$ if $j^\ddagger \neq i^\ddagger$, $\pi^{\mathtt{II}}(i) = \frac{3}{2mT}$, and $\pi^{\mathtt{II}}(j) = 0$ for every $j \notin \{i^\ddagger, j^\ddagger, i\}$.
- Given $(i^\ddagger, j^\ddagger, \underline{\mathrm{U}}, \tilde{\Theta})$, we let $\pi^{(0)}$ denote a direct signaling scheme with $\pi^{(0)}(i^\ddagger) = \frac{1}{2} + \frac{1}{16m^2T}$, $\pi^{(0)}(j^\ddagger) = \frac{\underline{\mathrm{U}}}{8\lambda(j^\ddagger)}$ if $j^\ddagger \neq i^\ddagger$, $\pi^{(0)}(i) = \frac{1}{8m^2T}$ for every $i \in \tilde{\Theta} \backslash \{i^\ddagger, j^\ddagger\}$, and $\pi^{(0)}(i) = 0$ for every $i \notin \tilde{\Theta}$.

[23] When an incorrect interior point is given, MembershipLP terminates with a suboptimal solution. The number of queries to the oracle is the same as the one in Theorem 3.

Algorithm 3: LP-based Recommendation Policy (LP-RP)

Input: number of rounds T, number of states m, prior distribution λ, and linear program solver MembershipLP (Lee et al. 2018) with membership oracle access

/* exploring phase I */

Initialize $\underline{\mathrm{U}} \leftarrow \frac{1}{2}$

while $\underline{U} \geq \frac{1}{m^2 T}$ **do**

  if *there exists* $(i^\ddagger, j^\ddagger) \in [m] \times [m]$ *such that* $\texttt{CheckObed}(\pi^I) = \mathsf{True}$ **then**

    $\mathcal{S} \leftarrow \left\{(i^\ddagger, j^\ddagger) \in [m] \times [m] : \mathsf{CheckObed}\left(\pi^I\right) = \mathsf{True}\right\}$

    break

  end

  else

    $\underline{\mathrm{U}} \leftarrow \frac{\underline{\mathrm{U}}}{2}$

end

if $\mathcal{S} = \emptyset$ **then**

  Set $\pi^\dagger : [m] \to \{0\}$ to be the signaling scheme which reveals no information.

  Move to exploiting phase.

end

/* exploring phase II */

Initialize $\tilde{\Theta} \leftarrow \emptyset$

for *each state pair* $(i^\ddagger, j^\ddagger) \in \mathcal{S}$ **do**

  $\tilde{\Theta} \leftarrow \tilde{\Theta} \cup \left\{i \in [m] : i = i^\ddagger \text{ or } i = j^\ddagger \text{ or } \mathsf{CheckObed}\left(\pi^{\mathrm{II}}\right) = \mathsf{True}\right\}$

end

/* exploring phase III */

for *for each pair* $(i^\ddagger, j^\ddagger) \in \mathcal{S}$ **do**

  Solve $\pi^{(i^\ddagger, j^\ddagger)}$ by linear program solver MembershipLP for program $\mathcal{P}^{\mathsf{opt}}_{\tilde{\Theta}}$: set the interior point $x^{(0)} \leftarrow \pi^{(0)}$, lower-bound radius $r \leftarrow \frac{1}{16 m^2 T}$, upper-bound radius $R \leftarrow \sqrt{m}$, precision $\epsilon \leftarrow \frac{1}{T}$ and success probability $\delta \leftarrow \frac{1}{T}$.

end

Set $\pi^\dagger$ be best $\pi^{(i^\ddagger, j^\ddagger)}$ $\left(\text{i.e., maximizing } U\left(^{(i^\ddagger, j^\ddagger)}\right)\right)$ for all $(i^\ddagger, j^\ddagger) \in \mathcal{S}$

/* exploiting phase */

Use signaling scheme $\pi^\dagger$ for all remaining rounds.

6. Simulations

In this section, we provide some insights from numerical experiments that test the empirical performance of our proposed algorithm and highlight some of its salient features. The main goal of this section is to evaluate the performance of our main algorithm (i.e., ConRP). Thus, we will focus on the case where the user's ordinal preference is known to the platform, but the cardinal preference remains unknown to the platform. We simulate an instance of the Bayesian recommendation problem with the number of states $m = 50$ and time horizon $T = 2000$, where the prior distribution λ is generated

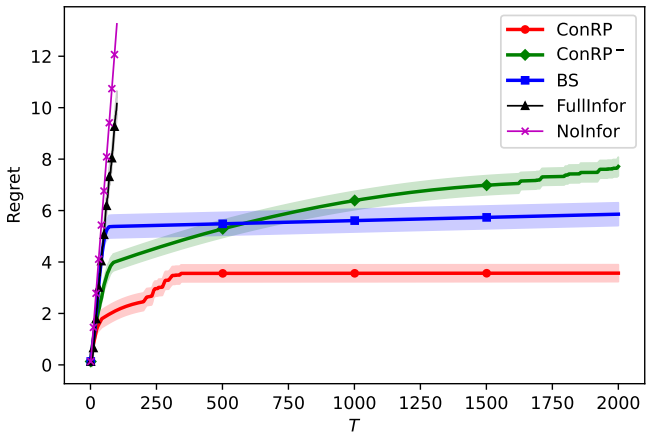

Figure 1 The above figure plots the regret growth with T for various algorithms on a randomly generated instance with $m = 50$, and platform's optimal payoff is $U(\pi^*) = 0.1326$.

randomly in the space $\Delta([m])$, and the user's preference differences $\{\delta(i)\}$ are generated randomly from $\mathsf{Unif}[-2, 2]$.[24] And, we compute the average regret based on 50 independent simulations from this randomly generated instance. In Figure 1, we report performance of the following algorithms:

1. `ConRP`: This is our proposed algorithm (Algorithm 2 in Section 3), which attains optimal $O(\log \log T)$ regret.
2. `ConRP`$^-$: This is a modification of our `ConRP` with no exploring phase I (i.e., the binary-search steps) to lower bound the platform's optimal payoff.
3. `BS`: This is a binary-search algorithm, notice that since the user's ordinal preference is known to the platform, identifying the optimal signaling scheme is equivalent to identify the platform's optimal payoff. Thus, this algorithm implements a binary-search to identify the value of platform's optimal payoff where the algorithm can utilize the user's response to determine whether the payoff of the current signaling scheme is lower/higher than the optimal payoff.
4. `FullInfor`: This is a naive algorithm that keeps using full-information signaling scheme. For the above generated random instance, the expected one-round payoff to the platform is 0.0304.
5. `NoInfor`: This is a naive algorithm that keeps using no-information signaling scheme. For the above generated random instance, the expected one-round payoff to the platform is 0.

The last two baseline algorithm `FullInfor` and `NoInfor` are used to demonstrate the usefulness of signalling in our Bayesian recommendation problem. It is expected to see that the performance of this two algorithms grow linearly with the time horizon T (in Figure 1, we only plot out the performance of these two algorithms for $T \leq 100$ due to the limited margin of the presented figure). As we can see in Figure 1, the superior performance of our proposed `ConRP` demonstrates its

[24] Notice that, in our setting, the user's preference differences are the sufficient quantities to summarize user's behavior.

effectiveness compared to other baseline algorithms. In particular, we observe that our ConRP performs significantly better than the binary-search algorithm BS, this is due to the asymmetric property of the payoff of the platform's signaling scheme (i.e., the payoff $U(\pi)$ of a particular signaling scheme π is $U(\pi) = \sum_i \lambda(i)\pi(i) \cdot \mathbb{1}[\pi \text{ is obedient}]$). Our ConRP also performs significantly better than the algorithm ConRP$^-$, and this significant improvements in performance is due to the additional exploring phase I to identify a lower bound of the platform's optimal payoff.

7. $\Omega(\log\log T)$ Regret Lower Bound

We now show a tight lower bound of $\Omega(\log\log T)$ regret of any online policy, even when the number of states is 2. Here we allow the online policy to be randomized (i.e., can commit to different signaling schemes at random) and have non-binary (but finite) signal spaces (i.e., can have multiple recommendation levels).

THEOREM 4. *No online policy can achieve an expected regret better than $\Omega(\log\log T)$, even for the family of binary-state problem instances.*

We note that for every binary-state instance, the user's utility can be represented as an affine function over the state space. Thus, the above regret lower bound also holds when the user's utility is an affine function over the state space.

Overview of the proof. To show Theorem 4, we focus on problem instances with binary state. Our proof mainly consists of two steps. In the first step, we show that for problem instances with binary state, any online policy can be transformed into a randomized online policy that only uses signaling schemes with binary signal space. This statement is no longer true for general problem instances with non-binary state, since the classic revelation principle fails. The key technical ingredient (Lemma 10) is to show that any posterior distribution of binary state can be induced by a convex combination of signaling schemes with binary signal space, which may be independent of interest. In the second step, we show a reduction from the single-item dynamic pricing problem to our online Bayesian recommendation problem with binary state. Thus, the $\Omega(\log\log T)$ regret lower bound known in dynamic pricing problem (Kleinberg and Leighton 2003) can be extended to our problem.

Below we provide detailed discussion and related lemmas for the above mentioned two steps. In the end of this subsection, we combine all pieces together to conclude the proof of Theorem 4.

Step 1: Binary signals suffice. Our first step is to show that every online policy can be transformed into a randomized online policy with binary signal space. While this might appear obvious at first as binary signal suffices in the optimal signaling scheme in hindsight, it is not a-priori clear whether restricting binary signal is without loss in an online policy without knowing user's utility.

LEMMA 9. *Given any problem instance with binary state, for any online policy* $\mathtt{ALG}$*, there exists an online policy* $\mathtt{ALG}^\dagger$ *which only uses signaling schemes with binary signal space and has regret* $\mathbf{REG}\left[\mathtt{ALG}^\dagger\right] = \mathbf{REG}[\mathtt{ALG}]$.

We now first sketch the intuition behind Lemma 9. Fix an arbitrary online policy ALG, we construct a randomized online policy $\mathsf{ALG}^\dagger$ with binary signal space that uses the original policy ALG as a blackbox. Briefly speaking, in each round t, policy $\mathsf{ALG}^\dagger$ first asks which signaling scheme π_t is used by ALG in this round. Then, $\mathsf{ALG}^\dagger$ uses a signaling scheme $\pi_t^\dagger$ with binary signal space at random[25] such that the distribution of user t's posterior belief induced in $\pi_t^\dagger$ (over the randomness of state, signaling scheme $\pi_t^\dagger$ used by ALG, and $\pi_t^\dagger$ itself) is the same as the one induced by π_t. Note that from user t's perspective, his best response is uniquely determined by his posterior belief. Thus, the distribution of user t's action is the same in both ALG and $\mathsf{ALG}^\dagger$. Finally, $\mathsf{ALG}^\dagger$ sends user t's action a_t as the feedback to ALG, and moves to the next round.

The following lemma guarantees that for any distribution μ of posterior belief over binary state, there exists a distribution of signaling schemes with binary signal space that implements μ.

LEMMA 10. *Let* $\pi : [2] \to \Delta(\Sigma)$ *be a signaling scheme that maps binary state into probability distributions over finite signal space* Σ*, and* $\mu : \Sigma \to \Delta([2])$ *be the distribution of posterior belief induced by* π*. There exists a positive integer* K*, and a finite set* $\{\pi^{(k)}\}_{k\in[K]}$ *where each* $\pi^{(k)} : [2] \to \Delta(\{0,1\})$ *is a signaling scheme with binary signal space. Let* $\mu^{(k)}$ *be the distribution of posterior belief induced by* $\pi^{(k)}$ *for each* $k \in [K]$*. Then, there exists a distribution* F *over* $[K]$ *such that for every possible posterior belief realization* $x \in \mathit{supp}(\mu)$*,* $\mathit{Pr}[\mu = x] = \mathbb{E}_{k\sim F}\left[\mathit{Pr}\left[\mu^{(k)} = x\right]\right]$.

The proof of Lemma 10 relies on Lemma 11 and Lemma 12 as follows.

LEMMA 11 (**Kamenica and Gentzkow 2011**). *Let* $\lambda \in \Delta([2])$ *be a prior distribution over binary state space* $[2]$*. A distribution of posterior belief* $\mu \in \Delta(\Delta([2]))$ *is implementable (i.e., can be induced by some signaling scheme) if and only if* $\mathit{Pr}_{x\sim\mu,\theta\sim x}[\theta = 1] = \lambda(1)$.

