Human-Centered Machine Learning: Algorithm Design and Human Behavior

Wei Tang

Ph.D. Dissertation. 2022

WASHINGTON UNIVERSITY IN ST. LOUIS

McKelvey School of Engineering
Department of Computer Science and Engineering

Dissertation Examination Committee:
Chien-Ju Ho, Chair
Yiling Chen
Brendan Juba
Yevgeniy Vorobeychik
William Yeoh

Human-Centered Machine Learning: Algorithm Design and Human Behavior

by
Wei Tang

A dissertation presented to
the McKelvey School of Engineering
of Washington University in
partial fulfillment of the
requirements for the degree
of Doctor of Philosophy

August 2022
St. Louis, Missouri

Table of Contents

List of Figures

List of Tables

Acknowledgments

I am indebted to many people for their support throughout my PhD studies.

First and foremost, I must begin by thanking my advisor Chien-Ju (CJ) Ho. CJ is the best advisor that I could ever wish for. CJ has been giving me endless freedom to explore my research interests and has been patiently guiding me to develop and shape my own research style. CJ has been a constant source of invaluable advice and resource to me in every aspect of research. Whenever I encounter any problems or challenges, CJ is always there to provide me help, always believing in me, always encouraging me, and giving me the confidence to be an independent researcher. Above all, CJ made me feel many times I am not chatting with my advisor, but with a closest friend. I am extremely grateful to have CJ as my advisor.

I would like to thank other committee members: Yiling Chen, Brendan Juba, Yevgeniy Vorobeychik, and William Yeoh. They have provided me with invaluable suggestions in shaping my thoughts about this thesis. I want to especially thank Yiling for her help and suggestions in my job search. I want to thank Brendan and Eugen for their suggestions and comments which made me think about a more coherent story behind this thesis and pointed out many interesting future research directions. I want to thank Will, who has been providing me with so many valuable suggestions and feedback to help me improve my presentation.

In addition, I have been very privileged to collaborate with and learn from many other great researchers. Ming Yin brings me to the area of behavioral experiments, which has inspired me a lot in shaping my own research agenda. Yang Liu and we together have been exploring many interesting problems, and he is always responsive whenever I have questions, and can always provide me with technique tools to solve these problems. Thank Bolin Ding for accepting me as a research intern working with him, and this intern experience has led me to the area of information design. I met Yiding Feng in 2018 at my first time attending EC. Yiding is among the brightest minds that I know and collaborating with Yiding in my last two years of my PhD studies has greatly shaped my thoughts to formulate a research problem, think about the problem, and solve the problem. I am also deeply grateful for his continuous support and advice on different aspects of my research career, including pursuing the clarity and rigorousness in writing, presenting, and speaking. I learned a lot from working with Haifeng Xu. Working on some theoretical problems, Haifeng would always impress me with his incredible mathematical intuitions and has shown me the importance of how to think about problems from a different perspective. Thank my labmates: Saumik Narayanan, and Guanghui Yu for teaching me coding skills and thank Yatong Chen, Sixie Yu for together exploring interesting questions.

Many thansk to my friends in St. Louis. Especially my cohort: Zhihao Xia, Bei Wu, Xiaojian Xu, Kyle Singer, Jeffery Jung, Aaron Park, Ruixuan Dai, Tiantian Zhu, Yu Sun, etc. My friends from senior years: Haipeng Dai, Shali Jiang, Dingwen Li, Zhuoshu Li, Yifan Xu, Hao Yan, Huayi Zeng, Muhan Zhang, Liang Zhou, etc. I will always remember the good time we spent in Jolley Hall.

Finally, thank my mom, my dad, and my sister for always being there for their unconditional and endless love. Thank my beloved wife, Rusi Yan, for always trusting me and supporting me. You are clearly the best thing I find in my PhD.

Wei Tang

Washington University in St. Louis

August 2022

Dedicated to my parents, my sister.

ABSTRACT OF THE DISSERTATION

Human-Centered Machine Learning: Algorithm Design and Human Behavior

by

Wei Tang

Doctor of Philosophy in Computer Science

Washington University in St. Louis, 2022

Professor Chien-Ju Ho, Chair

Machine learning is increasingly engaged in a large number of important daily decisions and has great potential to reshape various sectors of our modern society. To fully realize this potential, it is important to understand the role that humans play in the design of machine learning algorithms and investigate the impacts of the algorithm on humans.

Towards the understanding of such interactions between humans and algorithms, this dissertation takes a human-centric perspective and focuses on investigating the interplay between human behavior and algorithm design. Accounting for the roles of humans in algorithm design creates unique challenges. For example, humans might be strategic or exhibit behavioral biases when generating data or responding to algorithms, violating the standard independence assumption in algorithm design. How do we design algorithms that take such human behavior into account? Moreover, humans possess various ethical values, e.g., humans want to be treated fairly and care about privacy. How do we design algorithms that align with human values? My dissertation addresses these challenges by combining both theoretical and empirical approaches. From the theoretical perspective, we explore how to design algorithms that account for human behavior and respect human values. In particular, we formulate models of human behavior in the data generation process and design algorithms that can leverage data with human biases. Moreover, we investigate the long-term impacts of algorithm decisions and

design algorithms that mitigate the reinforcement of existing inequalities. From the empirical perspective, we have conducted behavioral experiments to understand human behavior in the context of data generation and information design. We have further developed more realistic human models based on empirical data and studied the algorithm design building on the updated behavior models.

Chapter 1

Introduction

Machine learning is increasingly engaged in a large number of important daily decisions and has great potential to reshape various sectors of our modern society. To fully realize this potential, it is important to understand the role that humans play in the design of machine learning algorithms and investigate the impacts of the algorithm on humans.

To provide more motivation, when there exists a sufficient amount of data, machine learning algorithms can often discover the patterns within the data and make predictions accordingly. However, in many cases, such data is obtained from humans either directly or indirectly. For example, researchers have leveraged workers in crowdsourcing markets to annotate data. In online services, human feedback (such as upvotes or views) is utilized to learn whether a product or service is meeting customers' needs. To efficiently solve problems that humans are involved in, we need to design proper algorithms that address the human components in the process. On the other hand, the decisions made by algorithms could also impact human welfare. For example, a discriminatory machine learning algorithm can negatively affect the well-being of minority communities. Therefore, to advance the deployment of machine

learning algorithms or applications, it is important to understand any potential impact of it on both individual and societal levels.

This dissertation advances the understanding of human-centered machine learning through both theoretical and empirical approaches. In particular, this dissertation answers the questions of how humans would impact algorithm design in machine learning, and how algorithms could be designed to align with human values.

1.1 Algorithm Design: Accounting for Human Behavior

One major theme of this dissertation has focused on how to design learning algorithms from human behavior. For example, many works often assume data collected from human are independent, and even identically distributed, which is usually violated especially when humans are involved in to generate the data in need. On the other hand, full knowledge of how human respond to the algorithm's output are often required to get tractable analysis, which is also not realistic in complex environments. To solve the learning problems that humans are involved in, we need to design proper algorithms that address the human components in the process.

Learning from Biased Human Feedback. I first studied how to include human biased behavior in online learning frameworks. User-generated content platform usually relies on user feedback (e.g., number of likes, upvotes, etc) to learn the content qualities so that to select the best content to display to users. Empirical studies show that humans exhibit a tendency to agree with the majority opinion even if their personal opinion disagrees. In [173], I modeled the platform's learning problem as a multi-armed bandit problem, with arm representing a content. When an arm is played, (present such content to a user), a user receives a realized reward drawn from the distribution of the arm. She then provides a

biased feedback of the realized reward, that depends on both the realized reward and the feedback history of the arm. Differing from the standard bandit problem where the learner (i.e., platform) can directly observe the realized reward of an arm, in our setting, the learner can only observe user's biased feedback on the realized reward.

The goal of the learner is to design a strategy to sequentially choose arms to maximize the total rewards users receive while only having access to the biased user feedback. The challenge is that the learner can observe only the biased feedback but not the realized rewards. I explored two natural feedback models, one is that user feedback is biased only by the arm's average feedback, and one is biased by both the average feedback and the number of arm's collected feedback. Under this behavioral model, I showed it is possible for platform to learn an optimal policy. However, in the other model when user feedback is biased by both the average feedback and the number of feedback instances, I proved there exists no efficient algorithm for the platform to learn the arm's quality if there's no intervention taken on how to collect human-generated feedback. The results demonstrate the importance of understanding human behavior in algorithm design. A small deviation on the user behavior model and/or the design of the information structure could have significant impacts on the overall system outcome. Therefore, platforms and decision makers should carefully take these into account when designing learning algorithms in systems with humans in the loop.

Robust Learning from Uncertain Human Behavior. One ignored issue in previous research is the need of the *full knowledge* of human behavior model. A *transparent* algorithm allows humans to verify the data and trace the steps that led to a specific decision and, thereby, to allow human discretion to change algorithmic decisions that cause undesirable outcomes by manipulating actionable data they have access to. In response to this "gaming" behavior, there has been a recent flurry of work in studying decision-making under strategic behavior. To make the analysis tractable, many current works explicitly assume decision-maker's full

knowledge of humans' action space and the corresponding costs for manipulating the features. To relax this assumption, I studied the design of *robust* optimal decision rules with strategic agent [177]. I defined the robustness as used in robust contract design in economics. I first showed that under mild conditions, for any robust optimal decision rule, there exists a linear one that is equally robust optimal. I then explored the computational problem of searching for the robust optimal decision rule. By leveraging techniques from distributionally robust optimization, our results inform efficient algorithms for searching the robust optimal one especially in settings when non-robust strategic decision-making problem is efficiently solvable.

1.2 Algorithm Design: Aligning with Human Values

The previous discussions have examined how human would impact algorithm design, through the lens of human behavior. On the other hand, the consequential decisions output from an algorithm will, in turn, induce complex social dynamics by changing human outcomes. Furthermore, the algorithm's process to output such consequential decisions usually rely on personal information provided by the participants. If not handled carefully, a data breach over these information can have potentially disastrous consequences. Here, I describe several examples of this research where I first draw on how the consequential outcomes from the algorithms would have the impact on humans in repeated decision-making environment, and then I discuss the information-leakage issue in a general decision-making framework.

Consequential Outcome Impact of Repeated Decision-making. I examined the long-term impact of actions informed by the consequential decisions [175]. These long-term impacts of actions often came up when the well-being of the people is involved. Consider following concerns: If being insensitive with the long-term impact of actions, the decision

maker may risk treating a historically disadvantaged group unfairly. Making things even worse, these unfair and oblivious decisions might reinforce existing biases and make it harder to observe the true potential for a disadvantaged group. To formulate the above problem, I generalized the multi-armed bandit setting by introducing the *impact functions* that encode the dependency of the "bias" due to the action history of the learning to the arm rewards. This history-dependency structure of observed rewards makes the problem substantially more challenging. In particular, I first showed that applying standard bandit algorithms leads to linear regret, i.e., existing approaches will obtain low rewards with a biased learning process. I then demonstrated that, under relatively mild conditions, efficient algorithms with theoretical guarantees for solving this problem are possible.

Privacy-preserving in Sequential Decision-making. I consider the following sequential learning framework: A *learner* makes sequential decisions with online arriving *agents*. The agent gives feedback based on the action, and the learner obtains utility based on the agent's feedback. The learner aims to maximize her reward, which can be defined either as the cumulative reward over time or the reward based on the learning outcome. The above sequential learning framework is general and covers a wide range of real-world applications. For example, the online sellers price their products using buyers' information; and in federated learning, the learner aims to optimize the parameters of their learning models using gradient decent where the gradient information comes from data-holding users. However, this framework bears potential privacy-leakage issues on both user's end and learner's end. I first studied privacy leakage on agent's (i.e., user's) end in the contextual dynamic pricing setting [176]. By adopting differential privacy as privacy measure, I explored the design of differentially private pricing algorithms that minimize the *regret* w.r.t the oracle policy that knows the distribution of users' preferences, while satisfying a pre-defined privacy guarantee. I proposed a differentially private algorithm that achieves sublinear regret. To complement this line of

research, I then studied privacy preserving on learner's end [178]. In particular, I studied the *secure* stochastic convex optimization, in which the learner aims to optimize the *accuracy*, i.e., obtain an accurate estimate to the optimal point, while securing her *privacy*, i.e., preventing an adversary from inferring what she learned. I formalized the notions of accuracy and privacy using probably approximately correct style notions and provided lower/upper bounds characterizations of the query complexity for this secure learning problem.

1.3 Human Behavior Modeling via Behavioral Experiments

In addition to working with standard human behavioral models, I have also conducted behavioral experiments to better understand human behavior in the context of AI-assisted decision making and communication in crowdsourcing tasks

Human Behavior Modeling – Bayesian Rationality in Information Design. Modern AI technologies have increasingly enabled many machine-assisted decision making. To give a few examples, online recommendation systems encompass a class of techniques and (machine learning) algorithms which are able to suggest "relevant" items to users. Users then make their decision on which video to watch, which news article to read, which product/service to buy, and so on. In healthcare, the automatic analysis provided by AI produces rapid recommendations that can be presented to both the clinician and the patient. It largely simplifies the process so that together the clinician and patient can review the analysis to make the best decision. The above examples manifest an important theme in human-centered computation, that is, the AI-assisted decision making. In words, AI usually has access to unlimited computational power backed by the availability of a large amount of data so that is

capable to intelligently abstract out useful information. Human then review the information output by the AI and make the final call on what decisions to take.

One natural question in the above framework is: How does humans respond to the information presented by AI? In [47], I follow the standard assumption that humans are Bayesian rational and studies the competition of multiple information providers. However, since this Bayesian rational assumption might not hold in real world, especially in low-stake context. To make more sense of human behavior in real world, I conducted online behavioral experiments where I recruited 400 human subjects (i.e., workers) from Amazon Mechanical Turk to examine given prior beliefs, how workers update their beliefs and take actions [174]. The experimental results demonstrate that worker behavior has significantly deviated from the model of Bayesian rationality. We also show that an alternative human model (discrete choice model coupled with probability weighting) better aligns with workers' real behavior.

Human Behavior Modeling – Learning from Peer Communication. In addition to study the human behavior in information design problem, I also investigated the relaxation of *independency* assumption among workers in crowdsourcing tasks. In particular, I explored *peer communication*, in which a pair of crowd workers directly communicate when producing the data. Crowdsourcing has become a popular tool for large-scale data collection where it is often assumed that crowd workers complete the work *independently*. I relaxed such independence property and explore the usage of *peer communication* – a kind of direct interaction between workers – in crowdsourcing [179]. Experimental results conducted on three types of tasks consistently suggest that work quality is significantly improved in tasks with peer communication compared to tasks where workers complete the work independently. I further explored how to utilize peer communication to optimize the requester's utility while taking into account higher data correlation and higher cost introduced by peer communication. I modeled the requester's online decision problem of *whether* and *when* to use peer communication in

crowdsourcing as a constrained Markov decision process which maximizes the requester's total utility under budget constraints. Our proposed approach is empirically shown to bring higher total utility compared to baseline approaches.

1.4 Overview of this Dissertation

This dissertation studies the human-centered machine learning both from the perspective of algorithm design and the human behavior. In Chapter 2, we explore the problem of human impacts on algorithm design. We consider a setting where a user-generated content platform has to rely on user feedback (e.g., number of likes, upvotes, etc) to learn the content qualities so that to select the best content to display to users. The platform's goal is to design an efficient learning algorithm with sublinear regret guarantee. This chapter is based on joint work with Chien-Ju Ho [173]. In Chapter 3, we then explore the problem of algorithm impacts on human welfare. We study the long-term impact of actions informed by the consequential decisions in a sequential decision-making environment. This chapter is based on joint work with Chien-Ju Ho and Yang Liu [175]. To better align the theory and practice, in Chapter 4 and Chapter 5, we run online behavior experiments to study what is the real human behavior in practice. In particular, on Amazon Mechanical Turk, we run behavioral experiments to understand how human respond to information presented to them under the scenario that AI can provide assistive information to humans to help them make the decision. Based on the empirical observations, we develop a more realistic human behavior model in this AI-assisted decision-making environment. This is based on joint work with Chien-Ju Ho [174]. We also run behavioral experiments to understand how human learn from the their communications with the peers. We then also develop an algorithmic framework to utilize peer communication to optimize the requester's utility while taking into account higher data correlation and

higher cost introduced by peer communication. This part of work is based on joint work with Chien-Ju Ho and Ming Yin [179].

Chapter 2

Algorithm Design: Accounting for Human Behavior

In a multi-armed bandit problem, a learner sequentially selects from a set of arms. Each arm is associated with some unknown reward distribution. After selecting an arm, the learner observes the realized reward for the selected arm. The goal of the learner is to maximize the total rewards obtained from selected arms over time. The performance of the bandit algorithm is often measured in terms of *regret*, defined as the difference between the algorithm performance and the performance of an oracle which can select the best arm in hindsight. The multi-armed bandit formulation provides a theoretical framework for resolving the classical exploration-exploitation tradeoffs in online decision problems under uncertainty. Therefore, multi-armed bandits have been studied in a wide range of applications in various domains, such as medical trials, online auctions, or web advertisements.

We explore the applications of bandits settings to human-in-the-loop systems. For example, consider user-generated content platforms, such as Youtube, Quora, or Stack Exchange. On

these platforms, content qualities vary across a wide spectrum. Ideally, the platform would like to select the best content to display to users to optimize users' experience. However, the content qualities are often not known in advance, and the platform needs to learn the content qualities through user feedback (e.g., number of likes, upvotes, etc). This naturally leads to a bandit problem, in which the platform needs to balance exploration (display content with fewer feedback instances to users to acquire more information) and exploitation (display content with higher empirical ratings to optimize users' happiness), as studied in the literature [65, 119].

Many challenges arise when humans are involved in the bandit learning process. In recent years, researchers have addressed various strategic issues brought up by humans involved in bandit learning [65, 119, 125, 142]. However, in these works, it is assumed that users' feedback is *unbiased* in representing the reward of selecting an arm (e.g., in user-generated content platforms, users' average ratings are used as the estimates for content qualities). On the other hand, as the empirical studies suggest [134, 154, 160], user feedback is often biased by other users' feedback. For example, users have the tendency to provide feedback that agrees with the majority opinion even if their experience disagrees (i.e., the herding effect). These empirical evidences suggest a different stochastic model in that each observed feedback instance might be biased by the feedback history. Moreover, this biased user feedback introduces additional challenges. Since user feedback only represents biased reports of the realized rewards, suppose the goal of the platform is to maximize the total rewards over time (which may be interpreted as the overall user experience), can a platform achieve sublinear regret from only observing biased feedback?