LEMMA 12. *Let* X *be a random variable with discrete support* $\mathit{supp}(X)$*. There exists a positive integer* K*, a finite set of* K *random variables* $\{X_k\}_{k\in[K]}$*, and convex combination coe cients* $\mathbf{f} \in [0,1]^K$ *with* $\sum_{k\in[K]} f_k = 1$ *such that:*

1. *Bayesian-plausibility: for each* $k \in [K]$*,* $\mathbb{E}[X_k] = \mathbb{E}[X]$*;*
2. *Binary-support: for each* $k \in [K]$*, the size of* X_k*'s support is at most 2, i.e.,* $|\mathit{supp}(X_k)| \le 2$
3. *Consistency: for each* $x \in \mathit{supp}(X)$*,* $\mathit{Pr}[X = x] = \sum_{k\in[K]} f_k \cdot \mathit{Pr}[X_k = x]$

The proof of the above lemma is provided in Section E. Now we are ready to show Lemma 10.

[25] Namely, $\mathsf{ALG}^\dagger$ randomly picks a signaling scheme $\pi_t^\dagger$ and commits to it in round t.

Proof of Lemma 10. Let $\lambda \in \Delta([2])$ be the prior distribution over binary state space $[2]$, and θ be the state drawn from λ. Fix an arbitrary signaling scheme π and let μ be the distribution of posterior belief induced by π. Let σ be the signal issued by signaling scheme π, and set random variable $X = \Pr[\theta = 1 \mid \sigma]$. By Lemma 11, $\mathbb{E}_\sigma[X] = \lambda(1)$.

Lemma 12 ensures that there exists a positive integer K, a finite set of K random variable $\{X_k\}$, and convex combination coefficients $\mathbf{f}$ that satisfy "Bayesian-plausibility" property, "binary-support" property, and "consitency" property. Invoking Lemma 11, we know that each random variable X_k can be thought as a distribution of posterior belief $\mu^{(k)}$ which can be induced by some signaling scheme $\pi^{(k)}$ due to the "Bayesian-plausibility" property. The "binary-support" property ensures that $\pi^{(k)}$ has binary signal space. Let F be the distribution over $[K]$ such that $\Pr_{k\sim F}[k = \ell] = f_\ell$. The "consitency" property guarantees that for every possible posterior belief realization $x \in \mathsf{supp}(\mu)$, $\Pr[\mu = x] = \mathbb{E}_{k\sim F}\left[\Pr\left[\mu^{(k)} = x\right]\right]$. □

Now with Lemma 10, we can prove Lemma 9, whose proof is provided in Section E.

Step 2: Reduction from dynamic pricing. The second step in the proof of Theorem 4 is a reduction from the single-item dynamic pricing problem to our online Bayesian recommendation problem. The definition of single-item dynamic pricing problem is as follows.

Definition 3. In the *single-item dynamic pricing problem*, there is a seller with unlimited units of a single item and T buyers. In each round $t \in [T]$, the seller wants to sell a new unit of the item (by setting a price p_t) to buyer t. Buyer t has a private value v^* that is unknown to the seller, and will buy the item (and pay p_t) if and only if $v^* \geq p_t$. The regret of a dynamic pricing mechanism ALG is

$$\mathbf{REG}[\mathsf{ALG}] \triangleq T \cdot v^* - \mathbb{E}_{p_1,\ldots,p_T}\left[\sum_{t\in[T]} p_t \cdot \mathbb{1}[p_t \leq v^*]\right]$$

where p_t is the price posted by ALG in each round $t \in [T]$.

Theorem 5 (**Kleinberg and Leighton 2003**). *In single-item dynamic pricing problem, no randomized dynamic pricing mechanism can achieve an expected regret better than* $\Omega(\log\log T)$.

The following lemma formally states the reduction from the single-item dynamic pricing problem to our online Bayesian recommendation problem.

Lemma 13. *For every single-item dynamic pricing problem instance I, there exists an online Bayesian recommendation problem instance $I^\dagger$ with binary state. For every online policy $\mathtt{ALG}^\dagger$ with binary signal space and regret $\mathbf{REG}_{I^\dagger}\left[\mathtt{ALG}^\dagger\right]$ on online Bayesian recommendation instance $I^\dagger$, there exists a dynamic pricing mechanism $\mathtt{ALG}$ with regret $\mathbf{REG}_I[\mathtt{ALG}] \leq \mathbf{REG}_{I^\dagger}\left[\mathtt{ALG}^\dagger\right] + 1$ on dynamic pricing instance I.*

Here we present a sketch of our reduction from the single-item dynamic pricing problem to our problem. The formal proof of Lemma 13 is deferred to Section E.

Proof sketch of Lemma 13. Consider the following reduction, which contains a mapping from dynamic pricing instance I to online Bayesian recommendation instance $I^\dagger$,[26] and a mapping from online policy $\mathsf{ALG}^\dagger$ to dynamic pricing mechanism ALG.

Instance mapping: Fix an arbitrary single-item dynamic pricing problem instance $I = (T, v^*)$ where there are T rounds and each buyer has private value v^*. Consider the following Bayesian recommendation instance $I^\dagger$. There are $m^\dagger = 2$ states, and $T^\dagger = T$ rounds. Let $\epsilon = 1/T^\dagger$. State 1 is realized with probability $\lambda^\dagger(1) = \epsilon$ and state 2 is realized with probability $\lambda^\dagger(2) = 1 - \epsilon$. The users' utility is defined as follows,

$$\text{for state 1:} \quad \rho^\dagger(1, a^\dagger) = \mathbb{1}\left[a^\dagger = 1\right], \quad \text{for state 2:} \quad \rho^\dagger(2, a^\dagger) = -\frac{\epsilon}{v^*} \cdot \mathbb{1}\left[a^\dagger = 1\right] \tag{1}$$

By construction, $\omega^\dagger(1) = \epsilon$, $\omega^\dagger(2) = -\frac{\epsilon(1-\epsilon)}{v^*}$, and the optimal signaling in hindsight $\pi^{*\dagger}$ satisfies that $\pi^{*\dagger}(1) = 1$, $\pi^{*\dagger}(2) = \frac{v^*}{1-\epsilon}$, and $U(\pi^{*\dagger}) = v^* + \epsilon$.

Policy mapping: Fix an arbitrary online policy $\mathsf{ALG}^\dagger$ with binary signal space for online Bayesian recommendation instance $I^\dagger$. We construct dynamic pricing mechanism ALG round by round. Suppose signaling scheme $\pi_t^\dagger$ is used by $\mathsf{ALG}^\dagger$ in round t. Here we assume that $\pi_t^\dagger(1) = 1$.[27] Then dynamic pricing mechanism ALG posts price $p_t \triangleq (1-\epsilon)\pi_t^\dagger(2)$ in round t for the dynamic pricing instance I.

Reduction analysis: To see why $\mathbf{REG}_I[\mathsf{ALG}] \le \mathbf{REG}_{I^\dagger}\left[\mathsf{ALG}^\dagger\right] + 1$, let us fix an arbitrary round t. Under the assumption $\pi_t^\dagger(1) = 1$, user t takes action 1 if and only if the realized signal $\sigma^\dagger = 1$ and her expected utility of taking action 1 is better than taking action 0 under her posterior belief, i.e.,

$$\omega^\dagger(1)\,\pi_t^\dagger(1) + \omega^\dagger(2)\,\pi_t^\dagger(2) \ge 0 \quad \Rightarrow \quad \pi_t^\dagger(2) \le \frac{v^*}{1-\epsilon}$$

Hence, the expected regret induced by signaling scheme $\pi_t^\ddagger$ is

$$\begin{aligned}\mathbf{REG}_{I^\dagger}\left[\pi_t^\dagger\right] &= U(\pi^{*\dagger}) - \left(\lambda^\dagger(1)\pi_t^\dagger(1) + \lambda^\dagger(2)\pi_t^\dagger(2)\right) \cdot \mathbb{1}\left[\text{user } t \text{ takes action } 1 \mid \sigma^\dagger = 1\right] \\ &= v^* + \epsilon - \left(\epsilon + (1-\epsilon)\pi_t^\dagger(2)\right) \cdot \mathbb{1}\left[\pi_t^\dagger(2) \le \frac{v^*}{1-\epsilon}\right]\end{aligned}$$

On the other hand, when price $p_t \triangleq (1-\epsilon)\pi_t^\dagger(t)$ is posted by ALG, the regret is

$$\mathbf{REG}_I[p_t] = v^* - p_t \cdot \mathbb{1}[p_t \le v^*] \le \mathbf{REG}_{I^\dagger}\left[\pi_t^\dagger\right] + \epsilon$$

[26] Here we use notation † to denote the online Bayesian recommendation instance.

[27] In the formal proof of Lemma 13 (Section E), we show that this assumption is without loss of generality.

Since dynamic pricing mechanism ALG has more information than online policy $\mathsf{ALG}^\dagger$,[28] ALG can simulate $\mathsf{ALG}^\dagger$ in the future rounds. The total regret is

$$\mathbf{REG}_I[\mathsf{ALG}] - \mathbf{REG}_{I^\dagger}\left[\mathsf{ALG}^\dagger\right] = \sum_{t\in[T]} \left(\mathbf{REG}_I[p_t] - \mathbf{REG}_{I^\dagger}\left[\pi_t^\dagger\right]\right) \le \epsilon \cdot T = 1$$

which finishes the sketch of our reduction.

REMARK 5. We would like to note that the binary-state instance constructed in (1) also satisfies that the user's utility function is an affine function over the states.

Putting all pieces together, we are ready to proof Theorem 4.

Proof of Theorem 4. Combining Theorem 5 and Lemma 13, in the online Bayesian recommendation problem with binary state, no randomized online policy with binary signal space can achieve an expected regret better than $\Omega(\log\log T)$. Invoking Lemma 9 finishes the proof. □

Comparison with (contextual) dynamic pricing problem. In Lemma 13, we give a reduction from the single-item dynamic pricing problem to our online Bayesian recommendation problem with binary state. Roughly speaking, our problem with binary state can be interpreted as a dynamic pricing problem with probabilistic feedback – when price p is posted, seller only learns whether buyer's value is greater than price p with probability $1/p$. Since the feedback is probabilistic, the classic dynamic pricing mechanism with $O(\log\log T)$ regret studied in Kleinberg and Leighton (2003) suffers significantly larger regret in our problem. In contrast, our ConRP uses exploring phase I to resolve this issue.

When the size of state space (i.e., m) is large, ConRP incurs an $\mathrm{O}(m\cdot 2^{m-1})$ regret dependence, which may not be ideal. A natural question is whether we can improve the dependence on m to $\mathsf{poly}(m)$. To answer this question, one natural attempt is to revisit the multi-dimension generalization of the single-item dynamic pricing problem – contextual dynamic pricing problem, in which Leme and Schneider (2018) design a contextual dynamic pricing mechanism with $O(\mathsf{poly}(m)\log\log T)$ regret. In the contextual dynamic pricing problem, the item has m features, and buyers have private value $v^*(i)$ for each feature i. In each round $t\in[T]$, the nature selects a vector $(x_t(1),\ldots,x_t(m))\in\mathbb{R}^m_{\ge 0}$, and the seller wants to sell a new unit of the item by setting a price p_t to buyer t, who will buy the item (and pay p_t) if and only if $\sum_{i\in[m]} v^*(i)x_t(i)\ge p_t$.

The contextual dynamic pricing problem shares some similarity to our problem with multiple states. Specifically, there is an unknown vector $\{v^*(i)\}$ (resp. $\{\delta(i)\}$), and the optimal in hindsight benchmarks can be formulated as similar linear programs depending on $\{v^*(i)\}$ (resp. $\{\delta(i)\}$).

[28] In particular, dynamic pricing mechanism deterministically learns whether $p_t\le v^*$ (a.k.a., $\mathbb{1}\left[\pi_t^\dagger(2)\le\frac{v^*}{1-\epsilon}\right]$), while online policy $\mathsf{ALG}^\dagger$ only learns this information when signal 1 is realized.

Nonetheless, there exist fundamental differences between the two problems besides the probabilistic and limited feedback feature mentioned before. In particular, in each round, the contextual dynamic pricing mechanism chooses a price p_t which is a scalar, while the online Bayesian recommendation policy chooses a signaling scheme (i.e., a high-dimensional function). In Leme and Schneider (2018), authors obtain $O(\mathsf{poly}(m)\log\log T)$ regret by formulating the contextual dynamic pricing as solving linear programs with a separation oracle.[29] However, in our problem, it is unclear if such a simple separation oracle exists. In Section 5, we introduce an online policy with $O(\mathsf{poly}(m\log T))$ regret by formulating our problem as solving linear programs with a membership oracle.[30]

8. Conclusions and Future Work

In this paper, we have studied the online Bayesian recommendation problem with featuring a two-sided information asymmetry where the platform knows the payoff-relevant state but does not know the user's preference (and belief), and the user knows his preference but is uncertain about the payoff-relevant state. Focusing on policies that minimize the Stackelberg regret, we present two algorithms. The first algorithm is a conservative recommendation policy (`ConRP`). We show that this algorithm can achieve $O(\log\log T)$ regret when the platform knows user's ordinal preference over the states. Moreover, this algorithm can also be readily adapted to the setting with unknown ordinal preference. In particular, same regret $O(\log\log T)$ can be achieved when the user's preference is affine with respect to the state, and regret $O(m2^{m-1}\cdot\log\log T)$ can be achieved for arbitrary preference. Our second algorithm is a linear programming-based algorithm (`LP-RP`) that utilizes the problem structure and can achieve $O(\mathsf{poly}(m\log T))$ regret, which is more desired when the number of states m is large and the user's preference is arbitrarily.

Future research. Our research opens a number of interesting and challenging questions for future research.

Better regret dependency on state space. Our lower bound only establishes regret dependency on the time horizon T, and does not rule out the possibility to design an algorithm with achieving $O(\mathsf{poly}(m)\cdot\log\log T)$ regret. Thus, it would be interesting to explore whether one can tighten up the lower bound that has exponential dependency on the number of states m, or design an algorithm whose regret has only polynomial dependency on the number of states m with still double-logarithmically depending on time horizon T. Making progress in this direction likely requires a more judicious characterization on the underlying geometry of our online problem.

[29] In particular, vector $(x_t(1),\ldots,x_t(m))$ is served as the separating hyperplane for the oracle.

[30] Membership oracle is weaker than separation oracle. See more discussion between the two oracles in Section 5.

Extension with users' heterogeneity. In the paper, we focus on the setting where the platform is interacting with users that have the same preference. However, in some applications, users may be heterogeneous and have different preferences or beliefs over the payoff-relevant states. Thus, an important extension of our problem with significant practical implications would be to consider a setting that captures the users' heterogeneity.

Strategical information revealing in joint matching and recommendation. Finally, the key that motivates this paper is the two-sided information asymmetry between the platform and the users, and we study how to optimally recommend a particular item (e.g., the short video in Tiktok) in the long run. Another ongoing line of research (e.g., Bimpikis et al. 2020, Ashlagi et al. 2021) focuses on how to strategically disclose information to achieve an optimal matching between the items and the users. Hence, a more general problem beyond our setting is that: under the two-sided information asymmetry, how to make a joint decision on matching and recommending a particular item from a pool of available items with the user who has the potential interest.

References

Shipra Agrawal, Vashist Avadhanula, Vineet Goyal, and Assaf Zeevi. Mnl-bandit: A dynamic learning approach to assortment selection. *Operations Research*, 67(5):1453–1485, 2019.

Ricardo Alonso and Odilon Camara. Bayesian persuasion with heterogeneous priors. *Journal of Economic Theory*, 165:672–706, 2016.

Itai Arieli, Yakov Babichenko, Rann Smorodinsky, and Takuro Yamashita. Optimal persuasion via bi-pooling. *Theoretical Economics*, 18(1):15–36, 2023.

Itai Ashlagi, Faidra Monachou, and Afshin Nikzad. Optimal dynamic allocation: Simplicity through information design. In *Proceedings of the 22nd ACM Conference on Economics and Computation*, pages 101–102, 2021.

Yakov Babichenko, Inbal Talgam-Cohen, Haifeng Xu, and Konstantin Zabarnyi. Regret-minimizing bayesian persuasion. In *Proceedings of the 22nd ACM Conference on Economics and Computation*, pages 128–128, 2021.

Kostas Bimpikis, Yiangos Papanastasiou, and Wenchang Zhang. Information provision in two-sided platforms: Optimizing for supply. *Available at SSRN 3617351*, 2020.

Avrim Blum and Yishay Mansour. From external to internal regret. *Journal of Machine Learning Research*, 8(6), 2007.

Avrim Blum, Nika Haghtalab, and Ariel D Procaccia. Learning optimal commitment to overcome insecurity. *Advances in Neural Information Processing Systems*, 27, 2014.

Ozan Candogan. Persuasion in networks: Public signals and cores. *Operations research*, (4):2264–2298, 2022.

Ozan Candogan and Philipp Strack. Optimal disclosure of information to a privately informed receiver. In *Proceedings of the 22nd ACM Conference on Economics and Computation*, pages 263–263, 2021.

Matteo Castiglioni, Andrea Celli, Alberto Marchesi, and Nicola Gatti. Online bayesian persuasion. *Advances in Neural Information Processing Systems*, 33, 2020.

Matteo Castiglioni, Alberto Marchesi, Andrea Celli, and Nicola Gatti. Multi-receiver online bayesian persuasion. In *International Conference on Machine Learning*, pages 1314–1323. PMLR, 2021.

Xi Chen, Yining Wang, and Yuan Zhou. Optimal policy for dynamic assortment planning under multinomial logit models. *Mathematics of Operations Research*, 46(4):1639–1657, 2021.

Yiling Chen, Yang Liu, and Chara Podimata. Learning strategy-aware linear classifiers. *Advances in Neural Information Processing Systems*, 33:15265–15276, 2020.

Laurens G Debo, Christine Parlour, and Uday Rajan. Signaling quality via queues. *Management Science*, 58 (5):876–891, 2012.

Jinshuo Dong, Aaron Roth, Zachary Schutzman, Bo Waggoner, and Zhiwei Steven Wu. Strategic classification from revealed preferences. In *Proceedings of the 2018 ACM Conference on Economics and Computation*, pages 55–70, 2018.

Shaddin Dughmi and Haifeng Xu. Algorithmic bayesian persuasion. *SIAM Journal on Computing*, 50(3): STOC16–68, 2019.

Piotr Dworczak and Alessandro Pavan. Preparing for the worst but hoping for the best: Robust (bayesian) persuasion. 2020.

Yiding Feng, Wei Tang, and Haifeng Xu. Online bayesian recommendation with no regret. In *Proceedings of the 23rd ACM Conference on Economics and Computation*, pages 818–819, 2022.

Flixier. Ideal tiktok video length and size in 2022 - the ultimate guide. `https://flixier.com/blog/ideal-tiktok-video-length-and-size-in-2022`, 2022.

Matthew Gentzkow and Emir Kamenica. A rothschild-stiglitz approach to bayesian persuasion. *American Economic Review*, 106(5):597–601, 2016.

Werner Geyser. Tiktok video ad specs and best practices for 2022. `https://influencermarketinghub.com/tiktok-video-ad-specs/`, 2022.

Martin Grötschel, László Lovász, and Alexander Schrijver. *Geometric algorithms and combinatorial optimization*, volume 2. Springer Science & Business Media, 2012.

Ju Hu and Xi Weng. Robust persuasion of a privately informed receiver. *Economic Theory*, 72(3):909–953, 2021.

Bar Ifrach, Costis Maglaras, Marco Scarsini, and Anna Zseleva. Bayesian social learning from consumer reviews. *Operations Research*, 67(5):1209–1221, 2019.

Emir Kamenica and Matthew Gentzkow. Bayesian persuasion. *American Economic Review*, 101(6):2590–2615, 2011.

Robert Kleinberg and Tom Leighton. The value of knowing a demand curve: Bounds on regret for online posted-price auctions. In *44th Annual IEEE Symposium on Foundations of Computer Science, 2003. Proceedings.*, pages 594–605. IEEE, 2003.

Knowledge Sourcing Intelligence LLP. Social Networking Platforms Market - Forecasts from 2021 to 2026. https://www.researchandmarkets.com/reports/5332678/social-networking-platforms-market-forecasts, 2021.

Anton Kolotilin. Optimal information disclosure: A linear programming approach. *Theoretical Economics*, 13 (2):607–635, 2018.

Anton Kolotilin, Tymofiy Mylovanov, Andriy Zapechelnyuk, and Ming Li. Persuasion of a privately informed receiver. *Econometrica*, 85(6):1949–1964, 2017.

Svetlana Kosterina. Persuasion with unknown beliefs. *Work. Pap., Princeton Univ., Princeton, NJ*, 2018.

Ilan Kremer, Yishay Mansour, and Motty Perry. Implementing the "wisdom of the crowd". *Journal of Political Economy*, 122(5):988–1012, 2014.

Yin Tat Lee, Aaron Sidford, and Santosh S Vempala. Efficient convex optimization with membership oracles. In *Conference On Learning Theory*, pages 1292–1294. PMLR, 2018.

Renato Paes Leme and Jon Schneider. Contextual search via intrinsic volumes. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 268–282. IEEE, 2018.

David Lingenbrink and Krishnamurthy Iyer. Optimal signaling mechanisms in unobservable queues. *Operations research*, 67(5):1397–1416, 2019.

Allen Liu, Renato Paes Leme, and Jon Schneider. Optimal contextual pricing and extensions. In *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1059–1078. SIAM, 2021.

Ilan Lobel, Renato Paes Leme, and Adrian Vladu. Multidimensional binary search for contextual decision-making. *Operations Research*, 66(5):1346–1361, 2018.

Yishay Mansour, Aleksandrs Slivkins, and Vasilis Syrgkanis. Bayesian incentive-compatible bandit exploration. In *Proceedings of the Sixteenth ACM Conference on Economics and Computation*, pages 565–582, 2015.

Yishay Mansour, Alex Slivkins, Vasilis Syrgkanis, and Zhiwei Steven Wu. Bayesian exploration: Incentivizing exploration in bayesian games. *Operations Research*, 2021.

Jérôme Renault, Eilon Solan, and Nicolas Vieille. Optimal dynamic information provision. *Games and Economic Behavior*, 104:329–349, 2017.

Paat Rusmevichientong, Zuo-Jun Max Shen, and David B Shmoys. Dynamic assortment optimization with a multinomial logit choice model and capacity constraint. *Operations research*, 58(6):1666–1680, 2010.

Denis Sauré and Assaf Zeevi. Optimal dynamic assortment planning with demand learning. *Manufacturing & Service Operations Management*, 15(3):387–404, 2013.

Dongwook Shin, Stefano Vaccari, and Assaf Zeevi. Dynamic pricing with online reviews. *Management Science*, 69(2):824–845, 2023.

You Zu, Krishnamurthy Iyer, and Haifeng Xu. Learning to persuade on the fly: Robustness against ignorance. In *Proceedings of the 22nd ACM Conference on Economics and Computation*, pages 927–928, 2021.

Appendix

A. Extensions

In this section, we briefly discuss two extensions: (i) platform with state-dependent utility function, and (ii) platform uses misspecified prior beliefs. In both extensions, the regret guarantees for both `ConRP` and `LP-RP` continue to hold.

Platform with state-dependent utility function. Recall that our baseline model assumes that the platform's utility function $\xi(\cdot)$ is state-independent, i.e., the platform gains one unit of profit if a user takes action 1. In this subsection, we relax this assumption and consider the platform's utility function $\xi : [m] \times \mathcal{A} \to \mathbb{R}$ as a mapping from both the realized state and the user's action to the utility of the platform. Additionally, we assume that $\xi(i,0) = 0$ and $\xi(i,1) \in [0,1]$ for every state $i \in [m]$.

Under this variant model, it can be verified that the regret guarantees in Theorem 1 and Theorem 2 continue to hold for modified versions of `ConRP` and `LP-RP` as follows.

In the modified version of `ConRP`, variables $\underline{\mathrm{U}}, L, R$ now denote the lower bound and upper bound of $U(\pi^*) = \sum_{i\in[m]} \lambda(i)\xi(i,1)\pi^*(i)$. Each direct signaling schemes $\pi^{(r,u)}$ used in the exploring phase is a signaling scheme such that there exists a threshold state $i^\dagger$ such that (a) for every state $i \neq i^\dagger$, $\pi^{(r,u)}(i) = \mathbb{1}[r(i) > r(i^\dagger)]$, and (b) $\pi^{(r,u)}(i^\dagger) = \frac{u - \sum_{i\neq i^\dagger} \lambda(i)\xi(i,1)\pi^{(r,u)}(i)}{\lambda(i^\dagger)\xi(i,1)}$. All other parts of `ConRP` remain the same.