In this paper, we study a variant of the multi-armed bandit problem with human biased feedback. In our setting, the learner/platform only observes human-generated feedback instead of the realized reward when selecting an arm. The human feedback depends on both

the realized reward and other users' feedback for the selected arm. The goal of the learner is to maximize the total realized rewards for the selected arms while only having access to biased human-generated feedback.

To address the issues of user biased feedback, we explore two natural user feedback models and study their impacts to the design of bandit algorithms. The first model, avg-herding feedback model, assumes that user feedback for an arm depends on the realized reward and the *average feedback* (i.e., the ratio of positive feedback) of the arm so far. We show that, under this model, the dynamics of user feedback over time is mathematically connected to asymptotic approximation [151]. In particular, the average feedback changes over time as if users are performing online gradient descent on a latent function with a decreasing step size. With this mathematical connection, we characterize the convergence and convergence rates for the average feedback of an arm under some mild conditions. These convergence results enable us to design a bandit algorithm based on UCB (Upper Confidence Bound) algorithm and achieve sublinear regret.

While the results on the first model are promising, our results on another natural model, beta-herding feedback model, paint a very different picture. In this model, user feedback is biased by not only the average feedback in the past, but also the number of feedback instances the arm has received so far. This model captures a natural scenario that users might be biased more heavily if there exists more feedback instances in the history. We show that, under this model, the average feedback of an arm converges to a random variable with non-zero variance. This implies that, even with an infinitely number of feedback instances for the arm, the learner is not able to infer the expected reward of the arm through observing the average user feedback. We further show that, using arguments from information theory, there exist no bandit algorithms that can achieve sublinear regret when user feedback follows beta-herding feedback model.

We next present a toy example to demonstrate that it is possible to get around the above impossible result by modifying the information structure to break the assumption that users follow beta-herding feedback model. In particular, if the learner is allowed to hide the historical information from a small portion of the users, under some styled user models on how users respond to information structures, it is possible to design an algorithm achieving sublinear regret. This result opens up a potentially interesting line of future research: Can the learner adaptively *design* the information structures to improve the overall utility?

Our results demonstrate the importance of understanding human behavior when learning from human generated feedback. A small deviation on the user behavior model and/or the design of the information structure could have significant impacts on the overall system outcome. Therefore, platforms and decision makers should carefully take these into account when designing learning algorithms in systems with humans in the loop.

2.1 Related Work

In this section, we review the relevant literature in multi-armed bandit problems, recent studies on human-in-the-loop bandit learning, and the literature on social influences and social learning that share similar motivations of this work.

Multi-armed bandit problems. Our work is a variant of the well-studied multi-armed bandit problem [111]. Bandit problems traditionally assume the rewards generated by each arm at each round are directly observable, and the research focus has been divided into settings in which rewards are either independent and identically distributed (i.i.d.) [8] or adversarial [7, 9]. There exist other works that assume rewards are neither i.i.d. drawn nor adversarial. For example, bandits with Markovian rewards [136, 141] assume the state of each arm evolves according to a Markov process. Other non-stationary bandit problems [21, 61]

consider the setting in which the rewards distribution might change over time, independent of previous actions. More recently, researchers have addressed the setting in which the rewards are strategic choices of humans and could be influenced by how the bandit algorithm is designed [65, 119]. Our work differs from the above works in that, in our setting, the "state" (history information) of each arm is correlated with learner's actions and there might be infinitely many states. Moreover, in our setting, the algorithm cannot observe realized rewards but only has access to biased feedback while previous work assume the realized rewards are observable.

Human-in-the-loop bandit learning. Recently, there have been works exploring bandit learning with humans in the loop [56, 109, 125, 142]. In the setting of these works, the learner cannot directly choose which arms to play. Instead, at each time step, a myopic agent, who only aims to maximize her own reward at the single time step she is involved in, chooses which arm to play. Since the agent only cares about her instant payoff, she does not have incentives to explore and tends to always exploit, and this collective arm playing will lead to the convergence to the suboptimal arm. Researchers have been attempting to address this problem by considering different ways of *persuading* agents to perform exploration, including offering agents payments to perform exploration [56] or utilizing information asymmetry to lead agents to explore by designing what information to show to each agent [109, 125, 142]. The idea of utilizing information asymmetry to persuade agents is similar to Bayesian Persuasion [96] in economics. The above line of work has focused on settings in which humans are involved in *arm selection*, i.e., which arm is played in each round. In this work, we focus on a parallel aspect of human involvements, in which humans are involved in *feedback generation*.

Social influences and social learning. Our feedback models are motivated by the empirical evidences that users' decisions are influenced by not only their own experience but also other users' decisions [134, 154, 160]. For example, [134] empirically show that, a post on a forum is more likely to receive positive feedback (i.e., *upvotes*) if the platform insert an upvote right after the post is made. Similar discussion also appears in the social learning literature in economics [13, 23, 164]. They discuss the setting in which users' decisions might be influenced by other users' decisions. Therefore, under certain conditions, users might collectively make the bad decision since they might follow what other users do regardless of what they privately know. In prior work, there is not much discussion on either the convergence rate of users' aggregate behavior or the impacts on the system designer's perspective. In this work, we focus on deriving the dynamics of user feedback over time and explore the impacts on the design of bandit algorithms.

2.2 Model

Let K be the number of arms. Each arm $k \in [K] = \{1, ..., K\}$ is associated with an unknown quality $\theta_k \in [0, 1]$. Let $I^* = \arg\max_k \theta_k$ and $\theta^* = \theta_{I^*}$ be the index of the best arm and the associated highest expected quality. At each round t, a user randomly drawn from the population arrives, the learner selects an arm $I_t \in \{1, ..., K\}$ for the arriving user. The user then gets a binary reward Z_t (positive or negative experience) with mean θ_{I_t}.

$$Z_t \sim \texttt{Bernoulli}[\theta_{I_t}]$$

The reward is not observable to the learner. However, after receiving the reward, each user provides a binary feedback $X_t \in \{0, 1\}$ about this arm. The goal of the learner is to maximize the total rewards users receive while observing only the (potentially biased) feedback. Note

that when the feedback is the same as the realized reward, i.e., $X_t = Z_t$ for all t, this problem reduces to standard bandit setting. Below we describe the user feedback models, i.e., how X_t is generated.

User feedback models. Users' feedback depends on both the realized rewards and the feedback history of the arms. The feedback history of arm k up to time t can be summarized by $n_{k,t}$ and $\rho_{k,t}$, which represent the number of feedback instances and the ratio of positive feedback for arm k up to round t. We assume $n_{k,0} = \rho_{k,0} = 0$ to simplify the presentation, however, our results can be easily extended to settings with non-zero $n_{k,0}$ and $\rho_{k,0}$, which can be used to represent the users' *prior* of the arm quality. Again, if users provide unbiased feedback, we should have $X_t = Z_t$ for all t.

In this paper, we model the feedback generation as a stochastic process. We define a feedback function to model the probability of obtaining positive feedback for an arm from a user randomly drawn from the population. Note that a feedback function describes the characteristics of the *user population* the platform is interacting with instead of a single specific user. In particular, we introduce $\texttt{Feedback}(\theta, \rho, n)$ to model the probability of obtaining positive feedback from a user given that the arm quality is θ and the history information of the arm is summarized by its average feedback ρ and the number of feedback instances n.

As a special case, when $\texttt{Feedback}(\theta, \rho, n) = \theta$, user feedback represents unbiased samples of the arm quality.

In this paper, we explore two natural feedback models.

- Avg-herding feedback model:

In this feedback model, user feedback is biased by the average feedback of the arm. In particular, the feedback function has the form

$$\texttt{Feedback}(\theta, \rho, n) = F(\theta, \rho).$$

In Section 2.3, we study the stochastic process of user feedback specified by a general set of feedback functions F. We then discuss the impacts of this stochastic feedback generation on the design of bandit algorithms.

- Beta-herding feedback model:

  In this feedback model, user feedback is biased by the average feedback and the number of feedback instances. In particular, we consider a natural setting and assume users update their beliefs about the arm quality in a Bayesian manner. Users treat the historical ratings as the prior signals of the arm quality and update the posterior based on their own experience. They then provide feedback according to their posterior.

  We introduce a factor $m \geq 0$,which can be interpreted as the weights users put on their own experience. When the arm quality is θ and the arm history is (n, ρ), the expected number of *positive signals* the user will obtain is $m\theta + n\rho$, where the first term is the expected positive signals the users receive from their own experience (i.e., arm quality multiplied by the weight) and the second term is the number of positive signals from other users. The total number of signals is $m + n$.

  Therefore, the probability of obtaining positive feedback for arm k at round t can be written as

$$\texttt{Feedback}(\theta, \rho, n) = \frac{m\theta + n\rho}{m + n}. \tag{2.1}$$

Note that when $m \to \infty$, user feedback provides unbiased samples of the arm quality.

Regret notions. The goal of the learner is to maximize the sum of rewards users receive over time. Let $\mathcal{A}$ be the algorithm the learner deploys and $\{I_t\}$ are the arms selected by $\mathcal{A}$. We define the *regret* as $R_{\mathcal{A}}(T)$.

$$\mathbb{E}[R_{\mathcal{A}}(T)] = T\theta^* - \mathbb{E}_{\mathcal{A}}\left[\sum_{t=1}^{T} \theta_{I_t}\right],$$

where the expectation is taken over the randomness of the reward realization and the algorithm. In particular, we are interested in the region of $T \to \infty$ and aim to understand under what conditions we can achieve asymptotic sublinear regret, i.e., $\mathbb{E}[R(T)] = o(T)$, when user feedback is biased by historical feedback.

2.3 Bandits with Avg-Herding Feedback Model

In this section, we explore the bandit learning problem when user feedback follows avg-herding feedback model. We first derive the stochastic process of the feedback generation for a single arm and characterize the convergence and convergence rate of users' average feedback over time. We then discuss how this user feedback model impacts the design and analysis of bandit algorithms.

2.3.1 Stochastic process of feedback generation

In the following discussion, we explore the feedback dynamics of a single arm, i.e., the stochastic process of feedback generation. We omit the arm's index k in the subscript when it is clear from the context. Also, since user feedback is biased by the history of only the selected

arm, to simplify the presentation, we consider the case that the same arm is repeatedly selected and therefore $n_t \equiv t$ when studying the stochastic process for a single arm.

Connection to stochastic approximation

Recall that in avg-herding feedback model, when the quality of the arm is θ and the average feedback of the arm is ρ, the probability for a user to provide a positive feedback is $F(\theta, \rho)$. The stochastic process of the feedback dynamics can be expressed as follows: at the $(t+1)-$th round, the feedback X_{t+1} provided by the user is drawn randomly from a Bernoulli distribution: $\texttt{Bernoulli}[F(\theta, \rho_t)]$. The history information of the arm (n_{t+1}, ρ_{t+1}) are updated based on the realized feedback.

As mentioned, we simplify the presentation by setting $n_t \equiv t$. Therefore, we focus on how ρ_t evolves over time. By simple weighted averaging, we have

$$\rho_{t+1} = \frac{t}{t+1}\rho_t + \frac{1}{t+1}X_{t+1} = \rho_t - \frac{1}{t+1}\left(\rho_t - X_{t+1}\right).$$

Define the noise term $\xi_t = \mathbb{E}[X_t|\mathcal{F}_{t-1}] - X_t = F(\theta, \rho_{t-1}) - X_t$, where $\mathcal{F}_t = \sigma(\{X_t\}_{t\geq 1})$ is the filtration of the stochastic process. It is easy to see that $\mathbb{E}[\xi_t|\mathcal{F}_{t-1}] = 0$. Also let $\eta_t = 1/t$ be the step size (learning rate). We can rewrite the above recursive definition as an update rule in stochastic approximation [58, 151].

$$\rho_{t+1} = \rho_t - \eta_{t+1}(\rho_t - F(\theta, \rho_t) + \xi_{t+1}) \tag{2.2}$$

In particular, suppose there exists a latent function $G(\theta, \rho)$, such that $\partial G/\partial \rho = \rho - F(\theta, \rho)$, then Equation (2.2) is equivalent to the update rule for stochastic gradient descent with step

size η_{t+1}:

$$\rho_{t+1} = \rho_t - \eta_{t+1}(\nabla_\rho G(\theta, \rho_t) + \xi_{t+1})$$

With this observation, the stochastic process of the average feedback updates is equivalent to users collectively performing stochastic gradient descent for a latent function G with a decreasing step size. Below we utilize this mathematical connection and discuss conditions on the convergence and convergence rates of the average feedback ρ. We then discuss the impacts of this stochastic process on the design and analysis of bandit algorithms.

On the convergence and convergence rate of $\lim_{t\to\infty} \rho_t$.

We first specify the assumptions needed to establish the asymptotic behavior of the limit of average feedback.

A1. $F(\theta, \rho)$ is strictly increasing in θ and non-decreasing in ρ;

A2. $F(\theta, \rho)$ is differentiable and $L_F^\rho-$Lipschitz continuous with respect to ρ.

A1 implies that, conditional on the same quality (average feedback), an arm with better average feedback (quality) receives more positive feedback in expectation. *A2* assumes the improvement is smooth with respect to ρ. While the differentiable property of F can be satisfied if the population is large and smooth, we note that the differentiable property is only for analytical convenience. Our results still hold even if F is only continuously differentiable in some local neighbourhood of equilibrium points.

We would also like to note that these two assumptions are relatively mild. As an example, below we give a general set of feedback functions F that satisfy the above assumptions.

Example 2.3.1. *Consider the following set of feedback functions:* $F(\theta, \rho) = w_1\theta + w_2\rho$, *for any* $w_1, w_2 \geq 0$ *and* $w_1 + w_2 = 1$. *This set of feedback functions satisfies both of the*

assumptions. It also has very natural interpretations. In particular, it specifies that, the probability of receiving a positive feedback from a random user (drawn from the population) $F(\theta, \rho)$ is the weighted average of the arm quality θ and other users' average feedback ρ.

Armed with the above assumptions, we can formally characterize the convergence of ρ_t.

Lemma 2.3.1. *Let $\mathcal{S}_\theta := \{\rho : \rho - F(\theta, \rho) = 0\}$. We have $\mathbb{P}(\lim_{t\to\infty} \rho_t \in \mathcal{S}_\theta) = 1$.*

The above lemma demonstrates that ρ_t converges to one of the points in a set $\mathcal{S}_\theta$ and characterizes the points in $\mathcal{S}_\theta$. Recall that the latent function $G(\theta, \rho)$ satisfies $\partial G/\partial \rho = \rho - F(\theta, \rho)$. Therefore, the lemma illustrates that the average feedback will converge to one of the points in $\mathcal{S}_\theta$, the set of the local optimal points for the latent function G. This intuition suggests that, when the latent function G is strongly convex, since there exists only one local optimal point (which is the global optimal), we should be able to show that ρ_t will almost surely converge to the global optimal.

Moreover, the convexity of G is correlated with the value of the Lipschitz constant L_F^ρ. In particular, when $L_F^\rho < 1$, by definition, we have $\nabla_\rho F(\theta, \rho) < 1$ for all θ and ρ. Since $\nabla_\rho^2 G(\theta, \rho) = 1 - \nabla_\rho F(\theta, \rho)$, when $L_F^\rho < 1$, $\nabla_\rho^2 G(\theta, \rho) > 0$ for all θ and ρ. Therefore, G is strongly convex when $L_F^\rho < 1$. Below we formally characterize the convergence of ρ_t when G is strongly convex.

Corollary 2.3.2. *Given $L_F^\rho < 1$, i.e., G is strongly convex, there exists a unique ρ^* that satisfies $\rho^* - F(\theta, \rho^*) = 0$, such that $\mathbb{P}(\lim_{t\to\infty} \rho_t = \rho^*) = 1$.*

Next we provide the results on the convergence rate of ρ_t and focus on the case when G is strongly convex. In particular, we introduce $\bar{\lambda} > 0$, such that $\nabla_\rho^2 G \geq \bar{\lambda} > 0$.

Theorem 2.3.3. *Given $L_F^\rho < 1$, i.e., G is strongly convex. $\forall \epsilon > 0$, we have,*

$$\mathbb{P}(|\rho_t - \rho^*| \geq \epsilon) \leq \exp\left(-\frac{(\epsilon - \epsilon_t)^2}{2\sum_{i=1}^t L_i}\right),$$

where $L_i = \eta_i^2(\prod_{j=i}^{t-1}(\eta_{j+1}^2(L_F^\rho - 1)^2) - 2\bar{\lambda}\eta_{j+1} + 1)$,
$\epsilon_t = \exp(-\bar{\lambda}S_t)|\rho_0 - \rho^*| + \sqrt{\sum_{i=0}^{t-1}\eta_{i+1}^2\exp(-2\bar{\lambda}(S_t - S_{i+1}))}$,
and $S_t = \sum_{i=1}^t \eta_i$.

Remark 2.3.2. *We would like to offer a few observations to help interpret the convergence bound*[1]*. In particular,*

- *when $t \to \infty$, $\epsilon_t \to 0$,*
- *when $\bar{\lambda} \in (0, 1/2)$, $\sum_{i=1}^t L_i = \mathcal{O}(t^{-2\bar{\lambda}})$, and*
- *when $\bar{\lambda} \in [1/2, \infty)$, $\sum_{i=1}^t L_i = \mathcal{O}(1/t)$.*

So we can characterize the bound in two regions based on whether $\bar{\lambda} \geq 1/2$. As a special case, when user feedback is unbiased, i.e., $F(\theta, \rho) = \theta$, we have $\bar{\lambda} = 1$, and the bound reduces to $\mathbb{P}(|\rho_t - \rho^*|) \geq \epsilon) \leq \mathcal{O}(e^{-\epsilon^2 t})$, the same as the standard Chernoff bound. Moreover, in our setting, since $F(\theta, \rho)$ is non-decreasing in ρ, i.e., $\nabla_\rho F \geq 0$. We have $\nabla_\rho^2 G = 1 - \nabla_\rho F \leq 1$. Therefore, while our bound holds for the region $\bar{\lambda} \in (0, \infty)$, in our setting, we focus on the region $\bar{\lambda} \in (0, 1]$.

Note that in this theorem, the convergence rate is a function of $\bar{\lambda}$, which is the property of the function G (hence the property of the feedback model F). As an intuitive interpretation, recall that $\nabla_\rho^2 G \geq \bar{\lambda}$ and $\nabla_\rho G = \rho - F(\theta, \rho)$. Therefore, small $\bar{\lambda}$ implies large $\partial F/\partial\rho$, which means user' feedback is influenced more by the other users' feedback and relatively less by

[1]The detailed derivations are included in the appendix of the full paper.

the arm quality. When users' feedback depends less on the arm quality, it requires more feedback to infer the arm quality, and therefore the convergence is slower. This intuition aligns with the theorem, in which smaller $\bar{\lambda}$ leads to a slower convergence rate.

2.3.2 Designing bandit algorithms

Given the convergence bound in Theorem 2.3.3, we can design a UCB-like algorithm that achieves sublinear regret. We assume the learner has knowledge of the feedback model F. Note that since F models the behavior of feedback generation for the *user population* the platform is interacting with, this assumption only requires the platform to have knowledge of the population instead of any particular users[2].

In each round of our algorithm, the learner maintains an estimator $\hat{\theta}_{k,t}$ of arm k's quality from the observation of average feedback $\rho_{k,t}$. From Lemma 2.3.1, an asymptotically unbiased and consistent estimator of arm's quality $\hat{\theta}_{k,t}$ can be obtained by solving the following equation.

$$\hat{\theta}_{k,t} = \max\{\min(\{\hat{\theta}_{k,t} : F(\hat{\theta}_{k,t}, \rho_{k,t}) = \rho_{k,t}\}, 1), 0\} \tag{2.3}$$

Intuitively, the solutions of the above equation represent the set of local optimal points of G. Moreover, we can show that the estimator $\hat{\theta}_{k,t}$ is unique for every $\rho_{k,t}$ if *A1* is satisfied.

Lemma 2.3.4. *Suppose A1 is satisfied, for any $\rho_{k,t}$, there exists a unique $\hat{\theta}_{k,t}$ that satisfies Equation (2.3).*

Given the convergence bounds and the estimator $\hat{\theta}_{k,t}$, we are ready to describe our proposed UCB-like algorithm Avg-UCB, as specified in Algorithm 1. The key differences to the standard

[2]In practice, this assumption can be approximately satisfied through market research or behavioral experiments, which study the connection between users' real experience (i.e., Z_t) and reported feedback (i.e., X_t). Moreover, our results are robust to small estimation noises of F.

UCB algorithms are that: First, we maintain a quality estimate $\hat{\theta}_{k,t}$ for each arm k at each time t by solving Equation (2.3) instead of using empirical average feedback. Second, the confidence interval in the UCB index is derived from the convergence rates as specified in Theorem 2.3.3. Our algorithm takes as input parameters β and $\bar{\lambda}$. β plays a similar role as the constant in UCB confidence radius to balance exploration and exploitation. $\bar{\lambda}$ is the parameter of the problem instance. Note that our algorithm only requires to find some $\bar{\lambda}$ such that $\nabla_\rho^2 G \geq \bar{\lambda}$.

Algorithm 1 Avg-UCB for Avg-Herding Feedback Model

Input: β, $\bar{\lambda}$, K.
Initializations: first K rounds, play each arm once
for $t = K+1, ..., T$ **do**
    for each $k \in \{1, ..., K\}$ **do**
        Compute $\hat{\theta}_{k,t-1}$ from (2.3).
        $\texttt{UCB}_{k,t} = \hat{\theta}_{k,t-1} + \sqrt{\frac{\beta \ln(t-1)}{n_{k,t-1}^{\min\{1, 2\bar{\lambda}\}}}}$.
    Choose arm $I_t \in \arg\max_{k=1,...,K} \texttt{UCB}_{k,t}$.
    (Ties are broken in some consistent way)
    Receive feedback X_t.
    $\rho_{I_t,t} \leftarrow (\rho_{I_t,t-1} \times n_{I_t,t-1} + X_t)/(n_{I_t,t-1} + 1)$
    $\rho_{k,t} = \rho_{k,t-1}, \forall k \neq I_t$.
    $n_{I_t,t} \leftarrow n_{I_t,t-1} + 1$
    $n_{k,t} \leftarrow n_{k,t-1}, \forall k \neq I_t$.

The following theorem gives the regret bound for the algorithm Avg-UCB.

Theorem 2.3.5. *Suppose A1 and A2 are satisfied and $L_F^\rho < 1$. Let $\bar{\lambda}' = \max\{1, 1/(2\bar{\lambda})\}$, $\Delta_k = \theta^* - \theta_k$. With appropriately chosen β [3] the expected regret for Avg-UCB is bounded by:*

$$\mathbb{E}[R(T)] \leq \sum_{k \neq I^*} \Delta_k (4 \ln T / (C\Delta_k^2))^{\bar{\lambda}'} + K\pi^2/6,$$

[3]The choice of β depends on the parameters of $F(\theta, \rho)$. The detailed derivation is tied with the proof and is included in the appendix of the full paper.

where C is a constant that is dependent on the properties of feedback function F.

We introduce an additional notion $\bar{\lambda}' = \max\{1, 1/(2\bar{\lambda})\}$ to simplify the presentation due to the different convergence rates on whether $\bar{\lambda} < 1/2$ as discussed in Remark 2.3.2. Similar to the discussion on the convergence rate, the dependency of the above upper regret bound on $\bar{\lambda}'$ implies that it is harder to learn the quality of an arm if users are biased more by the historical information rather than the arm quality.

The above regret bound is a gap-dependent bound. In particular, let $\Delta_{\min} = \min_{k:k\neq I^*} \Delta_k$. The regret bound can be written as: $\mathbb{E}[R(T)] = \mathcal{O}\left(\frac{(\ln T)^{\bar{\lambda}'}}{\Delta_{\min}^{2\bar{\lambda}'-1}}\right)$. Observe that $\lim_{T\to\infty} \mathbb{E}[R(T)]/T \to 0$ for any $\bar{\lambda}' > 0$. Therefore, the algorithm achieves sublinear regret as long as G is strongly convex (i.e., $\bar{\lambda}' > 0$).

Moreover,we can derive gap-independent bounds from the above bound. When $\bar{\lambda} \geq 1/2$ (which includes the unbiased feedback setting with $\bar{\lambda} = 1$), we can show that $\mathbb{E}[R(T)] = \mathcal{O}(\sqrt{T \ln T})$, which matches the standard regret bound without biased feedback.

What if G is not convex. Our algorithm relies on the assumption that the latent function G is convex, i.e., $L_F^\rho < 1$. This assumption implies that users' feedback is not influenced too heavily by the change of feedback history. While this assumption seems mild, it is natural to wonder whether we can obtain similar results when G is not convex.

We would like to note that even in settings when G is non-convex, the statements of Lemma 2.3.1 and 2.3.4 still hold. This means the average user feedback for each arm still converges to some point, and we can infer the arm quality from the converged average feedback. The main obstacle to overcome is to derive the convergence rate as in Theorem 2.3.3. This problem is challenging as it is equivalent to deriving the convergence rate of optimization for non-convex functions. There have been recent works focusing on deriving the convergence

rates in non-convex optimization in different settings [3, 62]. As long as one could characterize the convergence rate of ρ_t for non-convex function G, our bandit strategy can be adapted to generate a sublinear regret strategy (by changing the "confidence interval" in the UCB index based on the derived convergence rate).

2.4 Bandits with Beta-Herding Feedback Model

In the previous section, we explore avg-herding feedback model, in which user feedback is biased only by the average feedback of the selected arm. We show that, under some mild conditions, the average feedback for an arm almost surely converges to some value, and we can infer the arm quality from the average feedback, and therefore we can design a UCB-like algorithm for achieving sublinear regret.

However, in some scenarios, user feedback may be biased by not only the average feedback but also the number of feedback instances of the arm. In this section, we explore another natural feedback model, beta-herding feedback model, and prove impossibility results. In particular, we assume users give feedback in a Bayesian manner. They treat the feedback history as the prior, i.e., for an arm with history (n, ρ), there are $n\rho$ positive signals and $n(1-\rho)$ negative signals for the arm. After they experience the binary reward (drawn according to the arm's quality distribution), they update their posterior by treating their experience as m signals and then provide feedback according to the posterior. Therefore, in expectation, the probability for them to provide positive feedback for an arm with quality θ and history (n, ρ) is $\texttt{Feedback}(\theta, \rho, n) = (m\theta + n\rho)/(m+n)$.

2.4.1 Stochastic process of feedback generation

The first natural attempt is to replace $F(\theta, \rho_t)$ with $\texttt{Feedback}(\theta, \rho, n)$ in Equation (2.2) and apply similar analysis using stochastic approximation. However, when $\texttt{Feedback}(\theta, \rho, n)$ follows beta-herding feedback model, one can not directly apply this approach. Briefly speaking, the update rule in Equation (2.2) aims to find the equilibrium points of the feedback function. However, in beta-herding feedback model, the feedback function is changing over time, and it is not trivial whether the converged points satisfy the set of properties as derived with avg-herding feedback model.

Instead, we make the observation that the stochastic process of beta-herding feedback model is similar to the urn process [75]. We utilize the property of *exchangeability* for the feedback history to give the characterization of ρ_t process. Below we formally characterize the stochastic process of ρ_t with beta-herding feedback model.

Lemma 2.4.1. *Consider the stochastic process in Equation (2.2) with the feedback model described in Equation (2.1),* $\lim_{t\to\infty} \rho_t$ *converges almost surely to a random variable specified by a beta distribution. In particular,*

$$\lim_{t\to\infty} \rho_t \sim \text{Beta}(m\theta, m(1-\theta)).$$

Note that when the feedback is unbiased, i.e., when $m \to \infty$, the beta distribution will shrink to a Dirac delta function which has the point mass exactly in θ.

2.4.2 The impossibility result

In this section, we show that there exist no bandit algorithms that achieve sublinear regret if user feedback follows beta-herding feedback model.

Lemma 2.4.1 implies that, even if we obtain an infinite number of feedback instances for an arm, we cannot accurately infer the arm quality with high probability from the empirical average feedback ρ_∞. A natural next question to ask is, if we take into account all the feedback generated in the process, whether it is possible to infer the true arm quality. Below we use the notion of Fisher information to answer the question. In short, Fisher information provides a way to quantify the amount of information about the latent parameter θ we can obtain for observing each sample of a random variable X_i. Since Fisher information is additive, we can show that,

Lemma 2.4.2. *Consider the stochastic process in Equation (2.2) with the feedback model described in Equation (2.1). Let $\mathcal{I}_t(\theta)$ denote the Fisher information of θ for observing $t-th$ sample. We have*

$$\lim_{t\to\infty} \sum_{i=1}^{t} \mathcal{I}_i(\theta) = \mathcal{O}(1).$$

Using this fact, by the general Cramér-Rao bound, we know that, for any estimator $\hat{\theta}_t$, the variance of $\hat{\theta}_t$ must follow:

$$\mathrm{Var}(\hat{\theta}_t) \geq \Theta\left(\frac{1}{\sum_{i=1}^{t} \mathcal{I}_i(\theta)}\right)$$

Since $\lim_{t\to\infty} \sum_{i=1}^{t} \mathcal{I}_i(\theta)$ is bounded, the variance of any estimator will not shrink to zero even with infinitely many observations. Therefore, the learner cannot accurately infer the arm quality with high probability in the beta-herding feedback model and therefore cannot guarantee to identify the best arms even with infinitely many feedback instances. Since the learner only observes the feedback, we can conclude the following.

Theorem 2.4.3. *If users' feedback follows beta-herding feedback model, there exists no bandit algorithm that can achieve sublinear regrets in our setting.*

We note that the technique used in the proof can be extended to a more general feedback model for impossibility results. The intuition is to use Fisher information to quantify how informative a given data is with respect to a set of parameters and the influence of the data itself on the estimate. For different models, if the amount information for each feedback can be quantified, the same techniques can be applied.

2.4.3 An alternative approach: Designing information structures

Theorem 2.4.3 presents a strong impossibility result: if all feedback instances are generated according to beta-herding feedback model, we cannot design any bandit algorithms to achieve sublinear regret. A natural approach to get over this impossibility results is to break the assumption by taking interventions. Inspired by Bayesian persuasion [96], which designs the information structure to *persuade* agents to take certain actions, we explore whether we could design information structures to induce certain types of "feedback". For example, in the extreme case, if we do not show any historical information to users, and assume users provide unbiased feedback when no information is presented, then the problem reduces to standard bandit settings. However, in practice, we might not want to dramatically change the whole platform and might want to take as few interventions as possible. This leads to an interesting research question on whether we can minimally intervene the existing design of information structure, such that it is possible to design bandit algorithms with sublinear regrets.

In this section, we present a simple algorithm as a *toy example* to demonstrate the idea. A full study along this direction requires a careful and thorough modeling and is out of the scope of this paper. We consider the constrained setting in which the platform can only

choose among two information design in each round, either showing all history information to users (and assuming users' feedback follow beta-herding feedback model) or showing no history information (and assuming users provide unbiased feedback). Our goal is to minimize the number of rounds that show no information to users while achieving sublinear regret. In particular, we propose a *two-stage policy*, as described in Algorithm 2, which shows no historical information for the first $\lfloor T^\alpha \rfloor$ rounds and resumes to standard design afterwards.

The regret bound of Algorithm 2 is given as follows.

Algorithm 2 *two-stage policy*

Input: learning rounds parameter $\alpha \in (0,1)$, exploration parameter $\beta > 0$, number of arms K.
Initializations: first K rounds, play each arm once
for $t = K+1, ..., \lfloor T^\alpha \rfloor$ **do**
    for each $k \in \{1, ..., K\}$ **do**
        $\texttt{UCB}_{k,t} = \hat{\theta}'_{k,t-1} + \sqrt{\frac{\beta \ln(t-1)}{n_{k,t-1}}}$, where $\hat{\theta}'_{k,t-1} = \frac{\sum_{s=1}^{n_{k,t-1}} \mathbb{1}\{I_s = k\} X_{t-1}}{n_{k,t-1}}$.
    Choose arm $I_t \in \arg\max_{k=1,...,K} \texttt{UCB}_{k,t}$.
    Present arm I_t without showing its history information to the user, and get feedback X_t.
    $\rho_{I_t,t} \leftarrow (\rho_{I_t,t-1} \times n_{I_t,t-1} + X_t)/(n_{I_t,t-1} + 1)$.
    $\rho_{k,t} = \rho_{k,t-1}$ for $k \neq I_t$.
    $n_{I_t,t} \leftarrow n_{I_t,t-1} + 1$.
    $n_{k,t} \leftarrow n_{k,t-1}$ for $k \neq I_t$.
Let $I_\tau \in \arg\max_{k=1,...,K} n_{k,\lfloor T^\alpha \rfloor}$.
Present arm I_τ with associated history information to the user in the remaining rounds.
(all ties broken in some consistent way)

Theorem 2.4.4. *Let $\Theta = \{\theta_1, ..., \theta_K\}$ be a bandit instance, and $\alpha \geq \ln(K(K+2))/\ln T$, then the expected regret of two-stage policy, where $\beta > 1$, is bounded from above by:*

$$\mathbb{E}[R(T)] \leq \sum_{k \neq I^*} \left(\frac{4\alpha\beta \ln T}{\Delta_k} + 8\beta\Delta_k \right) + (T - T^\alpha) \left(\sqrt{\frac{4K\alpha\beta \ln T}{T^\alpha - K}} + \frac{K}{\beta - 1} \left(\frac{T^\alpha - K}{K} \right)^{2-2\beta} \right)$$

where the second term is in an order of $\mathcal{O}\left((T - T^\alpha)\sqrt{\frac{K\beta\alpha \ln T}{T^\alpha}}\right)$.

To interpret the bound, when $\alpha \geq 1/2$, the above regret bound is in the order of $\mathcal{O}(\sqrt{\alpha T^\alpha \ln T})$, while when $\alpha < 1/2$, the above regret bound is in the order of $\mathcal{O}(\sqrt{\alpha T^{1-\alpha} \ln T})$.

Algorithm 2 presents an example that we can achieve sublinear regrets by modifying the information structures presented to users. In particular, we only need to hide the historical information from T^α users, with $\alpha < 1$, out of T users to achieve subliner regrets. Note that we only consider a naive approach in a styled model, i.e., showing no information at all in some rounds, and assume simple user feedback models. We hope our results will encourage research that considers more fine-tuned information design and more thorough models of user feedback and platform utility.

2.5 Discussion on the Applications

In this section, we provide discussion on the applications of our setting. As the motivating example of this paper, we consider user-generated content platforms that need to learn content qualities through user feedback. Our analysis and results naturally extend to platforms that rely on user reviews to provide recommendations (such as Yelp or Amazon). However, to formulate the recommendation problem as a bandit learning problem, we need to make a simplifying assumption, as made in prior work [65, 119], that users are going to *follow* the recommendations. While this assumption seems strong, in practice, it approximates users'

behavior to a certain degree. In particular, empirical studies demonstrate that the probability for a users to view an item drops significantly when the position of the item decreases [39, 91, 149]. These empirical observations suggest that a significant amount of users are indeed following recommendations (since recommended items are ranked higher). Moreover, there have been recent studies on incentivizing exploration using information asymmetry [109, 125, 142] which demonstrate it is possible to make recommendations that users will *choose* to follow. The techniques in this paper can be applied in that line of work to explore the dynamics of feedback generation.

In addition to the above example, our setting applies to scenarios when the platform cannot observe the true objective but can only use (potentially biased) estimates as the proxy for the objective. Consider the following illustrating scenario: the police station needs to decide which area to send police officers to patrol at each time step. Each area i has an intrinsic, unknown crime rate p_i. When sending police officers to an area i, the police station obtains an unobserved reward $u(p_i)$, representing the value of increased safety for the area. Assume $u(p_i)$ is increasing in p_i. After the patrol, police officers need to report the amount of criminal activities during their patrol. However, these reports might be biased by the history of *reported* crime rate of the area. For example, if there are more reports of illegal activities in the area in the history, they might stop more people for inspection. This creates biases in the reports. If the goal is to maximize the sum of $u(p_i)$, this problem can be formulated using our setting, since the objective is a function of *true* crime rates, while the decision maker only has access to *reported* crime rates. Now assume the feedback model follows beta-herding feedback model. According to our results, without additional interventions, the police station might make *unfair* decisions in where to patrol using only the biased feedback, since it is impossible for them to infer the true crime rate from the reports. This example further

emphasizes the importance of understanding human behavior in learning problems, especially when the corresponding actions have significant impacts on humans.

Chapter 3

Algorithm Design: Aligning with Human Values

Algorithms have been increasingly involved in high-stakes decision making. Examples include approving/rejecting loan applications [59, 104], deciding on employment and compensation [14, 38], and recidivism and bail decisions [4]. Automating these high-stakes decisions has raised ethical concerns on whether it amplifies the discriminative bias against protected classes [33, 140]. There have also been growing efforts towards studying algorithmic approaches to mitigate these concerns. Most of the above efforts have focused on static settings: a utility-maximizing decision maker needs to ensure her actions satisfy some fairness criteria at the decision time, without considering the long-term impacts of actions. However, in practice, these decisions may often introduce long-term impacts to the rewards and well-beings for the human agents involved. For example,

- A regional financial institute may decide on the fraction of loan applications from different social groups to approve. These decisions could affect the development of these groups:

The capability of applicants from a group to pay back a loan might depend on the group's socio-economic status, which is influenced by how frequently applications from this group have been approved [15, 37].