The modification of `LP-RP` is similar. Variable $\underline{\mathrm{U}}$ now denotes the lower bound of $U(\pi^*)$. In the construction of signaling scheme π^I, π^{II}, and $\pi^{(0)}$, holding everything else the same as before, we modify $\pi^I(j^\ddagger) = \frac{\underline{\mathrm{U}}}{\lambda(j^\ddagger)\xi(j^\ddagger,1)}$, $\pi^{\mathrm{II}}(j^\ddagger) = \frac{\underline{\mathrm{U}}}{2\lambda(j^\ddagger)\xi(j^\ddagger,1)}$, and $\pi^{(0)}(j^\ddagger) = \frac{\underline{\mathrm{U}}}{8\lambda(j^\ddagger)\xi(j^\ddagger,1)}$. All other parts of `LP-RP` remain the same.

Users with misspecified beliefs. Our algorithm and results can be directly extended to the setting where users share different prior beliefs with the platform (Alonso and Camara 2016). In particular, we let $\lambda \in \Delta([m])$ denote the prior belief of the platform, and $\lambda^\dagger \in \Delta([m])$ denote the prior belief of users. In this setting, the optimal signaling scheme in hindsight π^* can be solved by the following linear program,

$$\begin{aligned}
\pi^* = \arg\max_{\pi} \quad & \sum_{i\in[m]} \lambda(i)\pi(i,1) && \text{s.t.} \\
& \sum_{i\in[m]} \left(\rho(i,1) - \rho(i,0)\right) \lambda^\dagger(i)\pi(i,1) \geq 0 && \\
& \pi(i,1) + \pi(i,0) = 1 && i \in [m] \\
& \pi(i,1) \geq 0, \pi(i,0) \geq 0 && i \in [m]
\end{aligned}$$

By rewriting $(\rho(i,1)-\rho(i,0))\,\lambda^\dagger(i)$ as $\rho'(i,1)-\rho'(i,0)$, we can observe that this is equivalent to[31] the original Bayesian recommendation problem solved by ConRP and LP-RP, where the platform and user share the same beliefs, and the user has the preference $\rho(\theta,a) \leftarrow \rho(\theta,a)\lambda^\dagger(\theta)$. Moreover, it can be verified that the regret guarantees for both online policies continue to hold in this extension.

B. Omitted Proofs in Section 2

In this section, we present the omitted proofs of Lemma 2, Lemma 3, and Lemma 4 in Section 2.

LEMMA 2. *A direct signaling scheme π is obedient if and only if $\sum_{i\in[m]}\delta(i)\lambda(i)\pi(i)\geq 0$.*

Proof. When action 1 is recommended, the posterior distribution is $\mu_t(1,i)=\frac{\lambda(i)\pi(i)}{\sum_{j\in[m]}\lambda(j)\pi(j)}$. Thus, user takes action 1 if and only if

$$\frac{1}{\sum_{j\in[m]}\lambda(j)\pi(j)}\sum_{i\in[m]}\rho(i,1)\lambda(i)\pi(i)\geq\frac{1}{\sum_{j\in[m]}\lambda(j)\pi(j)}\sum_{i\in[m]}\rho(i,0)\lambda(i)\pi(i)$$

Rearranging the terms finishes the proof. □

LEMMA 3. *Given a direct signaling scheme π, Procedure 1 returns* True *only if π is obedient, and returns* False *only if π is not obedient.*

Proof. By construction, Procedure 1 returns True if $\sigma_t=1=a_t$, which is exactly the definition of obedience. Similarly, Procedure 1 returns False if $\sigma_t=1=1-a_t$ or $\sigma_t=0=1-a_t$. The correctness of the former case holds due to the definition of obedience. To see the correctness of the latter case, note that when action 0 is recommended, user takes action 1 if and only if $\sum_{i\in[m]}\delta(i)\lambda(i)(1-\pi(i))\geq 0$. Hence,

$$\sum_{i\in[m]}\delta(i)\lambda(i)\pi(i)\leq\sum_{i\in[m]}\delta(i)\lambda(i)<0$$

where the last inequality holds due to Assumption 2. Invoking Lemma 2 finishes the proof. □

LEMMA 4. *Given a direct signaling scheme π, the expected regret of Procedure 1 is at most $\frac{U(\pi^*)}{\sum_i\lambda(i)\pi(i)}-\mathbb{1}[\pi$ is obedient$]$.*

Proof. Let Q be the number of rounds used in Procedure 1. We start by upper bounding $\mathbb{E}[Q]$. Note that Procedure 1 returns if action 1 is recommended, which happens with probability $\sum_i\lambda(i)\pi(i)$ in each round. Thus, $\mathbb{E}[Q]\leq\frac{1}{\sum_i\lambda(i)\pi(i)}$, and the expected regret is at most

$$\mathbb{E}[Q]\,(U(\pi^*)-U(\pi))\leq\frac{U(\pi^*)}{\sum_i\lambda(i)\pi(i)}-\frac{U(\pi)}{\sum_i\lambda(i)\pi(i)}\leq\frac{U(\pi^*)}{\sum_i\lambda(i)\pi(i)}-\mathbb{1}[\pi\text{ is obedient}]$$

where the last inequality holds since $U(\pi)\geq(\sum_i\lambda(i)\pi(i))\cdot\mathbb{1}[\pi$ is obedient$]$. □

Algorithm 4: Conservative Recommendation Policy (ConRP) for unknown order preference

Input: number of rounds T, number of states m, prior distribution λ

Initialize the set $\mathcal{P}$ to contain all possible orders described as above.

/* exploring phase I - identify $\underline{U}$ such that $\underline{U} \le U(\pi^*) \le 2\underline{U}$ */

Initialize $\underline{U} \leftarrow \frac{1}{2}$

while True **do**

  if *there exists* $r \in \mathcal{P}$ *such that* CheckObed$\left(\pi^{(r,\underline{U})}\right) =$ True **then**

    $\mathcal{P} \leftarrow \left\{r \in \mathcal{P} : \text{CheckObed}\left(\pi^{(r,\underline{U})}\right) = \text{True}\right\}$

    break

  end

  else

    $\underline{U} \leftarrow \frac{\underline{U}}{2}$

end

/* exploring phase II - identify a signaling scheme $\pi^\dagger$ such that $U(\pi^\dagger) \ge U(\pi^*) - \frac{1}{T}$ */

Initialize $R \leftarrow 2\underline{U}$, $L \leftarrow \underline{U}$, $\delta \leftarrow 1$

while $R - L \ge \frac{1}{T}$ **do**

  $\varepsilon \leftarrow \frac{\delta}{2}$, $S \leftarrow \lfloor \frac{R-L}{\varepsilon L} \rfloor$

  for $\ell = 1, 2, \ldots, S$ **do**

    if *there exists* $r \in \mathcal{P}$ *such that* CheckObed$\left(\pi^{(r,L+\ell\varepsilon L)}\right) =$ True **then**

      $\mathcal{P} \leftarrow \left\{r \in \mathcal{P} : \text{CheckObed}\left(\pi^{(r,L+\ell\varepsilon L)}\right) = \text{True}\right\}$

    end

    else

      $R \leftarrow L + \ell\varepsilon L$, $L \leftarrow L + (\ell-1)\varepsilon L$, $\delta \leftarrow \varepsilon^2$

      break

  end

end

Set $\pi^\dagger \leftarrow \pi^{(r,L)}$ for an arbitrary $r \in \mathcal{P}$

/* exploiting phase */

Use signaling scheme $\pi^\dagger$ for all remaining rounds.

C. Omitted Proofs and Algorithm in Section 4

C.1. Omitted Proof and Algorithm in Section Section 4.2

PROPOSITION 2. *When the platform has no knowledge of user's preference, the expected regret of a modified version (see Algorithm 4 in Section C.1) of* ConRP *is* $O(m2^{m-1} \cdot \log\log T)$.

We analyze the expected regret in exploring phase I, exploring phase II, and exploiting phase separately. We first assume that Algorithm 4 finishes exploring phase I and II before T rounds are

[31] Specifically, there exists a bijection between the problem instances in both problem, such that the optimal signaling schemes in hindsight and the corresponding utilities of the platform are identical.

exhausted. Similar argument follows for the other case where exploring phase I or exploring phase II is completed due to the exhaustion of rounds.

Exploring phase I. Let $K = -\lceil \log(U(\pi^*)) \rceil$. By definition, $\texttt{CheckObed}(\pi^{(r,2^{-k})}) = \texttt{False}$ for all $r \in \mathcal{P}$ and $k \in [K-1]$, and $\texttt{CheckObed}(\pi^{(r^*,2^{-K})}) = \texttt{True}$. Thus, in the end of exploring phase I, $\underline{\mathrm{U}}$ is 2^{-K}, and there are K iterations in the while loop. For each iteration $k \in [K]$, $\texttt{CheckObed}(\pi^{(r,2^{-k})})$ is called for every $r \in \mathcal{P}$. By Lemma 4, the total expected regret is

$$\sum_{k\in[K]} \sum_{r\in\mathcal{P}} \frac{U(\pi^*)}{\sum_{i\in[m]} \lambda(i)\pi^{(r,2^{-k})}(i)} \overset{(a)}{\leq} \sum_{k\in[K]} \sum_{r\in\mathcal{P}} \frac{2^{-(K-1)}}{2^{-k}} = |\mathcal{P}| \sum_{k\in[K]} 2^{-(K-k-1)} \leq 4(m!)$$

where the denominator in the right-hand side of inequality (a) is due to the construction of $\pi^{(r,2^{-k})}$.

Exploring phase II. By construction, there are $O(\log\log T)$ iterations in the while loop. Thus, it is sufficient to show the expected regret in each iteration is $O(m2^{m-1})$.

In each iteration k, for every $r \in \mathcal{P}$, let $\ell^\dagger \in [S]$ be the smallest index that the signaling scheme $\pi^{(r,L+\ell^\dagger\varepsilon L)}$ is not obedient. The expected regret in iteration k for $r \in \mathcal{P}$ is at most

$$\begin{aligned}
&\sum_{\ell=1}^{\ell^\dagger-1} \left(\frac{U(\pi^*)}{\sum_{i\in[m]} \lambda(i)\pi^{(r,L+\ell\varepsilon L)}(i)} - 1 \right) + \frac{U(\pi^*)}{\sum_{i\in[m]} \lambda(i)\pi^{(r,L+\ell^\dagger\varepsilon L)}(i)} \\
&\overset{(a)}{=} \sum_{\ell=1}^{\ell^\dagger-1} \left(\frac{U(\pi^*)}{L+\ell\varepsilon L} - 1 \right) + \frac{U(\pi^*)}{L+\ell^\dagger\varepsilon L} \overset{(b)}{\leq} \sum_{\ell=1}^{\ell^\dagger-1} \left(\frac{R}{L} - 1 \right) + \frac{R}{L} \overset{(c)}{\leq} (S-1)\frac{R-L}{L} + 2 \overset{(d)}{\leq} \frac{(R-L)^2}{\varepsilon L^2} + 2
\end{aligned}$$

where equality (a) holds due to the construction of $\pi^{(r,L+\ell\varepsilon L)}$ and $\pi^{(r,L+\ell^\dagger\varepsilon L)}$; inequality (b) holds since $U(\pi^*) \leq R$; inequality (c) holds since $\ell^\dagger \leq S$ and $R \leq 2L$; and inequality (d) holds since $S = \lfloor \frac{R-L}{\varepsilon L} \rfloor$.

We finish this part by showing $R - L \leq \sqrt{2\varepsilon}L$ by induction. Let $L^{(k)}, R^{(k)}$, $\delta^{(k)}$ and $\varepsilon^{(k)}$ be the value of $L, R, \delta, \varepsilon$ in each iteration k. The claim is satisfied for iteration $k = 1$, since $R^{(1)} - L^{(1)} = 2\underline{\mathrm{U}} - \underline{\mathrm{U}} = L^{(1)}$ and $\varepsilon^{(1)} = 1/2$. Suppose the claim holds for iteration $k-1$. Now, for iteration k, we know that $R^{(k)} - L^{(k)} = \varepsilon^{(k-1)}L^{(k-1)} \leq \varepsilon^{(k-1)}L^{(k)} = \sqrt{\delta^{(k)}}L^{(k)} = \sqrt{2\varepsilon^{(k)}}L^{(k)}$, which finishes the induction.

Exploiting phase. By construction, $\mathcal{P}$ is not empty in the end of exploring phase II, and thus signaling scheme $r^\dagger$ is well-defined. Additionally, we know that $\pi^\dagger$ is obedient and $U(\pi^\dagger) \geq U(\pi^*) - 1/T$, which concludes the proof. □

D. Formal Proofs of Algorithm 3

Here we explain each phase of Algorithm 3, i.e., LP-RP, in details. By combining the regret analysis in all phases, we prove Theorem 2 in the end of this section.