- The police department may decide on the amount of patrol time or the probability of patrol in a neighborhood (primarily populated with a demographic group). The likelihood to catch a crime in a neighborhood might depend on how frequent the police decides to patrol this area [64, 67].

These observations raise the following concerns. If being insensitive with the long-term impact of actions, the decision maker risks treating a historically disadvantaged group unfairly. Making things even worse, these unfair and oblivious decisions might reinforce existing biases and make it harder to observe the true potential for a disadvantaged group. While being a relatively under-explored (but important) topic, several recent works have looked into this problem of delayed impact of actions in algorithm design. However, these studies have so far focused on understanding the impact in a one-step delay of actions [74, 98, 116], or a sequential decision making setting without uncertainty [41, 86, 117, 118, 133, 186, 187].

Our work departs from the above line of efforts by studying the long-term impact of actions in sequential decision making under uncertainty. We generalize the multi-armed bandit setting by introducing the *impact functions* that encode the dependency of the "bias" due to the action history of the learning to the arm rewards. Our goal is to learn to maximize the rewards obtained over time, in which the rewards' evolution could depend on the past actions.

The history-dependency reward structure makes our problem substantially more challenging. In particular, we first show that applying standard bandit algorithms leads to linear regret, i.e., existing approaches will obtain low rewards with a biased learning process. To address this challenge, under relatively mild conditions for the dependency dynamics, we present

an algorithm, based on a phased-learning template which smoothes out the historical bias during learning, that achieves a regret of $\tilde{\mathcal{O}}(KT^{2/3})$. Moreover, we show a matching lower regret bound of $\Omega(KT^{2/3})$ that demonstrates that our algorithm is order-optimal. Finally, we conduct a series of simulations showing that our algorithms compare favorably to other state-of-the-art methods proposed in other application domains. From a policy maker's point of view, our paper explores solutions to learn the optimal sequential intervention when the actions taken in the past impact the learning environment in an unknown and long-term manner. We believe our work nicely complements the existing literature that focuses more on the "understanding" of the dynamics [86, 116, 186, 187].

3.1 Related work

Our work contributes to algorithmic fairness studied in sequential settings. Prior works either study fairness in sequential learning settings without considering long-term impact of actions [16, 66, 72, 92, 120, 143] or explore the delayed impacts of actions with focus on addressing the one-step delayed impacts or sequential learning with full information [15, 37, 41, 74, 86, 116, 133]. Our work differs from the above and studies delayed impacts of actions in sequential decision making under uncertainty. Our formulation bears similarity to reinforcement learning since our impact function encodes memory (and is in fact Markovian [141, 167]), although we focus on studying the exploration-exploitation tradeoff in bandit formulation. Our learning formulation builds on the rich bandit learning literature [8, 111] and is related to non-stationary bandits [20, 21, 107, 113, 162]. Our techniques share similar insights with Lipschitz bandits [108, 161] and combinatorial bandits [29] in that we also assume the Lipschitz reward structure and consider combinatorial action space. There are also recent works that have formulated delayed action impact in bandit learning [107, 144],

but in all of these works, the setting and the formulation are different from the ones we consider in the present work.

3.2 Model

We formulate the setting in which an institution sequentially determines how to allocate resource to different groups. For example, a regional financial institute may decide on the fraction or overall frequency of loan applications to approve from different social groups.The police department may decide on the amount of patrol time or the patrol probability allocated to different regions.

The institution is assumed to be a utility maximizer, aiming to maximize the expected reward associated with the allocation policy over time. If we assume the reward[4] for allocating a unit of resource to a group is i.i.d. drawn from some unknown distribution, this problem can be reduced to a standard bandit problem, with each group representing an *arm*. The goal of the institution is then to learn a sequence of arm selections to maximize its cumulative rewards.

In this work, we extend the bandit setting and consider the delayed impact of actions. Below we formalize our setup which introduces *impact functions* to bandit framework.

Action space. There are K *base arms*, indexed from $k = 1$ to K, with each base arm representing a group. At each discrete time t, the institution chooses an action, called a *meta arm*, which is a probability distribution over base arms. Let $\mathcal{P} = \Delta([K])$ be the $(K-1)$-dimensional probability simplex. We denote the meta arm as $\mathbf{p}(t) = \{p_1(t), \ldots, p_K(t)\} \in \mathcal{P}$, where $p_k(t)$ represents the probability of choosing a base arm k ($p_k(t)$ can be equivalently interpreted as the probability of allocating a unit resource to group k or the *portion* of the

[4]The reward could be whether a crime has been stopped or whether the lender pays the monthly payment on time. For applications that require longer time periods to assess the rewards, the duration of a time step, i.e., the frequency to update the policy, would also need to be adjusted accordingly.

resources allocated to group k). The institution only observes the reward from the arm it ends up with selecting.

Remark 3.2.1. *Note that interpreting the meta-arm as a probability distribution or a proportional allocation could impact the way the rewards are generated (i.e., does the institution observe only the reward of the realized based arm, or the rewards of all base arms with non-zero allocations.) Our analysis utilizes the idea of importance weighting and could deal with both cases in the same framework. To simplify the presentation, we focus on the harder case of interpreting the meta-arm as probability distributions, though our results apply to both interpretations.*

Delayed impacts of actions. We consider the scenario in which the rewards of actions are unknown a priori and are influenced by the action history. Formally, let $\mathcal{H}(t) = \{\mathbf{p}(s)\}_{s\in[t]}$ be the action history at time t. We define the *impact function* $\underline{\mathbf{f}(t) = F(\mathcal{H}(t))}$ to summarize the impact of the learner's actions to the reward generated in each groups, where $F(\cdot)$ is the function mapping the action history to its current impact on arms' rewards. In the following discussion, we make $F(\cdot)$ implicit and use the vector $\mathbf{f}(t) = \{f_1(t), \ldots, f_K(t)\}$ to denote the impact to each group, where $f_k(t)$ captures the impact of action history to arm k.

Rewards and regret. The reward for allocating resources to group k at time t depends on both $p_k(t)$ and the historical impact $f_k(t)$. In particular, when the institution allocates a unit of resource to group k, she obtains a reward (the instaneous reward is bounded within $[0, 1]$) drawn i.i.d. from a distribution with mean $\underline{r_k\left(f_k(t)\right) \in [0, 1]}$. $r_k(\cdot)$ is unknown a priori but is Lipschitz continuous (with known Lipschitz constant $L_k \in (0, 1]$) with respect to its input, i.e., a small deviation of the institution's actions has small impacts on the unit reward from

each group. When action $\mathbf{p}(t)$ is taken at time t, the institution obtains an expected reward

$$U_t(\mathbf{p}(t)) = \sum_{k=1}^{K} p_k(t) \cdot r_k\left(f_k(t)\right). \tag{3.1}$$

As for the impact function, we focus on the setting in which $\mathbf{f}(t)$ is a time-discounted average, with each component $f_k(t)$ defined as

$$f_k(t) = \frac{\sum_{s=1}^{t} p_k(s)\gamma^{t-s}}{\sum_{s=1}^{t} \gamma^{t-s}}, \tag{3.2}$$

where $\gamma \in [0, 1)$ is the time-discounting factor. [5] Intuitively, $f_k(t)$ is a weighted average with more weights on recent actions. We would like to highlight that our results extend to a more general family of impact functions and do not require the exact knowledge of impact functions (see discussion in **Section 3.5.2**). We also note that when $\delta = 0$, our setting reduces to a special case where the impact function only depends on the current action $p_k(t)$ (*action dependent*), instead of the entire history of actions (discounted by 0 right away). We study this special case of interest in Section 3.4.

Let $\mathcal{A}$ be the algorithm the institution deploys. The goal of $\mathcal{A}$ is to choose a sequence of actions $\{\mathbf{p}(t)\}$ that maximizes the total utility. The performance of $\mathcal{A}$ is characterized by regret, defined as

$$\mathrm{Reg}(T) = \sup_{\mathbf{p}\in\mathcal{P}} \sum_{t=1}^{T} U_t(\mathbf{p}) - \mathbb{E}\Big[\sum_{t=1}^{T} U_t(\mathbf{p}(t))\Big], \tag{3.3}$$

where the expectation is taken on the randomness of algorithm $\mathcal{A}$ and the utility realization[6].

[5] Here we follow the tradition to define $0^0 = 1$ when $\gamma = 0$.

[6] In this paper, we adopt the standard regret definition and compare against the optimal fixed policy. Another possible regret definition is to compare against the optimal dynamic policy that could change based on the history. However, calculating the optimal dynamic policy in our setting is nontrivial as it requires to solve an MDP with continuous states.

3.2.1 Exemplary Application of Our Setup

We provide an illustrative example to instantiate our model. Consider a police department who needs to dispatch a number of police officers to K different districts. Each district has a different crime distribution, and the goal (absent additional fairness constraints) might be to maximize the number of crimes caught [53]. [7] The effects of police patrol resource allocated to each district may aggregate over time and then impact the crime rate of that district. In other words, the crime rate in each district depends on how frequently the police officers been dispatched historically in this district.

To simplify the discussion, we normalize the police resource to be one unit. Each district k has a default average crime rate $\overline{r}_k \in (0, 1)$ at the beginning of the learning process. This crime rate can (at most) be decreased to $\underline{r}_k \in (0, \overline{r}_k)$. All of these are unknown to the police department. The police department makes a resource allocation decision at each time step. We use $r_k(t) \in (0, 1)$ to denote the crime rate in district k at time t, taking into account the impact of historical decisions. Assume $p_k(t)$ is the amount of police resource dispatched to district k at time t ($\sum_k p_k(t) = 1$ for all t), the expected number of crimes caught at district k at time t would be $p_k(t)r_k(t)$. Note that here $p_k(t)$ can be interpreted as the probability of allocating police resource (randomly sending the patrol team to one of the K districts) or the fraction of allocated police resource.

Below we provide one natural example of the interaction between the impact function and the reward. At time step $t + 1$, let $\mathcal{H}_k(t) := \{p_k(1), \ldots, p_k(t)\}$ denote the historical decisions of the police department for district k. Now given $\mathcal{H}_k(t + 1) = \{\mathcal{H}_k(t) \cup p_k(t + 1)\}$ where $p_k(t + 1)$ is the current decision for district k, assume that the crime rate at time $t + 1$ in

[7] As discussed by [53], there might be other goals besides simply catching criminals, including preventing crime, fostering community relations, and promoting public safety. We use the same goal they adopted for the illustrative purpose.

district k is in the following form:

$$r_k(t+1) = \overline{r}_k - f_k(\mathcal{H}_k(t+1)) \times (\overline{r}_k - \underline{r}_k), \tag{3.4}$$

where $f_k(\cdot) : [0,1]^t \to [0,1]$ is the impact function that summarizes how historical actions would impact the current crime rate. One possible example is $f_k(\mathcal{H}_k(t)) = \frac{\sum_{s=1}^{t} p_k(s)\gamma^{t-s}}{\sum_{s=1}^{t} \gamma^{t-s}}$ as we defined in Equation (3.2). This impact function has two natural properties:

- When $f_k(\mathcal{H}_k(t)) = 1$ (e.g., $p_k(s) = 1, \forall s \leq t$), the police department keeps dispatching the police officers to district k with probability 1, then district k will reach its lowest crime rate.
- When $f_k(\mathcal{H}_k(t)) \to 0$ (e.g., $p_k(s) \to 0, \forall s \leq t$), the police department rarely dispatch police officers to district k, The crime rate in district k will reach its highest level.

In this example, treating each district as an arm and directly applying standard bandit algorithms might reach suboptimal solutions since the reward dynamic is not considered. In this paper, we develop algorithms that can take into account this history-dependent reward dynamic and achieve no-regret learning. Our results hold for a general class of impact functions (under mild conditions) and do not need to assume the exact knowledge of the impact function.

3.3 Overview of Main Results

We summarize our main results in this section. First, we present an important, though perhaps not surprising, negative result: if the institution is not aware of the delayed impact of actions, applying existing standard bandit algorithms in our setting leads to linear regrets.

This negative result highlights the importance of designing new algorithms when delayed impact of actions are present.

Lemma 3.3.1 (Informal)**.** *If the institution is unaware of the delayed impact of actions, applying standard bandit algorithms (including UCB, Thompson Sampling) leads to linear regrets.*

The negative result points out the need to design new algorithms for settings with delayed impact of actions. The key challenge introduced by our setting is in estimating the arm rewards: when pulling the same meta arm at different time steps, the institution does not guarantee to obtain rewards drawn from the targeted distribution according to the chosen meta arm, as the arm reward depends on the impact function $\mathbf{f}(t)$. To address this challenge, we note that if the institution keeps pulling the same meta-arm repeatedly, the impact function (and thus the arm reward associated with the meta-arm) would converge to some value. This observation leads to our approaches. We first develop a bandit algorithm that works with impacts that converge "immediately" (or equivalently only depend on "immediate" actions, echoing the case with $\delta = 0$ in Equation (3.2)). We then propose a phased-learning reduction template that reduces our general setting to the above one and achieves a sublinear regret.

Theorem 3.3.2 (Informal)**.** *There is an algorithm that achieves an optimal regret bound $\tilde{\mathcal{O}}(KT^{2/3})$ for the bandit problem with the impact function defined in Equation* (3.2)*. In addition, there is a matching lower bound of $\Omega(KT^{2/3})$.*

To provide an overview of our approaches, we start with *action-dependent bandits* (Section 3.4), where the impact at time t depends only on the action at t, i.e., $\mathbf{f}(t) = \mathbf{p}(t)$, namely $\gamma = 0$ in Equation (3.2). This setting not only captures the one-step impact but also offers a backbone for the phase-learning template for the general history-dependent scenario. In this

setting, when a meta-arm $\mathbf{p} = \{p_1, \ldots, p_K\}$ is selected, one of the base arms k is realized with probability p_k, and the institution receives the realized reward for base arm k. However, since we also know the probability p_k for selecting each base arm, we may apply importance weighting to simulate the case as if the learner is selecting K probabilities and obtain K signals at each time step. This interpretation transforms our problem structure to a setting similar to combinatorial bandits. Furthermore, since both $r_k(\cdot)$ are Lipschitz continuous, we adopt the idea from Lipschitz bandits to discretize the continuous space of each p_k. With these ideas combined, we design a UCB-like algorithm that achieves a regret of $\underline{\mathcal{O}(KT^{2/3}(\ln T)^{1/3})}$.

With the solution of action-dependent bandits, we explore the general *history-dependent bandits* with impact functions following Equation (3.2) (Section 3.5) The main idea is to divide total time rounds into phases, and then selecting the same actions in each phase to smooth out impacts of historically made actions, which will then help reduce the problem to an action-dependent one. One challenge is to construct appropriate confidence bound and adjust the length of each phase to account for the historical action bias. With a careful combination with our results for action-dependent bandits, we present an algorithm which can also achieve a regret of the order $\underline{\tilde{\mathcal{O}}(KT^{2/3})}$. We further proceed to show that this bound is tight and provide numerical experiments.

3.4 Action-Dependent Bandits

In this section, we study action-dependent bandits, in which the impact function $\mathbf{f}(t) = \mathbf{p}(t)$, corresponding to $\gamma = 0$ in Equation (3.2). Our algorithm starts with a discretization over the space $\mathcal{P}$. Formally, we uniformly discretize $[0, 1]$ for each base arm into intervals of a fixed length ϵ, with carefully chosen ϵ such that $1/\epsilon$ is an positive integer.[8] Let $\mathcal{P}_\epsilon$ be the space of discretized meta arms, i.e., for each $\mathbf{p} = \{p_1, \ldots, p_K\} \in \mathcal{P}_\epsilon$, $\sum_{k=1}^{K} p_k = 1$ and

[8]Smarter discretization generally does not lead to better regret bounds [108].

$p_k \in \{\epsilon, 2\epsilon, \ldots, 1\}$ for all k. Let $\mathbf{p}^*_\epsilon := \sup_{\mathbf{p}\in\mathcal{P}_\epsilon} \sum_{k=1}^{K} p_k \cdot r_k(p_k)$ denote the optimal strategy in discretized space $\mathcal{P}_\epsilon$. After a meta arm $\mathbf{p}(t) = \{p_1(t), \ldots, p_K(t)\} \in \mathcal{P}_\epsilon$ is selected, a base arm $a_t \in [K]$ drawn according to the distribution $\mathbf{p}(t)$ will be realized. From now, we use $\tilde{r}_t$ to denote the realization of corresponding reward. The learner observes the realization of a_t and receives the instantaneous reward $\tilde{r}_t(p_{a_t}(t), a_t)$, but does not observe the rewards of other base arms. In the following discussion, we omit the second parameter and use $\tilde{r}_t(p_{a_t}(t))$ to denote $\tilde{r}_t(p_{a_t}(t), a_t)$ when it is clear from the context. We use importance weighting to construct the unbiased realized reward for each of the K elements in $\mathbf{p}$:

$$\hat{r}_t(p_k(t)) = \begin{cases} \tilde{r}_t(p_k(t))/p_k(t), & a_t = k \text{ and } p_k(t) \neq 0 \\ 0. & a_t \neq k \text{ or } p_k(t) = 0 \end{cases} \tag{3.5}$$

Since the probability of $a_t = k$ is $p_k(t)$, it is easy to see that $\mathbb{E}[\hat{r}_t(p_k(t))] = \mathbb{E}[\tilde{r}_t(p_k(t))]$. Given the importance-weighted rewards $\{\hat{r}_t(p_k(t))\}$, we re-frame our problem as choosing a K-dimensional probability measure (one value for each base arm). In particular, for each base arm k, p_k will take the value from $\{\epsilon, 2\epsilon, \ldots, 1\}$, and we refer to p_k as the *discretized arm.*

Remark 3.4.1. *The above importance-weighting technique enables us to "observe" samples of $r_k(p_k)$ for all base arms k when selecting $\mathbf{p} = \{p_1, \ldots, p_K\}$. This technique helps to bridge the gap between the interpretation of whether $\mathbf{p}$ is a probability distribution or an allocation over base arms. Our following techniques can be applied in either interpretation.*

By doing so, our problem is now similar to combinatorial bandits, in which we are choosing K discretized arms and observe the corresponding rewards. Below we describe our UCB-like algorithm based on the reward estimation of discretized arms. We define the set $\mathcal{T}_t(p_k) = \{s \in [t] : p_k \in \mathbf{p}(s)\}$ to record all the time steps such that the deployed meta arm $\mathbf{p}(s)$ contains the discretized arm p_k. We can maintain the empirical estimates of the mean

Algorithm 3 Action-Dependent UCB

Input: K, ϵ
Initialization: For each discretized arm, play an arbitrary meta arm such that this discretized arm is included (if the selection of the arm is not realized, then simply initialize its reward to 0; otherwise initialize it to the observed reward divided/reweighted by the selection probability).
for $t = \lceil K/\epsilon \rceil + 1, ..., T$ **do**
    Select $\mathbf{p}(t) = \arg\max_{\mathbf{p} \in \mathcal{P}_\epsilon} \texttt{UCB}_t(\mathbf{p})$ where $\texttt{UCB}_t(\mathbf{p})$ is defined as in (3.6).
    Draw an arm $a_t \sim \mathbf{p}(t)$ and observe its realized reward $\tilde{r}_t(p_{a_t}(t))$.
    Update the importance-weighted rewards $\{\hat{r}_t(p_k(t))\}$ as in (3.5) and update the empirical mean $\{\bar{r}_t(p_k(t))\}$ for each base arm as in (3.6).

reward for each discretized arm and compute the UCB index for each meta arm $\mathbf{p} \in \mathcal{P}_\epsilon$:

$$\bar{r}_t(p_k) = \frac{\sum_{s \in \mathcal{T}_t(p_k)} \hat{r}_s(p_k)}{n_t(p_k)}, \quad \texttt{UCB}_t(\mathbf{p}) = \sqrt{\frac{K^2 \ln \sqrt{K} t}{\min_{p_k \in \mathbf{p}} n_t(p_k)}} + \sum_{p_k \in \mathbf{p}} p_k \cdot \bar{r}_t(p_k), \tag{3.6}$$

where $n_t(p_k)$ is the cardinality of set $\mathcal{T}_t(p_k)$. With the UCB index in place, we are now ready to state our algorithm in Algorithm 3. The next theorem provides the regret bound of Algorithm 3.