The analysis of exploring phase I. We use the following lemma to characterize exploring phase I.

LEMMA 14 **(restatement of Lemma 5)**. *Suppose* $U(\pi^*) \geq \frac{1}{T}$. *Let* $i^\ddagger = \arg\max_{i\in[m]} \delta(i)\lambda(i)$. *When exploring phase I terminates,* $\underline{U} \geq \frac{U(\pi^*)}{m^2}$ *and there exists a state* $j^\ddagger \in [m]$ *such that* $(i^\ddagger, j^\ddagger) \in \mathcal{S}$.

Proof. Let state $j' = \arg\max_{j\in[m]} \lambda(j)\pi^*(j)$, namely, j' is the state that contributes the most to $U(\pi^*)$. Consider a direct signaling scheme π with $\pi(i^\ddagger) = 1$, $\pi(j') = \frac{\pi^*(j')}{m-1}$ if $j' \neq i^\ddagger$, and $\pi(i) = 0$ for every $i \notin \{i^\ddagger, j'\}$. We claim that π is obedient. To see this, note that if $j' \neq i^\ddagger$,

$$\sum_{i\in[m]} \delta(i)\lambda(i)\pi(i) = \omega(i^\ddagger)\pi(i^\ddagger) + \omega(j')\pi(j') = \omega(i^\ddagger) + \frac{1}{m-1}\omega(j')\pi^*(j')$$
$$\overset{(a)}{\geq} \frac{1}{m-1} \sum_{i\in[m]:i\neq j'} \omega(i)\pi^*(i) + \frac{1}{m-1}\omega(j')\pi^*(j') \overset{(b)}{\geq} 0$$

where inequality (a) holds since $\omega(i^\ddagger) \geq \omega(i)\pi^*(i)$ for all $i \in [m]$ by definition, and inequality (b) holds since π^* is obedient. A similar argument holds for $j' = i^\ddagger$. Moreover, we know that

$$U(\pi) \geq \lambda(j')\pi(j') \geq \frac{1}{m-1}\lambda(j')\pi^*(j') \overset{(a)}{\geq} \frac{1}{m-1}\frac{1}{m}\sum_{i\in[m]} \lambda(i)\pi^*(i) = \frac{1}{m(m-1)}U(\pi^*)$$

where inequality (a) holds due to the definition of state j'.

The existence of the obedient signaling scheme π constructed above implies that when the if-condition (line 3 in `LP-RP`) is satisfied if $\underline{U} < \frac{U}{m(m-1)}$. Hence, if $U \geq \frac{1}{T}$, when exploring phase I terminates, $\underline{U} \geq \frac{U(\pi^*)}{m(m-1)} \geq \frac{U(\pi^*)}{m^2}$.

Next, we argue the second part of lemma statement – when exploring phase I terminates, there exists a state $j^\ddagger \in [m]$ such that $(i^\ddagger, j^\ddagger) \in \mathcal{S}$. For each pair of states (i'', j'') such that $\omega(i'') \geq 0$, consider a direct signaling scheme $\pi^{(i'',j'')}$ with

$$\pi^{(i'',j'')}(i'') = 1 \qquad \pi^{(i'',j'')}(j'') = \lambda(j)\left(\mathbb{1}[\omega(j) \geq 0] + \min\left\{\frac{-\omega(i^\ddagger)}{\omega(j)}, 1\right\} \cdot \mathbb{1}[\omega(j) < 0]\right)$$
$$\pi^{(i'',j'')}(i) = 0 \qquad \text{for every state } i \notin \{i'', j''\}$$

Namely, $\pi^{(i'',j'')}$ is the obedient direct signaling scheme that maximizes $\lambda(j'')\pi(j'')$ when action 1 is only allowed to be recommended in state i'' or j''. By definition, among all pairs of states (i'', j''), the pair that maximizes $\lambda(j'')\pi(j'')$ must be $i'' = i^\ddagger$, which shows the second part of the lemma statement. □

Due to Lemma 14, when exploring phase I terminates, $\underline{U} \geq \frac{U(\pi^*)}{m^2}$. This enables us to build the regret bound of exploring phase I as follows.

LEMMA 15. *In* `LP-RP`*, the expected regret in exploring phase I is at most* $O(m^4)$.

Proof. Let $K_1 = -\log(U(\pi^*))$, and $K_2 = -\lceil \log\left(\frac{U(\pi^*)}{m^2}\right)\rceil$. By Lemma 14 when exploring phase I terminates, $\underline{U} \geq \frac{U(\pi^*)}{m^2}$, and thus there are at most K_2 iterations in the while loop (line 2 in `LP-RP`).

For each iteration $k \in [K]$, $\texttt{CheckObed}(\pi^I)$ is called for every pair of states (i,j) with $\underline{\mathrm{U}} \le \lambda(j)$. By Lemma 4, the total expected regret is at most

$$m^2 \cdot \sum_{k\in[K_2]} \frac{U(\pi^*)}{\sum_{i\in[m]}\lambda(i)\pi^I(i)} \overset{(a)}{\le} m^2 \cdot \sum_{k\in[K_2]} \frac{2^{-K_1}}{2^{-k}} = m^2 \cdot \sum_{k=K_1-K_2}^{K_1} 2^{-k} = O(m^4)$$

where the denominator in the right-hand side of inequality (a) is due to the construction of π^I. □

The analysis of exploring phase II.

The goal of exploring phase II is to identify an interior point $\pi^{(0)}$ for the linear program solver MembershipLP with membership oracle access. To achieve this, LP-RP excludes degenerated states which contributes little to $U(\pi^*)$, and the remaining states forms the subset $\tilde{\Theta}$. We characterize $\tilde{\Theta}$ by the following lemma.

LEMMA 16. *When exploring phase II terminates,*

- *for each state* $i \in \tilde{\Theta}$: $\delta(i)\lambda(i) \ge -mT \cdot \max_{j\in[m]}\omega(j)$;
- *for each state* $i \notin \tilde{\Theta}$: $\delta(i)\lambda(i) < -\frac{mT}{3} \cdot \max_{j\in[m]}\omega(j)$.

Proof. Let $i^\ddagger = \arg\max_{i\in[m]}\delta(i)\lambda(i)$. For each state $i \in \tilde{\Theta}$, suppose it is added into $\tilde{\Theta}$ due to pair of state $(i',j') \in \mathcal{S}$. Suppose $i' \ne j'$ (A similar argument holds for $i' = j'$). In this case, we know that the signaling scheme π^{II} corresponded to $(i',j',i,\underline{\mathrm{U}})$ is obedient, i.e.,

$$0 \le \sum_{j\in[m]} \omega(j)\pi^{\mathrm{II}}(j) \overset{(a)}{=} \omega(i') + \omega(j')\frac{\underline{\mathrm{U}}}{2\lambda(j')} + \omega(i)\frac{3}{2mT} \overset{(b)}{\le} \omega(i^\ddagger) + \frac{1}{2}\omega(i^\ddagger) + \omega(i)\frac{3}{2mT}$$

which implies that $\omega(i) \ge -mT\omega(i^\ddagger)$. Here equality (a) holds due to the construction of π^{II}, and inequality (b) holds since $\omega(i') \le \omega(i^\ddagger)$, $\omega(j') \le \omega(i^\ddagger)$, and $\underline{\mathrm{U}} \le \lambda(j')$.

By Lemma 14, there exists a state $j^\ddagger$ such that $(i^\ddagger, j^\ddagger) \in \mathcal{S}$. For each state $i \notin \tilde{\Theta}$, we know that the signaling scheme π^{II} corresponded to $(i^\ddagger, j^\ddagger, i, \underline{\mathrm{U}})$ is not obedient, i.e.,

$$0 > \sum_{j\in[m]} \omega(j)\pi^{\mathrm{II}}(j) \overset{(a)}{=} \omega(i^\ddagger) + \omega(j^\ddagger)\frac{\underline{\mathrm{U}}}{2\lambda(j^\ddagger)} + \omega(i)\frac{3}{2mT} \overset{(b)}{\ge} \omega(i^\ddagger) - \frac{1}{2}\omega(i^\ddagger) + \omega(i)\frac{3}{2mT}$$

which implies that $\omega(i) < -\frac{mT}{3}\omega(i^\ddagger)$. Here equality (a) holds due to the construction of π^{II}, and inequality (b) holds since $\omega(i^\ddagger) + \omega(j^\ddagger)\frac{\underline{\mathrm{U}}}{\lambda(j^\ddagger)} \ge 0$. □

The first part of Lemma 6 guarantees that there exists a pair of state $(i^\ddagger, j^\ddagger) \in \tilde{\Theta}$ such that the corresponding $\pi^{(0)}$ is an interior point of program $\mathcal{P}^{\mathsf{opt}}_{\tilde{\Theta}}$ (see Lemma 19). The second part of Lemma 6 guarantees that the optimal signaling scheme $\pi^\ddagger$ in program $\mathcal{P}^{\mathsf{opt}}_{\tilde{\Theta}}$ is close to the optimal signaling scheme π^* in program $\mathcal{P}^{\mathsf{opt}}$ (see Lemma 7).

LEMMA 17. *Let* $\pi^\ddagger$ *be the optimal solution in program* $\mathcal{P}^{opt}_{\tilde{\Theta}}$*, i.e.,* $\pi^\ddagger = \arg\max \mathcal{P}^{opt}_{\tilde{\Theta}}$*. Then* $U(\pi^\ddagger) \ge U(\pi^*) - O(\frac{1}{T})$.

Proof. By Lemma 1, in the optimal signaling scheme π^*, there exists a threshold state $i^\dagger \in [m]$. For each state i above $i^\dagger$, $\pi^*(i) = 1$; and for each state i below $i^\dagger$, $\pi^*(i) = 0$.

Let $i^\ddagger = \arg\max_{i\in[m]} \delta(i)\lambda(i)$. We first show that for each state i above $i^\dagger$, $i \in \pi^\ddagger$. To see this, note that π^* is obedient, i.e.,

$$0 = \sum_{j\in[m]} \omega(j)\pi^*(j) = \omega(i) + \sum_{j\in[m]\setminus\{i\}} \omega(j)\pi^*(j) \le \omega(i) + (m-1)\omega(i^\ddagger)$$

which implies that $\delta(i)\lambda(i) \ge -(m-1)\omega(i^\ddagger)$. By Lemma 6, we conclude that $i \in \tilde{\Theta}$.[32] Hence, we can now upperbound $U(\pi^*) - U(\pi^\ddagger)$ as follows,

$$U(\pi^*) - U(\pi^\ddagger) \overset{(a)}{\le} \lambda(i^\dagger)\pi^*(i^\dagger)\cdot \mathbb{1}\left[i^\dagger \notin \tilde{\Theta}\right] \le \pi^*(i^\dagger)\cdot \mathbb{1}\left[i^\dagger \notin \tilde{\Theta}\right] \overset{(b)}{\le} -\frac{(m-1)\omega(i^\ddagger)}{\omega(i^\dagger)}\cdot \mathbb{1}\left[i^\dagger \notin \tilde{\Theta}\right] \overset{(c)}{\le} O\left(\frac{1}{T}\right)$$

where inequality (a) holds since $i \in \tilde{\Theta}$ for every i such that $\pi^*(i) = 1$; inequality (c) holds due to Lemma 6; and inequality (b) holds due to the obedience of π^*, i.e.,

$$0 = \sum_{j\in[m]} \omega(j)\pi^*(j) = \omega(i^\dagger)\pi^*(i^\dagger) + \sum_{j\in[m]\setminus\{i^\dagger\}} \omega(j)\pi^*(j) \le \omega(i^\dagger)\pi^*(i^\dagger) + (m-1)\omega(i^\ddagger)$$

and $\omega(i^\dagger) < 0$ if $i^\dagger \notin \tilde{\Theta}$. □

Finally, we present the regret guarantee in exploring phase II.