Theorem 3.4.1. *Let $\epsilon = \Theta((K \ln(\sqrt{K}T)/T)^{1/3})$. The regret of Algorithm 3 (with respect to the optimal arm in non-discretized $\mathcal{P}$) is upper bounded as follows:* $\mathrm{Reg}(T) = \mathcal{O}\big(K^{4/3} T^{2/3} (\ln(\sqrt{K}T))^{1/3}\big)$.

Discussions. Our techniques have close connections to Lipschitz bandits [36, 123] and combinatorial bandits [28, 29]. Given the Lipschitz property of $r_k(\cdot)$, we are able to utilize the idea of Lipschitz bandits to discretize the strategy space and achieve sublinear regret with respect to the optimal strategy in the non-discretized strategy space. Moreover, we achieve a significantly improved regret bound by utilizing the connection between our problem setting and combinatorial bandits. In combinatorial bandits, the learner selects K actions out of action space $\mathcal{M}$ at each time step, where $|\mathcal{M}| = \Theta(K/\epsilon)$ in our setting. Directly

applying state-of-the-art combinatorial bandit algorithms [29] in our setting would achieve an instance-independent regret bound of $\mathcal{O}\big(K^{3/4}T^{3/4}(\ln T)^{1/4}\big)$, while we achieve a lower regret of $\mathcal{O}\big(KT^{2/3}(\ln T)^{1/3}\big)$. The reason for our improvement is that, for each base arm, regardless of which probability it was chosen, we can update the reward of the base arm, which provides information for all meta arms that select this arm with a different probability. This reduces the exploration and helps achieving the improvement. In addition to the above improvement, we would like to highlight that another of our main contributions is to extend the action-dependent bandits to the problem of history-dependent bandits, as discussed in **Section 3.5**.

Another natural attempt to tackle our problem is to apply EXP3 [10], which achieves sublinear regret even when the arm reward is generated adversarially. However, we would like to note that the optimal policy in our setting could be a mixed strategy, while the "sublinear" regret of EXP3 is with respect to a fixed strategy, and therefore it implies a linear regret in our setting.

3.5 History-Dependent Bandits

We now describe how to utilize our results for action-dependent bandits to solve the history-dependent bandit learning problem, with the impact function specified in Equation (3.2). The crux of our analysis is the observation that, in history-dependent bandits, if the learner keeps selecting the same strategy $\mathbf{p}$ for a long enough period of time, the expected one-shot utility will be approaching the utility of selecting $\mathbf{p}$ in the action-dependent bandits. More specifically, suppose after time t, the current action impact for all arms is $\mathbf{f}(t) = \mathbf{p}^{(\gamma)}(t) = \{p_1^{(\gamma)}(t), \ldots, p_K^{(\gamma)}(t)\}$. Assume that the learner is interested in learning about the utility of selecting $\mathbf{p} = \{p_1, \ldots, p_K\}$ next. Since the rewards are influenced by $\mathbf{f}(t)$, selecting $\mathbf{p}$ at

time $t+1$ does not necessarily give us the utility samples at $U(\mathbf{p})$. Instead, the learner can keep pulling this meta arm for a non-negligible s consecutive rounds to ensure that $\mathbf{f}(t+s)$ approaches $\mathbf{p}$. Following this idea, we decompose the total number of time rounds T into $\lfloor T/L \rfloor$ phases which each phase is associated with L rounds. We denote $m \in [1, \ldots, \lfloor T/L \rfloor]$ as the phase index and $\mathbf{p}(m)$ as the selected meta-arm in the m-th phase. To summarize the above phased-learning template:

- In each phase m, we start with an *approaching stage*: the first s_a rounds of the phase. This stage is used to "move" $\mathbf{f}(t+s)$ with $1 \leq s \leq s_a$ towards to $\mathbf{p}$.
- In the second stage, namely, *estimation stage*, of each phase: the remaining $L - s_a$ rounds. This stage is used for collecting the realized rewards and estimating the true reward mean on action $\mathbf{p}$.
- Finally, we leverage our tools in action-dependent bandits to decide what meta arm to select in each phase.

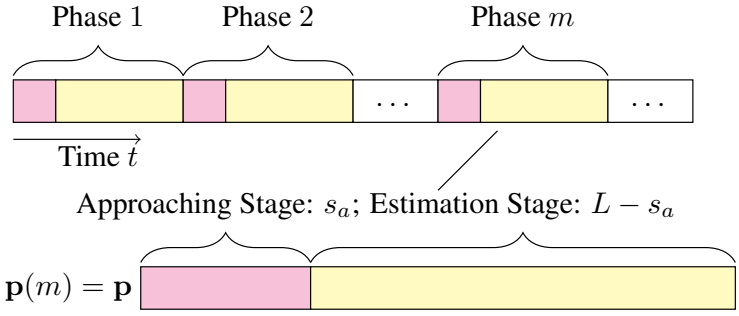


Figure 3.1: We deploy $\mathbf{p}$ for all rounds in m-th phases, therefore, we use $\mathbf{p}(m) = \mathbf{p}$ to represent $\mathbf{p}(t) = \mathbf{p}$ for simplicity.

Note that even if we keep pulling the arm k with the constant probability p_k in the approaching stage, the action impact in the estimation stage is not exactly the same as meta arm we want to learn, i.e., $\mathbf{f}(t+s) \neq \mathbf{p}$ for $s \in (s_a, L]$, due to the finite length of the stage. However, we

Algorithm 4 Reduction Template

Input: K, T; $\gamma, \epsilon, \rho \in (0,1)$, s_a.
Input: A bandit algorithm $\mathcal{A}$: History-Dependent UCB (Algorithm 5).
Split all rounds into consecutive phases of $L = s_a/(1-\rho)$ rounds each.
for $m = 1, \ldots$ **do**
    Query algorithm $\mathcal{A}$ for its meta arm selection $\mathbf{p}(m) = \mathbf{p}$.
    Each phase is separated into two stages:
     1). Approaching stage: $t = L(m-1)+1, \ldots, L(m-1)+s_a$;
     2). Estimation stage: $t = L(m-1)+s_a+1, \ldots, Lm$.
    for $t = L(m-1)+1, \ldots, L(m-1)+s_a$ **do**
        Deploy the meta arm $\mathbf{p}$.
    for $t = L(m-1)+s_a+1, \ldots, Lm$ **do**
        Deploy the meta arm $\mathbf{p}$;
        Collect the realized rewards $\tilde{r}_t$ to estimate the mean reward as in (3.7).
    Update $\overline{U}_t^{\texttt{est}}(\mathbf{p})$ as in (3.7).

can guarantee all $\mathbf{f}(t+s)$ for $s \in (s_a, L]$ is close enough to $\mathbf{p}$ by bounding its approximation error w.r.t $\mathbf{p}$. The above idea enables a more general reduction algorithm that is compatible with any bandit algorithm that solves the action-dependent case. Let $\underline{\rho = (L - s_a)/L}$ be the ratio of number of rounds in estimation stage of each phase. We present this reduction in Algorithm 4 and a graphical illustration in Figure 3.1.

3.5.1 History-Dependent UCB

In this section, we show how to utilize the reduction template to achieve a $\tilde{\mathcal{O}}(KT^{2/3})$ regret bound for history-dependent bandits. We first introduce some notations. For each discretized arm p_k, similar to action-dependent case, we define $\underline{\Gamma_m(p_k)} := \{s : \underline{s \in ((i-1)L + s_a, iL]}$ where $p_k \in \mathbf{p}(i), \forall i \in [m]\}$ as the set of all time indexes till the end of phase m in estimation stages such that arm k is pulled with probability p_k. We define the following empirical $\bar{r}_m^{\texttt{est}}(p_k)$

computed from our observations and the empirical utility $\overline{U}_m^{\texttt{est}}(\mathbf{p})$: [9]

$$\bar{r}_m^{\texttt{est}}(p_k) = \frac{1}{n_m^{\texttt{est}}(p_k)} \sum_{s \in \Gamma_m(p_k)} \hat{r}_s(p_k^{(\gamma)}(s)), \quad \overline{U}_m^{\texttt{est}}(\mathbf{p}) = \sum_{p_k \in \mathbf{p}} p_k \cdot \bar{r}_m^{\texttt{est}}(p_k), \tag{3.7}$$

where $n_m^{\texttt{est}}(p_k) := |\Gamma_m(p_k)|$ is the total number of rounds pulling arm k with probability p_k in all estimation stages, and $\hat{r}_s(p_k^{(\gamma)}(s))$ is defined similarly as in Equation (3.5). We use the smoothed-out frequency $\{p_k^{(\gamma)}(s)\}_{s \in \Gamma_m(p_k)}$ in the estimation stage as an approximation for the discounted frequency right after the approaching stage.

We compute our `UCB` for each meta arm at the end of each phase. We define and compute $\underline{\texttt{err} := K\gamma^{s_a}(L^* + 1)}$, the approximation error incurred after our attempt to smooth out the historical action impact. With these preparations, we present the phased history-dependent UCB algorithm (in companion with Algorithm 4) in Algorithm 5. The main result of this section is given as follows:

Algorithm 5 History-Dependent UCB

1: Construct `UCB` for each meta arm $\mathbf{p} \in \mathcal{P}_\epsilon$ at the end of each phase $m = 1, 2, \ldots,$ as follows:

$$\texttt{UCB}_m(\mathbf{p}) = \overline{U}_m^{\texttt{est}}(\mathbf{p}) + \texttt{err} + 3\sqrt{\frac{K^2 \ln\left(\sqrt{K}L\rho\right)}{\min_{p_k \in \mathbf{p}} n_m^{\texttt{est}}(p_k)}}.$$

2: Select $\mathbf{p}(m+1) = \arg\max_{\mathbf{p}} \texttt{UCB}_m(\mathbf{p})$ with ties breaking equally.

Theorem 3.5.1. *For any constant ratio $\rho \in (0,1)$ and $\gamma \in (0,1)$, let $\epsilon = \Theta((K \ln(\sqrt{K}T\rho)/(T\rho))^{1/3})$ and $s_a = \Theta(\ln(\epsilon^{1/3}/K)/\ln\gamma)$. The regret of Algorithm 4 with Algorithm 5 as input bounds as follows:* $\mathrm{Reg}(T) = \mathcal{O}\big(K^{4/3}T^{2/3}\left((\ln(\sqrt{K}T\rho))/\rho\right)^{1/3}\big)$.

[9] `est` in superscript stands for `est`timation stage.

For a constant ratio ρ, we match the optimal regret order for action-dependent bandits. When γ is smaller, the impact function "forgets" the impact of past-taken actions faster, therefore less rounds in approaching stage would be needed (see s_a's dependence in γ) and this leads to larger ρ.

Remark 3.5.1. *The dependence of our regret on the phase length L is encoded in ρ. When implementing our algorithm,we calculate L via s_a given the ratio ρ. We also run simulations of our algorithm on different ratios ρ, the results show that the performance of our algorithm are not sensitive w.r.t. specifying ρs - in practice, we do not require the exact knowledge of ρ, instead we can afford to use a rough estimation of its upper bound to compute L.*

3.5.2 Extension to General Impact Functions

So far, we discuss settings when the impact function is specified as in Equation (3.2). However, the same technique we presented earlier can be applied for a more general family of impact functions. In particular, as long as the impact function converges after the learner keeps selecting the same action, our result holds. To be more precise, we only require $\mathbf{f}(t)$ to satisfy the condition $|f_k(t+s) - g(p_k)| \leq \gamma^s, \gamma \in (0,1)$ when the learner keeps pulling arm k with probability p_k for s round. The function $g(\cdot)$ can be an arbitrary monotone function as long as it is continuous and differentiable, for example: $g(x) = x$. In fact, the property of $\mathbf{f}(t)$ is only used when we estimate how close $\mathbf{f}$ is to $g(\mathbf{p})$ after the approaching stage with repeatedly selecting $\mathbf{p}$. For a different $\mathbf{f}(t)$, we define new reward mean functions $r'_k(\cdot) = r_k(g(\cdot))$, and tune parameters ϵ and s_a accordingly to bound the approximation error for $\left|U(\mathbf{p}) - \overline{U}_m^{\texttt{est}}(\mathbf{p})\right|$ (change the Lipschitz constant). This way we can follow the same algorithmic template to achieve a similar regret.

Moreover, we do not require exact knowledge of the impact function $\mathbf{f}(t)$. We only require the impact functions to satisfy the above conditions for our algorithms/analysis to hold. With

the same arguments, while we assume the reward function $r_k(\cdot)$ is fed with the same impact function $\mathbf{f}$, our formulation generalizes to different impact functions for $r_k(\cdot)$, as long as these impact functions are able to stabilize given a consecutive adoption of the desired action.

3.6 Matching Lower Bounds

For both action- and history-dependent bandit learning problems, we have proposed algorithms that achieve a regret bound of $\tilde{\mathcal{O}}(KT^{2/3})$. We now show the above bounds are order-optimal with respect to K and T, i.e., the lower bounds of our action- and history-dependent bandits are both $\Omega(KT^{2/3})$, as summarized below.

Theorem 3.6.1. *Let $T > 2K$ and $K \geq 4$, there exist problem instances that for our action- and history-dependent bandits, respectively, the regret for any algorithm $\mathcal{A}$ follows:* $\inf_{\mathcal{A}} \mathrm{Reg}(T) \geq \Omega(KT^{2/3})$.

3.7 Conclusion and Future Work

We explore a multi-armed bandit problem in which actions have delayed impacts to the arm rewards. We propose algorithms that achieve a regret of $\tilde{\mathcal{O}}(KT^{2/3})$ and provide a matching lower regret bound of $\Omega(KT^{2/3})$. Our results complement the bandit literature by exploring the action history dependent biases in bandits. While our model have its limitations, it captures an important but relatively under-explored angle in algorithmic fairness, the long-term impact of actions in sequential learning settings. We hope our study will open more discussions along this direction.

Chapter 4

Human Behavior Modeling – Bayesian Rationality in Information Design

We study the problem of information design in human-in-the-loop systems, in which an informed *sender* (i.e., the system) aims to influence a *receiver* (i.e., humans in the system) in making decisions through designing information disclosure strategies. This problem is ubiquitous in our daily life. For example, online retailers might highlight a subset of product features to influence the buyers to make the purchases. Recommendation systems might selectively display other users' ratings to persuade users to take the recommendation. Public health officials might decide how to present vaccine information to encourage the general public to take vaccines to curb the pandemic. There have been various research efforts devoted to this problem from both economics [63, 68, 128, 147] and computer science [51, 54]. Among the growing literature on the study of information design, the model of Bayesian persuasion proposed by [97] is one of the most prominent ones and has inspired a body of studies. In this work, we also build on top of the framework of Bayesian persuasion and aim to relax the restrictive assumptions in their model.

In Bayesian persuasion, there are two players, a sender and a receiver.[10] The state of nature is randomly drawn from a distribution, with the prior known to both players. The sender has access to the realization of the state while the receiver does not. The sender can utilize the information advantage and selectively disclose information to the receiver to influence the receiver. Based on the prior information of the state and the information revealed by the sender, the receiver can take an action to maximize her own payoff, which depends on both the action and the realized state. The sender's objective also depends on the receiver's action, and the goal of the sender is to choose an information disclosure policy – which is determined before the state realization and is known to the receiver – to maximize his objective.

As an illustrative example, consider the scenario in which an online retailer (the sender) would like to persuade a buyer (the receiver) to make the purchase. The retailer's products are directly coming from the factory, and the product quality (the state of nature) is drawn from a distribution with known prior. The buyer's utility depends on both her purchase decision and the realized product quality, and the retailer's utility depends on the buyer's purchase decision. In order to persuade the buyers to purchase, the retailer can commit to perform (noisy) product inspections to reveal some information of the product quality (e.g., the inspection might signal the product quality is satisfactory with 80% chance if the quality of the product is indeed satisfactory and signal the product quality is unsatisfactory with 90% chance if the quality is indeed unsatisfactory). The retailer's goal is to find the optimal inspection policy to maximize the probability of selling the product to the buyer.

While Bayesian persuasion provides an elegant framework to address the above information design problem, it has made some restrictive assumptions. In particular, the receiver is assumed to be Bayesian rational, i.e., the receiver is able to form a posterior by incorporating

[10] In this paper, we use "he" to denote the sender and "she" to denote the receiver. Moreover, this work is motivated by scenarios of designing information for a population of users. Therefore, we use the term "receiver" to refer to a population of users, and sometimes we explicitly use the term "receivers".

the prior information and the signals revealed by the sender in a Bayesian manner, and then choose the action that maximizes her expected utility. However, as consistently observed in empirical studies [11, 121, 131, 166], humans often systematically deviate from being Bayesian or being rational.

In this work, we explore the problem of information design with non-Bayesian-rational receiver. We develop an alternative framework to Bayesian persuasion that incorporates discrete choice model [130, 163, 169] and probability weighting [145, 150, 181] to model non-Bayesian-rational receiver. We formulate the problem of solving the optimal information disclosure policy under our model and characterize the properties of the optimal information disclosure policy. To showcase the difference of the two frameworks, we investigate the information policies derived from both frameworks in a simple baseline setting. We then conduct behavioral experiments on Amazon Mechanical Turk with 400 workers to examine the two frameworks. Our results demonstrate that our framework better aligns with the behavior with real-world humans and lead to a better information disclosure policy.

4.1 Related Work

Our work builds on top of the seminal work of Bayesian persuasion [97], which initiated a rich theoretical literature on communication game in which a sender can design information to persuade a receiver to take certain actions. Their work has inspired an active line of research in information design. [e.g., see the recent surveys by 17, 95]. In this work, we extend this line of research on information design and focus on relaxing the assumption that the receiver is Bayesian rational through both developing an alternative framework and empirically examining human behavior.

Human models for decision making. In the problem of information design, the receiver needs to incorporate the information provided by the sender and make decisions accordingly. We can decompose this decision making process into two stages: 1) belief updating: how the receiver processes the information and updates her beliefs, and 2) decision making under uncertainty: how the receiver makes decisions with the updated belief. Since we are interested in settings in which receivers are human beings, in the following, we discuss existing human models for decision making in the above two stages.

For belief updating, Bayesian models have been the prominent model in algorithmic works [27, 70, 168]. However, it has also been consistently and widely observed in empirical studies that humans often deviate from being Bayesian [11, 93, 121, 131, 166, 171]. While there have been some alternative models in how humans process information to form their beliefs [124, 132, 145, 150, 157, 181], they are not widely adopted in algorithmic frameworks.

For decision-making under uncertainty, the commonly-used assumption is expected utility theory [137] which assumes humans take actions to maximize their expected utility. There is again a substantial body of work in behavioral economics in studying the systematic deviations of human behavior from expected utility theory. One important theory that summarizes these systematic biases is the *prospect theory* by [94]. Another commonly used theory, that accounts for the inherent randomness of human decision making by incorporating noises in the utility, is the discrete choice model [130, 163, 169].

In this work, to account for the receiver's deviation from being Bayesian rational, we adopt probability weighting function [145, 150, 181] for belief updating and discrete choice model [130, 163, 169] for decision making in our framework. We also examine whether our framework aligns with real-world human behavior through behavioral experiments. In addition, there have been some recent works that aim to incorporate human behavioral models in the computational

framework. For example, [105] and [106] study the planning for time-consistent agents in an environment characterized by a graphical model. [184] investigate the design of decision making environment for agents with decision biases. [1] explore soliciting data from strategic agents whose data is correlated with the cost for releasing their data. Our work aligns with this line of research that incorporates realistic human behavioral models in computation.

Behavioral experiments in information design. While there is a rich line of research on Bayesian persuasion, the amount of works on empirically investigating human behavior in information design is limited [5, 6, 57]. Among these works, [6] incorporate reciprocity into the standard persuasion setting and conduct a laboratory experiment to validate their model on reciprocity. [5] propose a unified framework to investigate the theoretical parallelism between information and mechanism design. [57] empirically examine different information design methods, including communications via cheap-talk, disclosure of verifiable information, and Bayesian persuasion. Our work departs from the above literature as we investigate the fundamental assumption of Bayesian rationality in human behavior. We create a decision-making scenario where the receiver is required to make a decision after seeing a signal realized according to some information disclosure policy to empirically measure how humans update their beliefs and make decisions.

Another closely-related work to ours is the one by [34] who also relax the Bayesian assumption of receiver's behavior in persuasion. They theoretically study how receiver's mistakes in probabilistic inference impact optimal persuasion and characterize a large class of belief updating rules that the concavification method developed by [97] can still be applied. However, their work focuses on theoretical characterization and the receiver is still assumed to be an expected utility maximizer. While in our model, we further relax this assumption by using a discrete choice model and empirically examine our models.

4.2 Model

In this section, we formalize the frameworks for the information design problem. We first describe the standard Bayesian persuasion framework that assumes Bayesian rational receiver. We then introduce our framework that relaxes the Bayesian rational assumption. In the later section, we compare the two frameworks on a simple baseline setting with two states and binary actions to showcase the differences of the frameworks. This simple baseline setting also motivates the design of our real-world behavioral experiments described in our experiment section.

4.2.1 Standard Framework: Bayesian Persuasion

We first describe the standard setting of Bayesian persuasion [97]. In this setting, there are two players: a sender and a receiver. The goal of the sender is to design an information disclosure policy to persuade the receiver in taking actions to maximize the sender's objective.

Let the (payoff-relevant) state of the world be θ, which is drawn from a finite set Θ according to a prior distribution $\mu_0 \in \Delta(\Theta)$. The prior is common knowledge to all players. The receiver's utility is characterized by the function $u^R(a, \theta)$ which depends on the action she takes $a \in \mathcal{A}$ from a compact action set $\mathcal{A}$ and the state θ. The sender's utility is characterized by the function $u^S(a, \theta)$ that also depends on the receiver's action and the state.

Before observing the realization of the state, the sender can choose an information disclosure policy (π, Σ), which consists of a finite signal space Σ and a family of conditional distributions $\{\pi(\cdot|\theta)\}_{\theta\in\Theta}$ over $\sigma \in \Sigma$. This information disclosure policy is known to the receiver and specifies how the sender discloses information to the receiver. In particular, when a state $\theta \in \Theta$ is realized, the sender can observe the state but the receiver cannot. To influence the

receiver's decision, the sender sends a signal σ, drawing from the conditional distribution $\pi(\cdot|\theta)$ specified in the information disclosure policy, to the receiver. The receiver forms her beliefs on the state of the world based on the prior and the signal provided by the sender. She then takes an action to maximize her own payoff.

In the Bayesian persuasion setting, it is assumed that the receiver is Bayesian rational, i.e., she updates her beliefs in a Bayesian manner and is an expected utility maximizer. Formally, upon seeing the signal realization σ from the sender, the receiver updates her belief, denoted by $\mu \in \Delta(\Theta)$, by applying Bayes' rule:

$$\mu(\theta|\sigma) = \frac{\pi(\sigma|\theta)\mu_0(\theta)}{\sum_{\theta' \in \Theta} \pi(\sigma|\theta')\mu_0(\theta')}. \tag{4.1}$$

Given the posterior belief μ, the receiver then chooses an action $a^* = a^*(\mu)$ that maximizes her expected payoff: $a^* \in \arg\max_{a \in \mathcal{A}} \sum_{\theta \in \Theta} u^R(a, \theta)\mu(\theta)$.[11] As a key insight by [97], the above two assumptions on the receiver's behavior allow the sender to reduce the problem of designing information disclosure policy to choosing a distribution of posterior beliefs that respects Bayes rule. Furthermore, a distribution $\tau \in \Delta(\Delta(\Theta))$ of posteriors can arise if and only if it is Bayes-plausible, i.e.,

$$\mathop{\mathbb{E}}_{\mu \sim \tau}[\mu] = \mu_0. \tag{4.2}$$

Therefore, it is without loss of generality to assume the set of available information disclosure policy to the sender is the set of Bayes-plausible distributions of posterior beliefs. By formulating the sender's *direct utility* $u^S(a, \theta)$, a function of the receiver's action, to an

[11] In the persuasion literature, most work consider sender-preferred Subgame Perfect Equilibrium, where the receiver chooses the sender-preferred action when there are ties.

indirect utility $\widehat{u}^S(\mu)$, a function of Bayesian posteriors, the standard *concavification* argument can be applied to derive the optimal information design.

4.2.2 Our Framework: Persuading Non-Bayesian-Rational Receiver

In contrast to the assumptions made in Bayesian persuasion, the receiver may, in practice, exhibit systematic biases both in probabilistic inferences and in decision making. In the following discussion, we first incorporate the discrete choice model and probability weighting to model non-Bayesian-rational receiver. We then formulate the optimal information design problem under this receiver model.

Modeling non-Bayesian-rational receiver. We first relax the assumption that the receiver is an expected utility maximizer but still assume the receiver is Bayesian in updating the belief. Specifically, we leverage the discrete choice model [131], a commonly-used alternative of expected utility theory, to characterize the receiver's behavior when making her decision.

To provide informal intuitions, in expected utility theory, the receiver takes an action that maximizes her expected utility. When there is no ties in action utility, this action choice is deterministic. On the other hand, the discrete choice model accounts for the inherent randomness in human decision making and models the decision as a probabilistic process. Specifically, in the discrete choice model, for each action $a \in \mathcal{A}$ the receiver can take, we add noise $\varepsilon(a)$ into the receiver's utility for taking action a. The receiver then takes an action that maximizes this noisy version of the utility. This noise captures several realistic aspects of human decision making, e.g., when there are additional inherent characteristics in the receiver's utility estimation that we cannot model, or when receiver is drawn from a population and individual differences need to be accounted for.

More formally, let $\mu \in \Delta(\Theta)$ denote the receiver's posterior induced by some signal realization. We define $\widehat{u}^R(a|\mu)$ as the noise-free expected utility for the receiver to choose action $a \in \mathcal{A}$ given the posterior belief μ, which can be written as $\widehat{u}^R(a|\mu) := \mathbb{E}_{\theta\sim\mu}\left[u^R(a,\theta)\right] = \sum_{\theta\in\Theta} u^R(a,\theta) \cdot \mu(\theta)$. In discrete choice model, the receiver takes actions based on the noisy version of the utility $\widetilde{u}^R(a|\mu)$, which can be written as

$$\widetilde{u}^R(a|\mu) := \beta \cdot \widehat{u}^R(a|\mu) + \varepsilon(a), \tag{4.3}$$

where $\epsilon(a)$ is the added noise and β is a parameter that tunes the relative strength of observable utility and the noises, e.g., when $\beta \to \infty$, the noise is negligible and the discrete choice model reduces to the standard expected utility theory.

Different choices of distributions of $\varepsilon(a)$ lead to different discrete choice models. In this work, we follow the commonly used Multinomial Logit (MNL) [129] and assume that each $\varepsilon(a)$ is distributed independently, identically extreme value, where the CDF follows $F(\varepsilon(a)) = \exp(-\exp(-\varepsilon(a)))$.

Lemma 4.3.1 ([129])**.** *Given posterior belief μ, the probability that receiver chooses action a can then be derived as*

$$\Pr(a|\mu) = \frac{\exp\left(\beta\widehat{u}^R(a|\mu)\right)}{\sum_{a'}\exp\left(\beta\widehat{u}^R(a'|\mu)\right)}. \tag{4.4}$$

Proof. Define $v^R(a|\mu) = \beta \cdot \widehat{u}^R(a|\mu)$. By definition,

$$\Pr(a|\mu) = \Pr\left(\widetilde{u}^R(a|\mu) > \widetilde{u}^R(a'|\mu), \quad \forall a' \neq a\right) = \Pr\left(\varepsilon(a') < \varepsilon(a) + v^R(a|\mu) - v^R(a'|\mu), \quad \forall a' \neq a\right).$$

Since the ε's are independent, this cumulative distribution over all $a' \neq a$ is the product of the individual cumulative distributions:

$$\Pr(a|\mu, \varepsilon(a)) = \prod_{a' \neq a} \exp\left(-\exp(-(\varepsilon(a) + v^R(a|\mu) - v^R(a'|\mu)))\right).$$

Since $\varepsilon(a)$ is not given, and so the choice probability is the integral of $\Pr(a|\mu, \varepsilon(a))$ over all values of $\varepsilon(a)$ weighted by its density

$$\Pr(a|\mu) = \int \prod_{a' \neq a} e^{-e^{-(\varepsilon(a)+v^R(a|\mu)-v^R(a'|\mu))}} e^{-\varepsilon(a)} e^{-e^{-\varepsilon(a)}} d\varepsilon(a).$$

Finally, by computing the integral over $\epsilon(a)$, we can obtain the closed-form expression (4.4).

□

With the above lemma, we have a closed-form formulation specifying the distribution of actions the receiver will choose given her posterior belief under discrete choice model. We now relax the assumption that the receiver might not be Bayesian in updating her beliefs.

To account for non-Bayesian belief updating, we utilize the ideas of probability weighting and introduce a non-decreasing *prior-specific* probability distortion function $\omega(\cdot; \mu_0) : \Delta(\Theta) \to \Delta(\Theta)$ to capture the receiver's final belief on making her decision. This formulation helps explain the human biases in over-weighting or under-weighting the prior when performing beliefs updates. Now one can derive the following choice probabilities by incorporating the distorted posterior $\omega(\cdot; \mu_0)$ into (4.4):

$$\Pr(a|\omega(\mu; \mu_0)) = \frac{\exp\left(\beta \widehat{u}^R(a|\omega(\mu; \mu_0))\right)}{\sum_{a'} \exp\left(\beta \widehat{u}^R(a'|\omega(\mu; \mu_0))\right)}. \tag{4.5}$$

Many parametric forms of the probability weighting function have been proposed [145, 150, 170, 181]. For example, an affine probability distortion function [52, 60, 172] specifies a distorted posterior that falls in between a reference belief $\mu^* \in \Delta(\Theta)$ and Bayesian posterior μ: $\omega(\mu|\mu_0) = \gamma\mu^* + (1-\gamma)\mu$ where μ^* is allowed to vary with μ_0 and $\gamma \in [0,1]$ is a constant.

Optimal information design. With the modeling of the receiver, we now characterize the sender's optimal information design. To simplify the exposition, we mainly state the analysis when the receiver's behavior follows the discrete choice model defined in (4.4). The analysis for the model including probability weighting is similar. For notation simplicity, let $p(a|\mu) := \Pr(a|\mu)$ denote the the probability for the receiver to choose action $a \in \mathcal{A}$ when the posterior μ is induced. With this expression, we are now ready to characterize the sender's optimal information design problem:

Theorem 4.3.2. *Let μ_0 be the prior. Assume the receiver's behavior follows* (4.4) *when μ is the posterior. The sender's problem is equivalent to*

$$\begin{aligned} \max_{\tau \in \Delta(\Delta(\Theta))} \quad & \mathop{\mathbb{E}}_{\mu\sim\tau}\left[\sum_{\theta\in\Theta}\mu(\theta)\sum_{a\in\mathcal{A}}p(a|\mu)u^S(a,\theta)\right] \\ s.t. \quad & \mathop{\mathbb{E}}_{\mu\sim\tau}[\mu] = \mu_0 \end{aligned} \tag{4.6}$$

Proof. Let $\nu(\mu) = \{p(a|\mu)\}_{a\in\mathcal{A}} \in \Delta(\mathcal{A})$. Given a posterior μ and the corresponding $\nu(\mu)$, we can compute the sender's indirect expected utility $\widehat{u}^S(\mu)$ as a function of μ:

$$\widehat{u}^S(\mu) = \mathop{\mathbb{E}}_{\theta\sim\mu}\left[\mathop{\mathbb{E}}_{a\sim\nu(\mu)}\left[u^S(a,\theta)\right]\right] = \sum_{\theta\in\Theta}\mu(\theta)\cdot\sum_{a\in\mathcal{A}}p(a|\mu)u^S(a,\theta).$$

Given the prior μ_0, an information disclosure policy π generates a distribution $\tau \in \Delta(\Delta(\Theta))$ over Bayesian posteriors. It is known that, should the receiver be Bayesian, a distribution τ of posteriors is feasible iff it is Bayes-plausible (4.2). Now the sender's expected utility can

be written as a function of the receiver's choices and the probability measure τ, we obtain the stated reformulation of the sender's problem. □

Note that the problem (4.6) can be further simplified when the sender's utility is state-independent, i.e., $u^S(a,\theta) = u^S(a), \forall\theta \in \Theta$, which is a common assumption in the persuasion literature. Indeed, we have the objective $\mathbb{E}_{\mu\sim\tau}\left[\sum_{a\in\mathcal{A}} p(a|\mu)u^S(a)\right]$ in (4.6). By writing the sender's problem as a function of the induced Bayesian posterior, then (4.6) can be addressed using the tools developed by [97]. In particular, for an arbitrary real-valued function $u : \Delta(\Theta) \to [0,1]$, let u^{cc} be the concave closure of u,

$$u^{\mathsf{cc}}(\mu) = \sup\{z|(\mu,z) \in \mathsf{co}(u)\}, \tag{4.7}$$

where $\mathsf{co}(u)$ is the convex hull of the graph of u.

Proposition 4.3.3. *The sender's expected utility under an optimal policy is $\widehat{u}^{\mathsf{cc}}(\mu_0)$, where $\widehat{u}$ is defined in* (4.7).

The above analysis can also be applied to deal with settings in which the receiver distorts the probabilities through a probability weighting function. In particular, the results in Theorem 4.3.2 still hold with the only difference being that the choice probabilities in (4.6) will accordingly correspond to (4.5). We can simplify the sender's problem (4.6) to the following optimization problem with a distorted Bayes-plausibility constraint:

Proposition 4.3.4. *Let μ_0 be the prior, μ be the Bayesian posterior and μ^R be the receiver's non-Bayesian posterior. Assuming the receiver's behavior follows* (4.5) *with the probability*

weighting function $\omega(\cdot|\mu_0) : \Delta(\Theta) \to \Delta(\Theta)$. *The sender's problem is equivalent to* [12]

$$\max_{\tau \in \Delta(\Delta(\Theta)\times\Delta(\Theta))} \quad \mathop{\mathbb{E}}_{(\mu,\mu^R)\sim\tau} \left[\sum_{\theta\in\Theta} \mu(\theta) \sum_{a\in\mathcal{A}} p\left(a|\mu^R\right) u^S(a,\theta) \right]$$
$$s.t. \qquad \mathop{\mathbb{E}}_{\mu^R\sim\tau^R} \left[\omega^{-1}\left(\mu^R|\mu_0\right)\right] = \mu_0,$$

where $\tau^R = \int_\mu \tau(\mu, \cdot) d\mu$.