LEMMA 18. *In LP-RP, the expected regret in exploring phase II is at most $O(m^5)$.*

Proof. In exploring phase II, CheckObed(π^{II}) is called for every pair of states $(i^\ddagger, j^\ddagger) \in \mathcal{S}$ and $i \in [m]\setminus\{i^\ddagger, j^\ddagger\}$. By Lemma 4, the total expected regret is at most

$$\sum_{(i^\ddagger,j^\ddagger)\in\mathcal{S}} \sum_{i\in[m]\setminus\{i^\ddagger,j^\ddagger\}} \frac{U(\pi^*)}{\sum_{j\in[m]}\lambda(j)\pi^{\text{II}}(j)} \overset{(a)}{\le} m^3 \cdot \frac{U(\pi^*)}{\frac{1}{2}\underline{U}} \overset{(b)}{\le} O(m^5)$$

where the denominator in the right-hand side of inequality (a) is due to the construction of π^{II}, and inequality (b) is due to Lemma 14. □

The analysis of exploring phase III. Let $H(\mathcal{P}^{\texttt{opt}}_{\tilde{\Theta}})$ be the convex set in program $\mathcal{P}^{\texttt{opt}}_{\tilde{\Theta}}$. Here we show that we can find an interior point $\pi^{(0)}$ for some pair of states $(i^\ddagger, j^\ddagger) \in \mathcal{S}$.

LEMMA 19 **(restatement of Lemma 8)**. *There exists a pair of state $(i^\ddagger, j^\ddagger) \in \mathcal{S}$ such that $\pi^{(0)}$ is an interior point of program $\mathcal{P}^{opt}_{\tilde{\Theta}}$. In particular, let $r = \frac{1}{16m^2T}$, then $\mathbf{B}_2(\pi^{(0)}, r) \subseteq H(\mathcal{P}^{opt}_{\tilde{\Theta}})$.*

Proof. Let $i^\ddagger = \arg\max_{i\in[m]} \delta(i)\lambda(i)$. By Lemma 14, there exists a state $j^\ddagger$ such that $(i^\ddagger, j^\ddagger) \in \mathcal{S}$. It is sufficient show that the signaling scheme $\pi^{(0)}$ corresponds to $(i^\ddagger, j^\ddagger, \tilde{\Theta})$ defined here satisfies the

[32] Here we assume $T \ge 3$.

requirement. In particular, Fix an arbitrary $\pi \in \mathbf{B}_2(\pi^{(0)}, r)$. Below, we show that every constraint in program $\mathcal{P}^{\mathsf{opt}}_{\tilde{\Theta}}$ is satisfied.

We first examine the feasibility constraint, i.e., $\pi(i) \in [0,1]$ for every $i \in \tilde{\Theta}$. For every state $i \neq j^\ddagger$, the feasibility constraint is satisfied obviously. For state $j^\ddagger$, note that $\underline{\mathrm{U}} \geq \frac{1}{m^2 T}$ and thus $\pi^{(0)}(j^\ddagger) \geq \frac{1}{m^2 T}$, which guarantees the feasibility constraint.

We next examine the constraint that $\sum_i \lambda(i)\pi(i) \geq \frac{1}{16}\underline{\mathrm{U}}$. To see this, note that

$$\sum_{i \in \tilde{\Theta}} \lambda(i)\pi(i) \geq \lambda(j^\ddagger)\pi(j^\ddagger) \geq \lambda(j^\ddagger)\left(\frac{\underline{\mathrm{U}}}{8\lambda(j^\ddagger)} - r\right) \geq \lambda(j^\ddagger)\frac{\underline{\mathrm{U}}}{16\lambda(j^\ddagger)} = \frac{1}{16}\underline{\mathrm{U}}$$

Finally, we examine the obedience constraint.

$$\begin{aligned}
\sum_{i \in [m]} \omega(i)\pi(i) &\geq \frac{1}{2}\omega(i^\ddagger) + \omega(j^\ddagger)\pi(j^\ddagger) + \sum_{i \in \tilde{\Theta}\setminus\{i^\ddagger, j^\ddagger\}} \omega(i)\pi(i) \\
&\overset{(a)}{\geq} \frac{1}{4}\omega(i^\ddagger) - \frac{\lambda(j^\ddagger)}{\underline{\mathrm{U}}}\omega(i^\ddagger)\pi(j^\ddagger) + \sum_{i \in \tilde{\Theta}\setminus\{i^\ddagger, j^\ddagger\}} \left(\omega(i^\ddagger)\frac{1}{4m} - mT \cdot \omega(i^\ddagger)\pi(i)\right) \\
&\geq \frac{1}{4}\omega(i^\ddagger) - \omega(i^\ddagger)\left(\frac{1}{8} + r\frac{\lambda(j^\ddagger)}{\underline{\mathrm{U}}}\right) + \sum_{i \in \tilde{\Theta}\setminus\{i^\ddagger, j^\ddagger\}} \left(\omega(i^\ddagger)\frac{1}{4m} - mT \cdot \omega(i^\ddagger)\left(\frac{1}{8m^2T} + r\right)\right) \\
&\overset{(b)}{\geq} 0
\end{aligned}$$

where inequality (a) holds since $\omega(i^\ddagger) + \omega(j^\ddagger)\frac{\underline{\mathrm{U}}}{\lambda(j^\ddagger)} \geq 0$, and $\omega(i) \geq -mT \cdot \omega(i^\ddagger)$ by Lemma 6; and inequality (b) holds since $r\frac{\lambda(j^\ddagger)}{\underline{\mathrm{U}}} \leq \frac{1}{16}$. □

Next, we present the regret guarantee in exploring phase III.

LEMMA 20. *In LP-RP, the expected regret in exploring phase III is at most* $O\left(m^6 \log^{O(1)}(mT)\right)$.

Proof. In exploring phase II, MembershipLP is executed for each $(i^\ddagger, j^\ddagger) \in \mathcal{S}$ where $|\mathcal{S}| \leq m^2$. Within each execution of MembershipLP, $\mathsf{CheckObed}(\pi^{\mathrm{II}})$ is called as the membership oracle. Note that we run Procedure 1 only if constraint $\sum_i \lambda(i)\pi(i) \geq \frac{1}{16}\underline{\mathrm{U}}$ is satisfied. Thus, by Lemma 4 and Lemma 14, the expected regret in exploring phase III is at most $O(m^2)$ times the total number of queries to the membership oracle in all executions. Invoking Theorem 3 finishes the proof. □

The analysis of exploiting phase.

Here we present the regret guarantee in exploiting phase.

LEMMA 21. *In LP-RP, the expected regret in exploiting phase is at most* $O(1)$.

Proof. There are three different cases. If exploring phase III terminates due to $\underline{\mathrm{U}} < \frac{1}{m^2 T}$, by Lemma 14, we know that $U(\pi^*) < \frac{1}{T}$. Hence, using any signaling scheme (including $\pi^\dagger$) induces $O(1)$ regret.

If exploring phase III terminates with $\mathcal{S} \neq \emptyset$, then linear program solver MembershipLP is executed. Recall that MembershipLP is a randomized algorithm with success probability $1 - \frac{1}{T}$. If it fails, the regret is at most T, which happens with probability $\frac{1}{T}$. If it successes, by Lemma 19, $U(\pi^\dagger) \geq U(\pi^\ddagger) - O(\frac{1}{T}) \geq U(\pi^*) - O(\frac{1}{T})$. □

Combining the regret analysis in all phases, we can prove Theorem 2.

Proof of Theorem 2. Invoking Lemma 15, Lemma 18, Lemma 20, and Lemma 21 finishes the proof. □

E. Omitted Proofs in Section 7

In this section, we present the omitted proofs in Section 7.

LEMMA 9. *Given any problem instance with binary state, for any online policy* $\mathtt{ALG}$*, there exists an online policy* $\mathtt{ALG}^\dagger$ *which only uses signaling schemes with binary signal space and has regret* $\mathbf{REG}\left[\mathtt{ALG}^\dagger\right] = \mathbf{REG}[\mathtt{ALG}]$.

Proof of Lemma 9. Fix an arbitrary problem instance I with binary state, and an arbitrary online policy ALG. Below we construct a randomized online policy $\mathtt{ALG}^\dagger$ that only uses signaling scheme with binary signal. Then, through a coupling argument, we show that users' actions under ALG and users' actions under $\mathtt{ALG}^\dagger$ are the same for each sample path, which finishes the proof.

We can consider online policy $\mathtt{ALG} : [T] \times \mathcal{H}_a \times \mathcal{H} \to \Pi$ as a mapping from the round index t, users' action history $h_a := (a_1, \ldots, a_{t-1})$ in the previous $t-1$ rounds, and the other user-irrelevant history[33] h to the signaling scheme π used by ALG in round t. Here $\mathcal{H}_a$ is the set of all possible users' action history, $\mathcal{H}$ is the set of all possible user-irrelevant history, and Π is the set of all signaling schemes.

Now we describe the construction of $\mathtt{ALG}^\dagger$ which uses mapping ALG as a blackbox.[34] We also need to define a coupling between the sample path under ALG and the sample path under $\mathtt{ALG}^\dagger$. We construct $\mathtt{ALG}^\dagger$ and its coupling with ALG inductively (i.e., round by round). Under the construction of $\mathtt{ALG}^\dagger$, together with its coupling, each user t forms the same posterior belief and takes the same action on both the sample path under ALG and the sample path under $\mathtt{ALG}^\dagger$.

We start with round 1. Let h be the user-irrelevant history under ALG. Since h is user-irrelevant, it can be simulated in $\mathtt{ALG}^\dagger$. $\mathtt{ALG}^\dagger$ first determines the signaling scheme $\pi_1 \triangleq \mathtt{ALG}(1, \emptyset, h)$ that ALG uses in round 1 given users' action history $\emptyset$ (which is empty in the beginning of round 1), and user-irrelevant history h. By Lemma 10, there exists a distribution $F_1^\dagger$ over signaling schemes with binary signal space such that the distribution of posterior belief is the same as π_1. Then $\mathtt{ALG}^\dagger$ randomly draws a

[33] For example, user-irrelevant history may encode the realized states in previous rounds and the random seed for the randomness of selecting signaling schemes in ALG.

[34] We use notation † to denote terms under constructed policy $\mathtt{ALG}^\dagger$.

signaling scheme $\pi_1^\dagger$ from distribution $F_1^\dagger$, and commits to it in round 1. By coupling state θ_1 with state $\theta_1^\dagger$, and properly coupling signal σ_1 from π_1 with signal $\sigma_1^\dagger \sim \pi_1^\dagger$ ($\sim F_1^\dagger$), we can ensure that the realized posterior belief μ_1 under ALG is the same as the realized posterior belief $\mu_1^\dagger$ under $\mathsf{ALG}^\dagger$, and thus user 1's action a_1 under ALG is the same as his $a_1^\dagger$ under $\mathsf{ALG}^\dagger$.

Suppose we have constructed $\mathsf{ALG}^\dagger$ together with its coupling for the first $t-1$ rounds. In round t, let h_a be the users' action history under ALG, and $h_a^\dagger$ be the users' action history under $\mathsf{ALG}^\dagger$. Because of the coupling in the first $t-1$ rounds, we have $h_a = h_a^\dagger$. Let h be the user-irrelevant history under ALG. Again, since h is user-irrelevant, $\mathsf{ALG}^\dagger$ can compute the distribution of h that is consistent with the users' action history $h_a^\dagger = h_a$, and sample $h^\dagger$ from this distribution. Here we couple the user-irrelevant history h under ALG with the simulated $h^\dagger$ in $\mathsf{ALG}^\dagger$, so that $h^\dagger = h$. $\mathsf{ALG}^\dagger$ first determines the signaling scheme $\pi_t \triangleq \mathsf{ALG}(t, h_a^\dagger, h^\dagger)$ that ALG uses in this round t given users' action history $h_a^\dagger$, and user-irrelevant history $h^\dagger$. The remaining construction of distribution $F_t^\dagger$ over signaling schemes with binary signal space, realized signaling scheme $\pi_t^\dagger$ and their coupling (so that $a_t^\dagger = a_t$) are the same as what we do in round 1. We omit them to avoid redundancy.

Given the construction of $\mathsf{ALG}^\dagger$ and its coupling described above, we conclude that users' actions are the same under $\mathsf{ALG}^\dagger$ and ALG, which finishes the proof. □

LEMMA 12. *Let X be a random variable with discrete support $\mathit{supp}(X)$. There exists a positive integer K, a finite set of K random variables $\{X_k\}_{k\in[K]}$, and convex combination coefficients $\mathbf{f} \in [0,1]^K$ with $\sum_{k\in[K]} f_k = 1$ such that:*

1. *Bayesian-plausibility: for each $k \in [K]$, $\mathbb{E}[X_k] = \mathbb{E}[X]$;*
2. *Binary-support: for each $k \in [K]$, the size of X_k's support is at most 2, i.e., $|\mathit{supp}(X_k)| \le 2$*
3. *Consistency: for each $x \in \mathit{supp}(X)$, $\mathit{Pr}[X = x] = \sum_{k\in[K]} f_k \cdot \mathit{Pr}[X_k = x]$*

Proof. For notation simplicity, we first introduce the following notations. We denote $\mathbb{E}[X]$ as λ, and $\mathsf{supp}(X)$ as $\mathcal{S}$. We partition $\mathcal{S}$ into $\mathcal{S}_+ \triangleq \{x_+^{(1)}, \dots, x_+^{(n_1)}\}$ and $\mathcal{S}_- \triangleq \{x_-^{(1)}, \dots, x_-^{(n_2)}\}$ where $\forall i \in [n_1]$, $x_+^{(i)} \ge \lambda$; and $\forall j \in [n_2]$, $x_-^{(j)} < \lambda$. We first show the statement holds if $\lambda \notin \mathcal{S}$. A similar argument holds for the other case where $\lambda \in \mathcal{S}$.

Now suppose $\lambda \notin \mathcal{S}$. Let $q_+^{(i)}$ denote $\Pr\left[X = x_+^{(i)}\right]$ and $q_-^{(j)}$ denote $\Pr\left[X = x_-^{(j)}\right]$. Consider the following linear system with variable $\{f_{ij}\}_{i\in[n_1], j\in[n_2]}$:

$$\begin{cases} \sum_{j\in[n_2]} \frac{\lambda - x_-^{(j)}}{x_+^{(i)} - x_-^{(j)}} f_{ij} = q_+^{(i)} & \forall i \in [n_1] \\ \sum_{i\in[n_1]} \frac{x_+^{(i)} - \lambda}{x_+^{(i)} - x_-^{(j)}} f_{ij} = q_-^{(j)} & \forall j \in [n_2] \\ f_{ij} \ge 0 & \forall i \in [n_1], j \in [n_2] \end{cases} \tag{2}$$

Below we first show how to construct random variable $\{X_k\}_{k\in[K]}$ required in lemma statement with any feasible solution in linear system (2) as the convex combination coefficients. Then we show the existence of the feasible solution in linear system (2).

Fix a feasible solution $\{f_{ij}\}_{i\in[n_1],j\in[n_2]}$ in linear system (2). Let $K = n_1 \cdot n_2$. Consider the set of random variables $\{X_{ij}\}_{i\in[n_1],j\in[n_2]}$ as follows,

$$X_{ij} = \begin{cases} x_+^{(i)} \text{ with probability } \frac{\lambda - x_-^{(j)}}{x_+^{(i)} - x_-^{(j)}} \\ x_-^{(j)} \text{ otherwise} \end{cases}$$

Note that $\{f_{ij}\}$ is valid convex combination coefficient for $\{X_{ij}\}_{i\in[n_1],j\in[n_2]}$. In particular,

$$\sum_{i\in[n_1]}\sum_{j\in[n_2]} f_{ij} = \sum_{i\in[n_1]}\sum_{j\in[n_2]} f_{ij}\left(\frac{\lambda - x_-^{(j)}}{x_+^{(i)} - x_-^{(j)}} + \frac{x_+^{(i)} - \lambda}{x_+^{(i)} - x_-^{(j)}}\right) = \sum_{i\in[n_1]} q_+^{(i)} + \sum_{j\in[n_2]} q_-^{(j)} = 1$$

To see why random variables $\{X_{ij}\}_{i\in[n_1],j\in[n_2]}$ with convex combination coefficient $\{f_{ij}\}_{i\in[n_1],j\in[n_2]}$ satisfy the statement requirement, note that "Bayesian-plausibility" property and "Binary-support" property are satisfied by construction. To verify "Consistency" property, consider each $x_+^{(i)} \in \mathcal{S}_+$,

$$\begin{aligned}\sum_{i\in[n_1]}\sum_{j\in[n_2]} f_{ij}\cdot \Pr\left[X_{ij} = x_+^{(i)}\right] &= \sum_{j\in[n_2]} f_{ij}\cdot \Pr\left[X_{ij} = x_+^{(i)}\right] \\ &= \sum_{j\in[n_2]} f_{ij}\cdot \frac{\lambda - x_-^{(j)}}{x_+^{(i)} - x_-^{(j)}} = q_+^{(i)} = \Pr\left[X = x_+^{(i)}\right]\end{aligned}$$

Similarly, for each $x_-^{(j)} \in \mathcal{S}_-$,

$$\begin{aligned}\sum_{i\in[n_1]}\sum_{j\in[n_2]} f_{ij}\cdot \Pr\left[X_{ij} = x_-^{(j)}\right] &= \sum_{i\in[n_1]} f_{ij}\cdot \Pr\left[X_{ij} = x_-^{(j)}\right] \\ &= \sum_{i\in[n_1]} f_{ij}\cdot \frac{x_+^{(i)} - \lambda}{x_+^{(i)} - x_-^{(j)}} = q_-^{(j)} = \Pr\left[X = x_-^{(j)}\right]\end{aligned}$$

Hence, we conclude that random variables $\{X_{ij}\}_{i\in[n_1],j\in[n_2]}$ with convex combination coefficient $\{f_{ij}\}_{i\in[n_1],j\in[n_2]}$ satisfy the statement requirement.

Next, we show the existence of feasible solution $\{f_{ij}\}_{i\in[n_1],j\in[n_2]}$ in linear system (2). Let $\hat{f}_{ij} = \frac{f_{ij}}{x_+^{(i)} - x_-^{(j)}}$. It is equivalent to show that

$$\begin{cases} \sum_{j\in[n_2]} (\lambda - x_-^{(j)})\hat{f}_{ij} = q_+^{(i)} & \forall i \in [n_1] \\ \sum_{i\in[n_1]} (x_+^{(i)} - \lambda)\hat{f}_{ij} = q_-^{(j)} & \forall j \in [n_2] \\ f_{ij} \geq 0 & \forall i \in [n_1], j \in [n_2] \end{cases} \tag{3}$$

has a feasible solution. We show this by an induction argument.

Induction Hypothesis. Fix any positive integer $n_1, n_2 \in \mathbb{N}_{\geq 1}$, and arbitrary non-negative numbers [35] $\{x_+^{(i)}, q_+^{(i)}\}_{i\in[n_1]}$, $\{x_-^{(j)}, q_-^{(j)}\}_{i\in[n_2]}$, and λ. If $x_+^{(i)} > \lambda$ for all $i \in [n_1]$, $x_-^{(j)} < \lambda$ for all $j \in [n_2]$, and

[35] Here we do not assume that $\sum_{i\in[n_1]} q_+^{(i)} + \sum_{j\in[n_2]} q_-^{(j)} = 1$.

$\sum_{i\in[n_1]} x_+^{(i)} q_+^{(i)} + \sum_{j\in[n_2]} x_-^{(j)} q_-^{(j)} = \left(\sum_{i\in[n_1]} q_+^{(i)} + \sum_{j\in[n_2]} q_-^{(j)}\right)\lambda$; then linear system (3) has a feasible solution.

Base Case ($n_1 = 1$ or $n_2 = 1$). Here we show the induction hypothesis holds for $n_1 = 1$. A similar argument holds for $n_2 = 1$. Consider a solution $\{\hat{f}_{1j}\}_{j\in[n_2]}$ constructed as follows,

$$\hat{f}_{1j} = \frac{q_-^{(j)}}{x_+^{(1)} - \lambda}$$

It is obvious that $\{\hat{f}_{ij}\}$ is non-negative and the equality for every $j \in [n_2]$ is satisfied. Now consider the equality for $i = 1$. Note that

$$\sum_{j\in[n_2]} (\lambda - x_-^{(j)})\hat{f}_{1j} = \sum_{j\in[n_2]} \frac{(\lambda - x_-^{(j)})q_-^{(j)}}{x_+^{(1)} - \lambda} = \frac{\sum_{j\in[n_2]}(\lambda - x_-^{(j)})q_-^{(j)}}{x_+^{(1)} - \lambda} \overset{(a)}{=} \frac{(x_+^{(1)} - \lambda)q_+^{(1)}}{x_+^{(1)} - \lambda} = q_+^{(1)}$$

where equality (a) uses the assumption that $x_+^{(1)} q_+^{(1)} + \sum_{j\in[n_2]} x_-^{(j)} q_-^{(j)} = \left(q_+^{(1)} + \sum_{j\in[n_2]} q_-^{(j)}\right)\lambda$.

Inductive Step ($n_1 \geq 2$ and $n_2 \geq 2$). Here we show the induction hypothesis holds for $n_1 \geq 2$ and $n_2 \geq 2$. In addition, we assume that $x_+^{(n_1)} q_+^{(n_1)} + x_-^{(n_2)} q_-^{(n_2)} \geq (q_+^{(n_1)} + q_-^{(n_2)})\lambda$. A similar argument holds for $x_+^{(n_1)} q_+^{(n_1)} + x_-^{(n_2)} q_-^{(n_2)} < (q_+^{(n_1)} + q_-^{(n_2)})\lambda$. Below we show that there exists a feasible solution where we fix

$$\forall i \in [n_1 - 1]:\ \hat{f}_{i n_2} = 0 \qquad \text{and} \qquad \hat{f}_{n_1 n_2} = \frac{q_-^{(n_2)}}{x_+^{(n_1)} - \lambda}$$

To see this, observe that the equality for $j = n_2$ is satisfied. Next, we invoke the induction hypothesis on instance $(n_1, n_2 - 1, x_+^{(1)}, q_+^{(1)}, \ldots, x_+^{(n_1-1)}, q_+^{(n_1-1)}, x_+^{(n_1)}, \tilde{q}_+^{(n_1)}, x_-^{(1)}, q_-^{(1)}, \ldots, x_-^{(n_2-1)}, q_-^{(n_2-1)}, \lambda)$ where $\tilde{q}_+^{(n_1)} = q_+^{(n_1)} - \hat{f}_{n_1 n_2}(\lambda - x_-^{(n_2)})$. It is sufficient to show that this instance satisfies the assumption in the induction hypothesis. In particular, we can verify that

$$\tilde{q}_+^{(n_1)} = q_+^{(n_1)} - \hat{f}_{n_1 n_2}(\lambda - x_-^{(n_2)}) = q_+^{(n_1)} - \frac{q_-^{(n_2)}(\lambda - x_-^{(n_2)})}{x_+^{(n_1)} - \lambda} \geq 0$$

since we assume $x_+^{(n_1)} q_+^{(n_1)} + x_-^{(n_2)} q_-^{(n_2)} \geq (q_+^{(n_1)} + q_-^{(n_2)})\lambda$; and

$$\begin{aligned}
&\sum_{i\in[n_1-1]} x_+^{(i)} \cdot q_+^{(i)} + x_+^{(n_1)} \cdot \tilde{q}_+^{(n_1)} + \sum_{j\in[n_2-1]} x_-^{(j)} \cdot q_-^{(j)} \\
&= \sum_{i\in[n_1]} x_+^{(i)} \cdot q_+^{(i)} + \sum_{j\in[n_2]} x_-^{(j)} \cdot q_-^{(j)} - \left(x_+^{(n_1)}(q_+^{(n_1)} - \tilde{q}_+^{(n_1)}) + x_-^{(n_2)} q_-^{(n_2)}\right) \\
&= \sum_{i\in[n_1]} x_+^{(i)} \cdot q_+^{(i)} + \sum_{j\in[n_2]} x_-^{(j)} \cdot q_-^{(j)} - \left(x_+^{(n_1)} \frac{q_-^{(n_2)}(\lambda - x_-^{(n_2)})}{x_+^{(n_1)} - \lambda} + x_-^{(n_2)} q_-^{(n_2)}\right) \\
&\overset{(a)}{=} \left(\sum_{i\in[n_1]} q_+^{(i)} + \sum_{j\in[n_2]} q_-^{(j)}\right)\lambda - \left(x_+^{(n_1)} \frac{q_-^{(n_2)}(\lambda - x_-^{(n_2)})}{x_+^{(n_1)} - \lambda} + x_-^{(n_2)} q_-^{(n_2)}\right)
\end{aligned}$$

$$\overset{(b)}{=} \left(\sum_{i\in[n_1]} q_+^{(i)} + \sum_{j\in[n_2]} q_-^{(j)} \right) \lambda - \left(\frac{q_-^{(n_2)}(\lambda - x_-^{(n_2)})}{x_+^{(n_1)} - \lambda} + q_-^{(n_2)} \right) \lambda$$
$$= \left(\sum_{i\in[n_1-1]} q_+^{(i)} + \tilde{q}_+^{(n_1)} + \sum_{j\in[n_2-1]} q_-^{(j)} \right) \lambda$$

where equality (a) uses the assumption that

$$\sum_{i\in[n_1]} x_+^{(i)} q_+^{(i)} + \sum_{j\in[n_2]} x_-^{(j)} q_-^{(j)} = \left(\sum_{i\in[n_1]} q_+^{(i)} + \sum_{j\in[n_2]} q_-^{(j)} \right) \lambda$$

and equality (b) is by algebra. □

LEMMA 13. *For every single-item dynamic pricing problem instance I, there exists an online Bayesian recommendation problem instance $I^\dagger$ with binary state. For every online policy $\mathit{ALG}^\dagger$ with binary signal space and regret $\mathbf{REG}_{I^\dagger}\left[\mathit{ALG}^\dagger\right]$ on online Bayesian recommendation instance $I^\dagger$, there exists a dynamic pricing mechanism ALG with regret $\mathbf{REG}_I[\mathit{ALG}] \le \mathbf{REG}_{I^\dagger}\left[\mathit{ALG}^\dagger\right] + 1$ on dynamic pricing instance I.*

Proof of Lemma 13. Fix an arbitrary single-item dynamic pricing problem instance $I = (T, v^*)$ such that there are T rounds and each buyer has private value v^*. Without loss of generality, we assume that $v^* \le \frac{1}{2}$ and $T \ge 2$. We first present the construction of the Bayesian recommendation problem instance $I^\dagger$. Then, given any online policy $\mathsf{ALG}^\dagger$ with binary signal space for the Bayesian recommendation instance $I^\dagger$, we present the construction of dynamic pricing mechanism ALG for the original dynamic pricing instance I with regret $\mathbf{REG}_I[\mathsf{ALG}] \le \mathbf{REG}_{I^\dagger}\left[\mathsf{ALG}^\dagger\right] + 1$.