Proof. Given the receiver's belief μ^R, let $\nu(\mu^R) = \{p(a|\mu^R)\}_{a\in\mathcal{A}} \in \Delta(\mathcal{A})$. Together with the Bayesian posterior μ, we have the following sender's indirect utility

$$\widehat{u}^S(\mu, \mu^R) = \mathop{\mathbb{E}}_{\theta\sim\mu} \left[\mathop{\mathbb{E}}_{a\sim\nu(\mu^R)} [u^S(a,\theta)] \right] = \sum_{\theta\in\Theta} \mu(\theta) \cdot \sum_{a\in\mathcal{A}} p(a|\mu^R) \mu^S(a,\theta).$$

Recall that μ^R is the result of the mapping of probability weighting function $\omega(\cdot|\mu_0)$ from the Bayesian posterior μ. As the mapping $\omega(\cdot|\mu_0)$ is invertible and μ satisfies Bayes-Plausibility, μ^R must satisfy $\mathbb{E}_{\mu^R\sim\tau^R}\left[\omega^{-1}\left(\mu^R|\mu_0\right)\right] = \mu_0$. Thus, we can achieve the above sender's reformulated optimization problem. □

Similarly, when the sender's utility is state-independent, sender's problem can be further simplified as $\max_{\tau\in\Delta(\Delta(\Theta))} \mathbb{E}_{\mu^R\sim\tau^R}\left[\sum_{a\in\mathcal{A}} p(a|\mu^R) u^S(a)\right]$ with the distorted Bayesian-plausibility constraint.

[12] Including probability weighting in our model is essentially the same as distorting updated beliefs proposed by [34]. We can show that a distorted version of Bayes-plausibility holds, and therefore the standard concavification technique to derive optimal information design can be applied.

4.4 A Baseline Setting with Two States and Binary Actions

In the above section, we formulate the information design problem for both the standard framework of Bayesian persuasion and our framework of persuading non-Bayesian-rational receiver. To instantiate the discussion and comparison, in this section, we consider a simple setting with two states and binary actions, a variant of the leading example in [97], to demonstrate the differences of the two frameworks. This setting also motivates our experiment design as presented in the next section.

Consider a world with two states $\Theta = \{\mathsf{X}, \mathsf{Y}\}$, where state X happens with probability $\mu_0 \in [0, 1]$ and state Y happens with probability $1 - \mu_0$. The receiver can choose from two actions $\mathcal{A} = \{a_\mathsf{X}, a_\mathsf{Y}\}$. The utility of the sender and the receiver both depend on the receiver's action and the realized state and have been summarized in Table 4.1.

Payoff	State X	State Y
Receiver chooses a_X	Receiver: 1. Sender: 1	Receiver: 0. Sender: 1
Receiver chooses a_Y	Receiver: 0. Sender: 0	Receiver: 1. Sender: 0

Table 4.1: Payoff structure.

In this payoff structure, the receiver aims to select the action that matches the state (i.e., select action $a_\mathsf{X}/a_\mathsf{Y}$ for state X/Y), while the sender wishes to persuade the receiver to select action a_X.

Optimal information design with Bayesian-rational receiver. In the following discussion, we use μ to denote the posterior probability of state X. If the receiver is Bayesian

rational, whenever the receiver sees a signal that induces a posterior $\mu \geq 0.5$, the receiver's best response is to choose action a_{X}. In other words, the receiver's response is a simple step function in posterior beliefs (the receiver chooses action a_{X} when $\mu \geq 0.5$ and action a_{Y} when $\mu < 0.5$). Given the receiver's behavior, the optimal information disclosure policy can be achieved with only 2 signals, represented using $\{\mathsf{R}, \mathsf{B}\}$, and the policy can be specified as below. [13]

Proposition 4.4.1 (Optimal policy assuming Bayesian rational receiver [97])**.** *When the prior $\mu_0 < 0.5$, an optimal information disclosure policy exists and satisfies:*

- *when state* X *is realized, always sends signal* R*;*
- *when state* Y *is realized, with prob.* $\frac{\mu_0}{1-\mu_0}$ *sends signal* R*, and with prob.* $1 - \frac{\mu_0}{1-\mu_0}$ *sends signal* B*.*

When $\mu_0 \geq 0.5$, an uninformative information disclosure policy is the optimal policy.

Below is the intuition of the optimal policy. When $\mu_0 \geq 0.5$, when deploying an uninformative information policy, the receiver's posterior is the same as prior, and she will always choose action a_{X}, and therefore an uninformative information policy is the optimal policy. When $\mu_0 < 0.5$, recall that the goal of the sender is to persuade the receiver to choose a_{X} when the prior of state Y is larger than half. In the optimal information policy, when the state is X, the sender wants to reveal the true information to encourage the receiver to choose a_{X}. When the state is Y, the sender wants to make the receiver have indifferent beliefs between the state to maximize the chance the receiver chooses a_{X}. The above policy generates two possible posteriors: $\mu = 0.5$ with probability $2\mu_0$ on seeing signal R, and $\mu = 0$ with probability $1 - 2\mu_0$ on seeing signal B.

[13] We choose $\{\mathsf{R}, \mathsf{B}\}$ as signal notations mainly for the consistency of our experiment presentation in our experiment section.

Optimal information design with non-Bayesian-rational receiver. When the receiver is not Bayesian rational, the receiver's probability of choose a_{X} is not a step-function of posterior μ as in Proposition 4.4.1. Instead, as described in our framework, it is a smoothed continuous function as below:

$$p(a = a_{\mathsf{X}}|\mu) = \frac{\exp(\beta\mu)}{\exp(\beta(1-\mu)) + \exp(\beta\mu)}. \tag{4.8}$$

We can also derive the sender's optimal information design when the receiver follows the model (4.8). In particular, since the sender obtains zero utility when the receiver chooses action a_{Y}, the sender's indirect utility $\widehat{u}^S(\mu)$ as a function of posterior μ is simply $\widehat{u}^S(\mu) = p(a = a_{\mathsf{X}}|\mu)$. A concavification argument allows us to characterize the following optimal information disclosure policy:

Proposition 4.4.2 (Optimal policy assuming non-Bayesian-rational receiver)**.** *Let $p(\mu) := p(a = a_{\mathsf{X}}|\mu)$ and let $\bar{\mu}$ be the unique solution of $\bar{\mu}p'(\bar{\mu}) = p(\bar{\mu}) - p(0)$. Given prior $\mu_0 \leq \bar{\mu}$, an optimal information disclosure policy exists and satisfies*

- *when state* X *is realized, always sends signal* R*;*
- *when state* Y *is realized, with prob.* $\frac{\mu_0(1-\bar{\mu})}{(1-\mu_0)\bar{\mu}}$ *sends signal* R*, and with other prob. sends signal* B*.*

When $\mu_0 > \bar{\mu}$, an uninformative information disclosure policy is the optimal policy.

The optimal information policy shares a similar structure as the one when the receiver is Bayesian rational (Proposition 4.4.1). However, the threshold $\bar{\mu}$, that characterizes when an uninformative policy is not optimal, and the probability for sending signal B when the realized state is Y are different and are influenced by the receiver model. Furthermore, as β in the receiver model (4.8) increases, the shape of $p(\mu)$, the probability for the receiver to

choose action a_{X} given posterior μ, is more towards a step function with breaking point at 0.5 and thus $\bar{\mu}$ is smaller. Intuitively, larger β implies that the impact of unobserved component $\varepsilon(a)$ is smaller on the receiver's utility, and thus the receiver is more towards an expected utility maximizer. The above discussion is graphically illustrated in Figure 4.1. The analysis when including the probability weighting is similar.

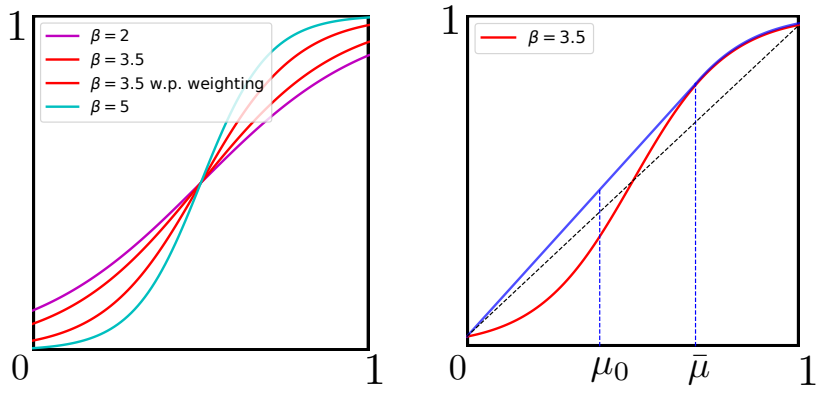


Figure 4.1: Left: Various shapes of $\widehat{u}^S(\mu)$ (or $p(\mu)$) and $\widehat{u}^S(\omega(\mu))$ (or $p(\omega(\mu))$) with an affine distorting function ω where $\gamma = 0.3, \mu^* = 0.5$. Right: Red line is the concavification $\widehat{u}^{\mathsf{cc}}(\mu)$ for $\widehat{u}^S(\mu)$.

4.5 Real-World Experiment

Our discussion in the previous sections demonstrates the different predictions on the receiver's behavior and the optimal information disclosure policy when we consider different receiver models. In this section, we describe the setup and results of our real-world behavioral experiments to examine these predictions. The experiment has been approved by IRB at Washington University.

In our experiment, we recruit online workers to answer a series of questions. In each question, workers are asked to perform a probabilistic-inference and decision-making task. We design

the questions in a way that we can control the prior and the information structure and then observe workers' corresponding actions. Moreover, given a prior and a realized signal from the information policy, we are able to derive the corresponding induced Bayesian posterior (calculated using Bayes rule). We are interested in examining the following two questions:

- **Q1:** Are workers Bayesian?

  To examine whether workers are Bayesian, we can design two scenarios that lead to the same induced posterior but have different priors and information policies. If workers are Bayesian, their decisions should depend only on the posterior, and we should observe the same worker behavior on the two scenarios.

- **Q2:** Are workers rational?

  To examine whether workers are rational, we can create scenarios that lead to different posteriors. If workers are rational, we should observe workers' behavior follows a step function over the induced posteriors.

4.5.1 Experiment Setup

We recruited 400 unique workers from MTurk, where each worker is required to complete 20 questions. We offer a \$0.5 base payment, and each worker may also receive a bonus payment of up to \$0.6 (the bonus rule will be explained shortly). The bonus amount is chosen to be large enough so workers are motivated to perform well. The average hourly rate is around \$12.15.

Task. Our goal is to evaluate the receiver's behavior. Therefore, we play the role of the sender and have all recruited workers play as the receiver. Each worker needs to complete 20 questions as described below.

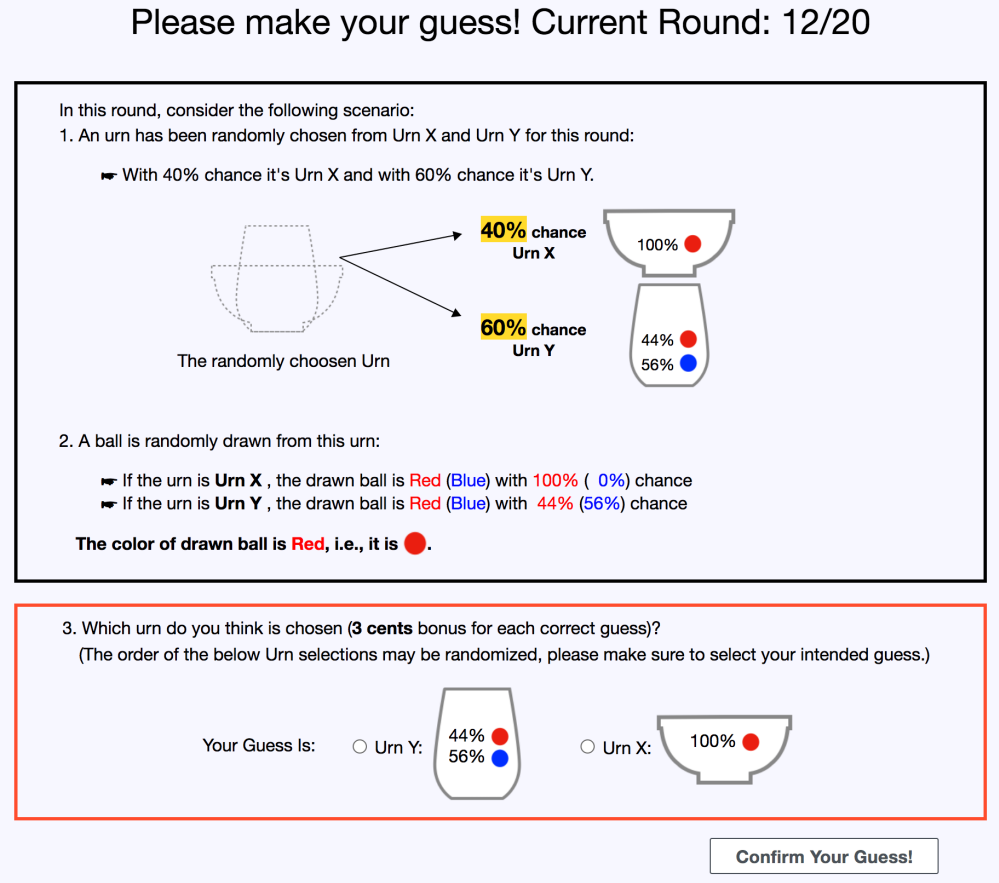


Figure 4.2: The task interface.

In each question, as shown in Figure 4.2, workers are informed that there are two urns, Urn X and Urn Y. At the beginning of the question, an urn is randomly drawn according to the prior distribution that is known to the workers. Each urn contains certain fraction of red balls and blue balls. The ball composition of each urn is also shown to workers. After an urn is realized, we choose a ball uniformly at random from this realized urn. The color of the drawn ball is then disclosed to the worker. Upon seeing the color, the worker is required to make guess on which urn is realized.

This experiment setup is designed to capture human decision-making process. The two urns represent the world state. The ball composition is the information disclosure policy. When seeing the realized ball, the workers update their prior beliefs (the prior of urn drawing)

with additional information (realized ball drawn according to the commonly known ball compositions in urns) and make decisions (guessing which is the realized urn).

Bonus rule. For each correct guess (i.e., worker's guess matches the realized urn), worker receives a bonus of \$0.03, thus each subject will receive at most \$0.6 in the game. The bonus for correct guess on Urn X and Urn Y is the same to match the setting in section about our baseline setting.

Treatment design. To answer our research questions, we conducted a randomized behavioral experiment. The experiment consists of two treatments, which differ in the prior distribution of the state. In the *high prior* treatment, we fixed the prior to be $(0.4, 0.6)$, while in the *low prior* treatment, the prior is fixed as $(0.2, 0.8)$. We then design eight ball compositions in urns (corresponding to information disclosure policies) such that, conditional on the realization of a red ball draw, the Bayesian posterior would be $(0.2, 0.3, \ldots, 0.9)$ for both treatments. The detailed setup of our ball composition is included in Table 4.2. For each arriving worker, she is randomly assigned to one of the treatments and needs to answer 20 questions. Each question corresponds to a ball composition. Each ball composition is repeated 2 to 3 times and the order of the question and the options are all randomized to alleviate any potential position bias.

This treatment design enables us to answer both research questions Q1 and Q2. Since we control the ball compositions so that both treatments lead to the same set of Bayesian posteriors (conditional on red ball draw), by comparing the worker behavior between the two treatments, we can answer Q1. Since the prior is fixed in each treatment, by examining the behavior with different induced posterior in the same treatment, we can answer Q2.

ball composition	prior $(0.2, 0.8)$	prior $(0.4, 0.6)$
posterior $(0.2, 0.8)$	$(100\%, 0\%, 100\%, 0\%)$	$(37\%, 63\%, 100\%, 0\%)$
posterior $(0.3, 0.7)$	$(100\%, 0\%, 58\%, 42\%)$	$(64\%, 36\%, 100\%, 0\%)$
posterior $(0.4, 0.6)$	$(100\%, 0\%, 37\%, 63\%)$	$(100\%, 0\%, 100\%, 0\%)$
posterior $(0.5, 0.5)$	$(100\%, 0\%, 25\%, 75\%)$	$(100\%, 0\%, 67\%, 33\%)$
posterior $(0.6, 0.4)$	$(100\%, 0\%, 17\%, 83\%)$	$(100\%, 0\%, 44\%, 56\%)$
posterior $(0.7, 0.3)$	$(100\%, 0\%, 11\%, 89\%)$	$(100\%, 0\%, 29\%, 71\%)$
posterior $(0.8, 0.2)$	$(100\%, 0\%, 6\%, 94\%)$	$(100\%, 0\%, 17\%, 83\%)$
posterior $(0.9, 0.1)$	$(100\%, 0\%, 3\%, 97\%)$	$(100\%, 0\%, 7\%, 93\%)$

Table 4.2: Ball compositions for different prior and different posterior on seeing red ball. In each cell, the first two numbers correspond to the fraction of red balls and blue balls in Urn X, and the last two numbers correspond to the fraction of red balls and blue balls in Urn Y.

4.5.2 Experiment Results

Among the 400 recruited workers, 199 workers were randomly assigned to the high prior $(0.4, 0.6)$ treatment and 201 workers were randomly assigned to the low prior $(0.2, 0.8)$ treatment. For the self-reported population demographic for the participants, there are 41.5% female, 71.25% under 40 years old, and over 90% of the participants reported to have at least college degrees.

Receiver's behavior. We first report the receiver's behavior on both treatments. Note that if workers are Bayesian rational, we should expect to see workers taking the same actions for any fixed posterior no matter which treatment they are in. In addition, workers' behavior should follows a step function within each treatment, with workers choosing urn X when the posterior is larger than 0.5 and choosing urn Y otherwise.