Construction of the Bayesian recommendation instance. Consider the following Bayesian recommendation instance $I^\dagger$.[36] There are $m^\dagger = 2$ states, and $T^\dagger = T$ rounds. Let $\epsilon = 1/T^\dagger$. State 1 is realized with probability $\lambda^\dagger(1) = \epsilon$ and state 2 is realized with probability $\lambda^\dagger(2) = 1 - \epsilon$. The users' utility is defined as follow,

$$\text{for state 1:} \quad \rho^\dagger(1, a^\dagger) = \mathbb{1}\left[a^\dagger = 1\right]$$
$$\text{for state 2:} \quad \rho^\dagger(2, a^\dagger) = -\frac{\epsilon}{v^*} \cdot \mathbb{1}\left[a^\dagger = 1\right]$$

For notation simplicity, in this section, we use $\omega(i)$ to denote $\delta(i)\lambda(i)$. By construction, $\omega^\dagger(1) = \epsilon$, $\omega^\dagger(2) = -\frac{\epsilon(1-\epsilon)}{v^*}$, and the optimal signaling in hindsight $\pi^{*\dagger}$ satisfies that $\pi^{*\dagger}(1) = 1$, $\pi^{*\dagger}(2) = \frac{v^*}{1-\epsilon}$, and $U(\pi^{*\dagger}) = v^* + \epsilon$.

Construction of the dynamic pricing mechanism. Fix an arbitrary online policy $\mathsf{ALG}^\dagger$ with binary signal space for the Bayesian recommendation instance $I^\dagger$. Here we show that there exists a

[36] We use notation † and ‡ to denote the Bayesian recommendation instance.

dynamic pricing mechanism ALG with regret $\mathbf{REG}_I[\mathsf{ALG}] \le \mathbf{REG}_{I^\dagger}\left[\mathsf{ALG}^\dagger\right] + 1$. Our argument contains two steps. First, we show that every online policy $\mathsf{ALG}^\dagger$ can be converted into an online policy $\mathsf{ALG}^\ddagger$ within a subclass, which has weakly smaller regret. Then, we show how to construct a dynamic pricing mechanism ALG based on online policy $\mathsf{ALG}^\ddagger$.

• **[Step I]** Notice that in the construction of the Bayesian recommendation instance $I^\dagger$, users prefer action 1 in state 1 and action 0 in state 2. Thus, in the optimal signaling scheme in hindsight $\pi^{*\dagger}$, the threshold state is state 2 and state 1 is above state 2. Intuitively speaking, it is natural to consider a subclass of signaling schemes $\Pi^\ddagger$ such that for every signaling scheme $\pi^\ddagger \in \Pi^\ddagger$, it issues signal $\sigma^\ddagger = 1$ (i.e., recommends action 1) deterministically (i.e., $\pi^\ddagger(1) = 1$) when the state is 1, and issues signal $\sigma^\ddagger = 1$ with probability $\pi^\ddagger(2)$ when the state is 2. Following this intuition, below we show that for any online policy $\mathsf{ALG}^\dagger$ (with binary signal space), there exists an online policy $\mathsf{ALG}^\ddagger$ which only uses signaling schemes in $\Pi^\ddagger$ and achieves weakly smaller regret.

Suppose in round t, signaling scheme $\pi_t^\dagger$ is used in $\mathsf{ALG}^\dagger$. Recall $\mu_t^\dagger(\sigma^\dagger, i)$ is the posterior probability $\Pr\left[\theta_t^\dagger = i \mid \sigma^\dagger\right]$ under $\pi_t^\dagger$. Without loss of generality, we assume that $\mu^\dagger(1,2) \le \lambda^\dagger(2) \le \mu^\dagger(0,2)$. Since $\omega^\dagger(1) + \omega^\dagger(2) < 0$, user t must take action 0 when the realized signal $\sigma^\dagger = 0$. Thus, the regret under $\pi_t^\dagger$ is

$$\begin{aligned}\mathbf{REG}_{I^\dagger}\left[\pi_t^\dagger\right] &= U(\pi^{*\dagger}) - (\lambda^\dagger(1)\pi_t^\dagger(1) + \lambda^\dagger(2)\pi_t^\dagger(2)) \cdot \mathbb{1}\left[\text{user } t \text{ takes action 1} \mid \sigma^\dagger = 1\right] \\ &= U(\pi^{*\dagger}) - (\lambda^\dagger(1)\pi_t^\dagger(1) + \lambda^\dagger(2)\pi_t^\dagger(2)) \cdot \mathbb{1}\left[\text{user } t \text{ takes action 1} \mid \text{posterior belief is } \mu^\dagger(1,\cdot)\right]\end{aligned}$$

Thus, in round t, $\mathsf{ALG}^\ddagger$ can use signaling scheme $\pi_t^\ddagger$ such that $\pi_t^\ddagger(1) \triangleq 1$ and $\pi_t^\ddagger(2) \triangleq \pi_t^\dagger(2)/\pi_t^\dagger(1)$. Since $\mu^\dagger(1,2) \le \lambda^\dagger(2)$ and $\mu^\dagger(1,2) = (\lambda^\dagger(2)\pi_t^\dagger(2))/(\lambda^\dagger(1)\pi_t^\dagger(1) + \lambda^\dagger(2)\pi_t^\dagger(2))$, we have $\pi_t^\ddagger(2)$ $(= \pi_t^\dagger(2)/\pi_t^\dagger(1)) \le 1$ and thus $\pi_t^\ddagger$ is well-defined. Additionally, by construction, posterior belief $\mu_t^\ddagger(1,\cdot)$ under signal 1 in $\pi_t^\ddagger$ is the same as posterior belief $\mu_t^\dagger(1,\cdot)$ under signal 1 in $\pi_t^\dagger$. Thus,

$$\begin{aligned}\mathbf{REG}_{I^\dagger}\left[\pi_t^\ddagger\right] &= U(\pi^{*\dagger}) - (\lambda^\dagger(1)\pi_t^\ddagger(1) + \lambda^\dagger(2)\pi_t^\ddagger(2)) \cdot \mathbb{1}\left[\text{user } t \text{ takes action 1} \mid \sigma^\ddagger = 1\right] \\ &= U(\pi^{*\dagger}) - (\lambda^\dagger(1)\pi_t^\ddagger(1) + \lambda^\dagger(2)\pi_t^\ddagger(2)) \cdot \mathbb{1}\left[\text{user } t \text{ takes action 1} \mid \text{posterior belief is } \mu^\ddagger(1,\cdot)\right] \\ &\le \mathbf{REG}_{I^\dagger}\left[\pi_t^\dagger\right]\end{aligned}$$

Therefore, suppose $\mathsf{ALG}^\dagger$ uses signaling scheme $\pi_t^\dagger$ in round t. $\mathsf{ALG}^\ddagger$ can mimic $\mathsf{ALG}^\dagger$ by using signaling scheme $\pi_t^\ddagger$ defined above and suffers a weakly smaller expected regret. Though posterior belief $\mu_t^\ddagger(0,\cdot)$ under signal 0 in $\pi_t^\ddagger$ may not equal to posterior belief $\mu_t^\dagger(0,\cdot)$ under signal 0 in $\pi_t^\dagger$, user t must take action 0 under both $\mu_t^\ddagger(0,\cdot)$ and $\mu_t^\dagger(0,\cdot)$. Thus, $\mathsf{ALG}^\ddagger$ has more information than $\mathsf{ALG}^\dagger$ and can continue mimicking $\mathsf{ALG}^\dagger$ in the future rounds.

• **[Step II]** Here we show that given any online policy $\mathsf{ALG}^\ddagger$ which only uses signaling scheme in $\Pi^\ddagger$ defined in [Step I], we can construct a dynamic pricing mechanism ALG with regret $\mathbf{REG}_I[\mathsf{ALG}] \le \mathbf{REG}_{I^\dagger}\left[\mathsf{ALG}^\ddagger\right] + 1$.

Suppose in round t, signaling scheme $\pi_t^\ddagger$ is used in $\mathsf{ALG}^\ddagger$. By the definition of $\mathsf{ALG}^\ddagger$, we know that $\pi_t^\ddagger(1)=1$. Thus, user t takes action 1 if and only if the realized signal $\sigma^\ddagger=1$ and his expected utility of taking action 1 is better than taking action 0 under his posterior belief, i.e.,

$$\omega^\dagger(1)\,\pi_t^\ddagger(1)+\omega^\dagger(2)\,\pi_t^\ddagger(2)\geq 0 \quad \Rightarrow \quad \pi_t^\ddagger(2)\leq \frac{v^*}{1-\epsilon}$$

Hence, the expected regret induced by signaling scheme $\pi_t^\ddagger$ is

$$\begin{aligned}\mathbf{REG}_{I^\dagger}\left[\pi_t^\ddagger\right] &= U(\pi^{*\dagger})-(\lambda^\dagger(1)\pi_t^\ddagger(1)+\lambda^\dagger(2)\pi_t^\ddagger(2))\cdot\mathbb{1}\left[\text{user } t \text{ takes action } 1 \mid \sigma^\ddagger=1\right]\\ &= v^*+\epsilon-\left(\epsilon+(1-\epsilon)\pi_t^\ddagger(2)\right)\cdot\mathbb{1}\left[\pi_t^\ddagger(2)\leq\frac{v^*}{1-\epsilon}\right]\end{aligned}$$

Therefore, suppose online policy $\mathsf{ALG}^\ddagger$ uses signaling scheme $\pi_t^\ddagger$ in round t for the Bayesian recommendation instance $I^\dagger$. We can construct the following dynamic pricing mechanism ALG which posts price $p_t\triangleq(1-\epsilon)\pi_t^\ddagger(2)$ in round t for the dynamic pricing instance I. The regret of posting price p_t is

$$\mathbf{REG}_I[p_t]=v^*-p_t\cdot\mathbb{1}[p_t\leq v^*]\leq\mathbf{REG}_{I^\dagger}\left[\pi_t^\ddagger\right]+\epsilon$$

Since dynamic pricing mechanism ALG has more information than online policy $\mathsf{ALG}^\ddagger$,[37] ALG can simulate $\mathsf{ALG}^\ddagger$ in the future rounds. The total regret is

$$\mathbf{REG}_I[\mathsf{ALG}]-\mathbf{REG}_{I^\dagger}\left[\mathsf{ALG}^\ddagger\right]=\sum_{t\in[T]}\left(\mathbf{REG}_I[p_t]-\mathbf{REG}_{I^\dagger}\left[\pi_t^\ddagger\right]\right)\leq\epsilon\cdot T=1$$

which concludes the proof. □

[37] In particular, dynamic pricing mechanism deterministically learns whether $p_t\leq v^*$ (a.k.a., $\mathbb{1}\left[\pi_t^\ddagger(2)\leq\frac{v^*}{1-\epsilon}\right]$), while online policy $\mathsf{ALG}^\ddagger$ only learns this information when signal 1 is realized.