The results, as shown in Figure 4.3, show that worker behavior has significantly deviated from the model of Bayesian rationality. In particular, the differences between the two treatments

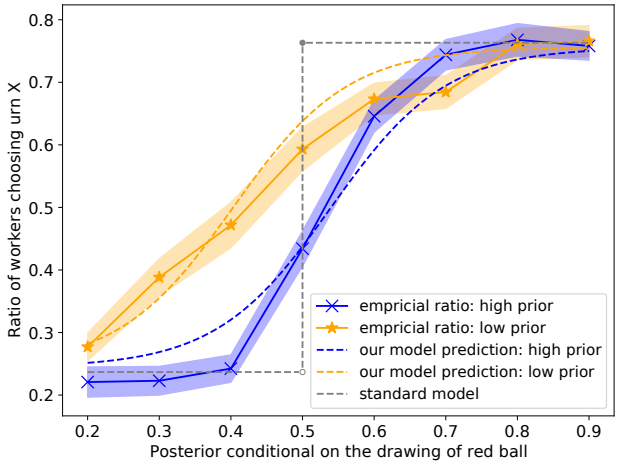
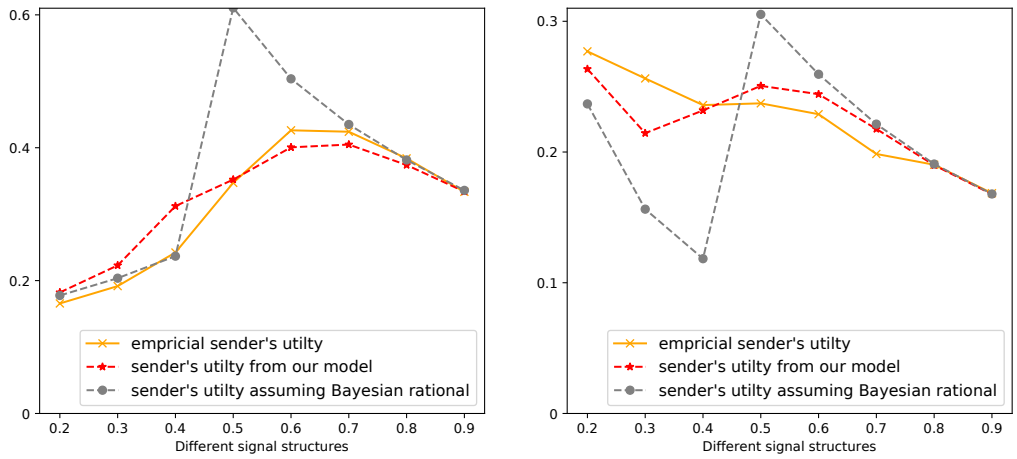


Figure 4.3: The solid lines represent the percentage of workers that choose Urn X conditional on a red ball realization. Shaded regions correspond to the regions of plus/minus one standard error. Dashed lines correspond to fitted models in our framework.

demonstrate that workers are not updating their beliefs in a Bayesian manner. The sigmoid-shape curve in workers' behavior demonstrates that worker behavior aligns better with the discrete choice model instead of the expected utility theory (which leads to a step function).

Fitting receiver behavior to our framework. Next we examine how well our framework explains the empirical worker behavior by fitting the empirical observations to our model as described in Equation (4.5). For the probability weighting function ω, we choose a simple but an intuitive affine probability weighting function. In addition, since the data quality by online workers have known to be inconsistent [87, 89], when fitting the data to models, we consider the case that there is a $(1-\alpha)$ fraction of workers who might always be random guessing (choosing urn X with 0.5 chance).

The fitted curves are also included in Figure 4.3. Compared with the step function as predicted with the Bayesian rationality assumption, our model aligns better with real-world human behavior.



(a) High prior. (b) Low prior.

Figure 4.4: Comparisons between the empirical sender's utility collected in data, sender's utility predicted by our model, and the sender's utility predicted by assuming workers are Bayesian rational.

Details on the model evaluation. Recall that our model, after including an affine probability distorting function $\omega(\cdot|\mu_0)$ and α-fraction random workers, is defined as follows

$$\begin{aligned} &p(a = a_{\mathsf{X}}|\omega(\mu|\mu_0)) \\ =&\frac{\alpha \exp(\beta\omega(\mu|\mu_0))}{\exp(\beta(1-\omega(\mu|\mu_0))) + \exp(\beta\omega(\mu|\mu_0))} + (1-\alpha)0.5, \end{aligned}$$

where $\omega(\mu|\mu_0) = \gamma\mu^* + (1-\gamma)\mu$, $\alpha, \gamma \in [0,1], \beta > 0$ and μ^* is a reference belief that may depend on the prior information. Using non-linear least squares, we jointly optimize the parameters of function $p(a = a_{\mathsf{X}}|\omega(\mu|\mu_0))$, while ensuring parameters (α, β, γ) to be the same for both treatments and allowing $\mu^* \in [0,1]$ to vary with the prior, to be fitted to the data of both treatments. To assure for fair comparisons, we also include the prediction if we assume α fraction of workers are Bayesian rational (see gray dashed line in Figure 4.3). Recall that in Proposition 4.4.1, the response of Bayesian rational workers is a step function $p(a = a_{\mathsf{X}}|\mu) = \mathbf{1}\{\mu \geq 0.5\}$. Thus, with $(1-\alpha)$ fraction random workers, the prediction should be characterized by $p(a = a_{\mathsf{X}}|\mu) = \alpha\mathbf{1}\{\mu \geq 0.5\} + (1-\alpha)0.5$.

To evaluate how well each model fits the data, we use 5-fold cross-validation to estimate the out-of-sample prediction error of the model. In particular, we split the available data randomly into 5 equally-sized disjoint subsets. In each iteration, we choose one subset as the test data and the remaining subsets as training data to find out the model parameters. The out-of-sample performance is then evaluated on the chosen test data. After iterating all subsets, we compute the average out-of-sample error across 5 test sets.

The evaluation errors, computed via the sum of squared residuals, together with the errors if we assume workers are Bayesian rational, are shown in Table 4.3. The results demonstrate that our framework explains the real human behavior better than Bayesian persuasion does.

	error using our model	error assuming Bayesian rational
prior $(0.2, 0.8)$	0.0506	0.1230
prior $(0.4, 0.6)$	0.0417	0.1231

Table 4.3: 5-fold cross validation error (computed via the sum of squared residuals) for the models in Figure 4.3.

Implication to information design. Finally, we discuss the impacts of receiver models to the information design problem. In particular, note that each of the ball composition of urns corresponds to an information disclosure policy. For each policy, we can compute the expected utility for each receiver model by assuming the receiver takes action follows the model prediction. In addition, given the data collected by our experiments, we can compute the empirical average utility achieved by each policy (i.e., multiply the empirical ratio of workers choosing Urn X by the probability of red ball realization) and use it as the ground truth for comparison.

The results, as shown in Figure 4.4, demonstrate that our model (fitted with data) makes a much more accurate prediction (red dashed line) on the empirical average utility (orange line) of different information disclosure policies than the one predicted by Bayesian persuasion (gray dashed line). For example, for the high prior treatment, different from the peak at $(0.5, 0.5)$ when assuming workers are Bayesian rational, the empirical data shows that the empirical optimal information disclosure policy is generating a posterior between $(0.6, 0.4)$ and $(0.7, 0.3)$ when seeing a red ball, and this is also reflected in our model prediction. Similar results can also be found for the low prior treatment.

4.6 Discussions and Future Work

In this section, we discuss the limitations of our current results and potential future directions.

Generalizability of our framework and experimental results. While our work has been one of the few empirical studies in examining human behavior in the persuasion literature, similar to prior work, our experiment is constructed on a more abstract setup (i.e., utilizing the urn and ball drawing problem). Developing a more realistic experimental setup that depicts real-world scenarios (e.g., how a seller selectively discloses product information to persuade the buyer to make the purchase decision) and/or conducting more extensive experiments (e.g., including more priors and posteriors, recruiting more workers) would help better understand and model real human behavior.

In addition, our current experiments has limited to a simple form of information presentation. It is therefore not trivial to claim that our findings hold for different presentations of information structure. In particular, in our experiment design, for each combination of prior and target posterior, we identify an information disclosure policy (i.e., a particular set of ball compositions in urns) that induces the target posterior from the prior when receiver

sees a red ball. In our design, almost all ball compositions have 100% red balls in Urn X except two compositions in the upper right of Table 4.2. There are several benefits for this style of composition. First, it aligns with the optimal information design as derived in Proposition 4.4.1, i.e., the sender always sends a signal R when Urn X is realized. For the two compositions that it is not feasible to have 100% red balls in Urn X, we choose to make Urn Y to contain 100% red balls to ensure that our ball compositions are consistent among different tasks, i.e., at least one urn has 100% red balls. Second, we believe the simplicity of these compositions also helps to alleviate human's cognitive burden when processing signal information. However, despite the above mentioned benefits, it limits the generalizability of our findings outside of this particular form of information presentation. Note that there are essentially infinite number of different ball compositions that we can use to induce the same target posterior. For example, in Figure 4.2, given the prior $(0.4, 0.6)$, any ball composition $(x, 1-x, y, 1-y)$ that satisfies $\frac{0.4x}{0.4x+0.6y} = 0.6, x, y \in [0, 1]$ can induce a posterior $(0.6, 0.4)$ whenever a worker sees a red ball. Understanding the impacts of different signal presentations has practical importance and would be an important future research direction.

We have considered a particular set of behavioral models, i.e., discrete choice model and probability weighting, to relax the Bayesian rational assumption. While these models have been well-examined in the literature, there have also been other models of human decision making to relax the assumption of Bayesian rationality. Empirically understanding whether and when other models are suitable and how different models impact the information design problem requires more future studies from both theoretical and experimental investigations.

Algorithmic solutions for information design. In Section 4.2, we develop an alternative framework to model the receiver's behavior and formulate the sender's optimization problem. In the section about our baseline setting, we then demonstrate that in a simple baseline setup with two states and binary actions, we can obtain a closed form of optimal information

structure. The natural next question to ask is that whether we can develop an algorithmic procedure to obtain the optimal information design for general settings in these frameworks.

Note that if the receiver is Bayesian rational, for a general information design problem, there have been earlier works [51] showing that it is #P-hard to exactly compute the expected sender utility for the optimal information structure. One interesting future direction is to explore whether the earlier computational complexity results still hold in our framework. More specifically, can we identify a polynomial-time algorithm to derive the optimal information disclosure policy, as defined in (4.6).

Potential negative societal impacts. Lastly, we would like to highlight the potential negative societal impacts of the usage of information design. When the sender's objective is to maximize the social welfare or to improve the quality of the receiver's action, the impacts of information design could be positive to the receiver and beneficial to the society. However, in our work and in almost the entire literature on Bayesian persuasion, we have often focused on how to identify an optimal information disclosure policy that maximizes the sender's payoff. Since the sender often represents the advantageous party (e.g., the government, the company, the platform, etc) that has access to more information, when the interests of the sender do not align with the interests of the receiver, optimizing the sender's utility could lead to potential negative social impacts to the receivers, who are often the general public. In other words, with ill-specified objective in information design, the sender could utilize the information advantage and create significant negative impacts. It is therefore also important to consider the impacts and the potential regulations on information design.

Chapter 5

Human Behavior Modeling – Learning from Peer Communication

Crowdsourcing has gained increasing popularity recently as a scalable data collection tool for various purposes, such as obtaining labeled data for training machine learning algorithms and getting high-quality yet cheap transcriptions for audio files. On a typical crowdsourcing platform like Amazon Mechanical Turk (MTurk), task requesters can post small jobs (e.g., image labeling or audio transcription tasks) as "microtasks" along with the specified payment for completing each task. Workers then can browse all available tasks on the platform and decide which ones to work on. Crowd workers are often assumed to complete tasks *independently*, and a substantial amount of crowdsourcing research has been focused on how to make better use of the independent workers. For example, a rich body of research has explored how to aggregate independent contributions from multiple workers by inferring task difficulties, worker skills, and correct answers simultaneously [32, 146, 180, 188]. Moreover, given a limited budget, researchers have further examined how to intelligently decide the number of independent workers needed for each task at the first place [30, 73, 114].

However, the validity and value of this independence assumption in crowdsourcing has been challenged recently. Through a combination of ethnographic and experimental methodologies, researchers have found that crowd workers, in fact, communicate and collaborate with each other through both online forums (like TurkerNation[14] and MTurkCrowd[15]) and one-on-one channels [69, 71, 88, 126, 153, 183]. Different from such collaboration which is organically arisen within the crowd and mostly about exchanging meta-level information related to crowd work (e.g., how to find well-paid tasks), an increasing number of studies in the human-computer interaction community have started to design certain level of interactions between workers in their actual work, which is shown to improve crowdsourcing outcomes in many cases. For example, various *workflows* are developed to coordinate workers to work on *different* subtasks and interact with each other through the pre-defined input-output handoffs [18, 31, 102, 103, 110, 115, 139, 148], which enable the crowd to jointly complete complex tasks.

More recently, worker interactions are further introduced between workers of the *same* task: [48] and [26] showed that in image/text labeling tasks, workers can improve their labeling accuracy when *indirect* interactions—in the form of showing each worker the alternative answer and associated justification produced by another worker who works on the same task—are enabled, and [155] observed that in text classification tasks, worker performance increases when they can debate their answers through *direct, real-time* deliberation with one another. While these research show the promise of an alternative way to structure crowd work that leads to higher performance, they also raise a number of open questions.

First, on the "micro" level, it is important to empirically examine whether adding interactions between workers working on the same task leads to an increase in work quality for *individual* tasks of *different* types, especially for those tasks with a large number of possible answers

[14] `http://turkernation.com/`
[15] `https://www.mturkcrowd.com/`

(rather than just a few options as in image labeling and text classification tasks). Indeed, when the number of possible answers in a task becomes large, workers may hardly agree with each other so it is unclear whether interactions between them would be meaningful and effective. It is also impractical for workers to argue against all alternative answers during their interactions, which may imply the need for new formats of interactions beyond providing justification and argumentation.

Furthermore, from a more "macro" point of view, requesters typically have a *large batch* of tasks at hand and need to solicit answers from multiple workers for each task. Their goal is to optimize their *overall* utility such as maximizing the quality obtained across *all* the tasks under a fixed budget. Yet, compared to independent work, allowing worker interactions in a task can bring up not only work quality improvement in that task, but also higher cost and higher correlation in workers' answers. Thus, for requesters to make better use of worker interactions, a critical problem to address is that given a limited budget, *whether* and *when* should worker interactions be used in each task, such that after combining the possibly correlated answers together for each task, the work quality for the entire batch of tasks is maximized when the budget is exhausted.

In this paper, we attempt to answer these two questions. In particular, inspired by the concept of peer instruction in education [40], we focus on studying a specific format of worker interactions that we refer to as *peer communication*—a *pair* of workers working on the same task are asked to first provide an independent answer each, then freely discuss the task with each other, and finally provide an updated answer, again independently, after the discussion. Compared to worker interaction formats used in the early research (e.g., justification and argumentation), we consider peer communication as a kind of direct and synchronous interaction that can be generalized to different types of tasks more easily. Our goal is to better understand not only whether and how peer communication would affect the

outcome of crowd work for various types of tasks, but also how requesters can use algorithmic approaches to better utilize the potential benefits brought up by peer communication.

To understand the effects of peer communication on crowd work, we design and conduct randomized experiments with three different types of tasks: image labeling, optical character recognition, and audio transcription. For all types of tasks in our experiments, regardless of how large the number of possible answers in the task is, we have consistently observed an increase in work quality when workers can talk with their peers in the task compared to workers who work independently. Yet, we do not observe any spillover effects of such quality improvement when workers who have engaged in peer communication work on similar tasks again independently.

Moreover, to examine how peer communication can be better utilized, we propose an algorithmic framework to help requesters make *online decisions* on whether and when to use peer communication for each task in their batch, with the goal of maximizing the overall work quality produced in all tasks given a budget constraint. One of the key challenges here is how to infer the correct answer for a task given multiple answers solicited from workers, where some of them may be produced following the peer communication procedure and thus may be correlated. To this end, we introduce the notions of *meta-workers* and *meta-labels* to describe a pair of workers who have engaged in a task with peer communication and the pair of answers produced by them. Such notions enable us to characterize the possible correlation in data, which further allow us to solve the requester's online decision-making problem by modeling it as a constrained Markov decision process.

We evaluate the effectiveness of the proposed algorithmic approach on real data collected through our experimental study. Results show that using our approach to decide the usage of peer communication in tasks, the requester can achieve higher overall quality across all

her tasks when the budget is exhausted, compared to when baseline approaches are adopted where peer communication is always used in all tasks or never used in any of the tasks, or when correlation in data is not explicitly considered. In addition, through two sets of simulated experiments, we further examine how the proposed algorithmic approach performs in various scenarios when the differences in work quality and cost between hiring pairs of communicating workers and hiring independent workers vary, and when answers produced by pairs of communicating workers are correlated to different extent.

In summary, we make the following contributions:

- We introduce peer communication, a general mechanism adapted from the concept of peer instruction in education for including worker interactions in crowd work.
- We empirically show that on different types of tasks, compared to independent work, peer communication consistently leads to a 32%–47% improvement in work quality for individual tasks.
- We propose an algorithmic framework to help requesters dynamically decide whether and when to use peer communication for each task in their batch, so as to maximize the overall quality obtained across all tasks given a budget constraint.
- Through evaluations on both real data from crowd workers and synthetic data, we demonstrate that compared to baseline approaches, using our proposed algorithm to determine the deployment of peer communication leads to higher requester utility.

5.1 Related Work

Our work joins a long line of research on improving the quality of crowd work. In traditional settings where it is assumed that workers independently complete tasks, various methods

have been proposed to address this problem, including post-hoc aggregation of workers' answers [32, 43, 44, 77, 78, 90, 146, 180, 188], designing effective incentives [76, 79, 80, 83, 84, 85, 112, 127, 152, 159, 182], designing interventions during tasks [50], appropriate assignment of tasks to workers [78, 81, 99, 100], etc.

We explore how to improve the quality of crowd work from a different angle, that is, by adding interactions between workers. Researchers have previously designed *workflows* for complex tasks to allow workers to work on different subtasks while indirectly interacting with one another through the pre-defined input-output handoffs [18, 31, 102, 103, 110, 115, 139, 148]. Different from these workflow-based approaches, we consider the addition of interactions between workers of the *same* task. A few previous and follow-up studies [26, 48, 49, 82, 155] have showed that enabling interactions between workers working on the same task, in the form of asking workers to provide justification for their answers, can lead to improvement in work quality, but these studies only test this idea on classification tasks. We aim to extend this idea to a wider range of tasks, especially for tasks with a large number of possible answers.

In this paper, we study a specific format of interactions, *peer communication*, which is adapted from "peer instruction" [40] and "think-pair-share" strategies [122] in the educational settings. There is a rich literature in the collaborative learning community suggesting that asking students to discuss conceptual questions with other students after they independently answer the questions leads to higher levels of understanding and post-test performance [35, 45, 165]. We thus design the peer communication procedure as first asking a pair of workers to provide an independent answer each, then allowing them to *freely* discuss the task with each other, and finally independently update their answers. While evidence in the collaborative learning community and results for adding argumentation in classification tasks seem to indicate peer communication would lead to higher work quality, other studies showed that allowing workers

to chat during work doesn't change work quality [185]. Thus, it is necessary to re-examine the impact of peer communication on the quality of crowd work, if any. In addition, as prior research on inter-task effects [2, 25, 138] suggests that when working on a sequence of tasks, workers' responses for later tasks could be influenced by the earlier tasks, we further examine whether peer communication brings any "spillover" effect on work quality. That is, whether workers produce higher independent work quality after engaging in similar tasks with peer communication.

Besides empirically showing the benefits of peer communication on work quality, we further provide an algorithmic framework for helping requesters better utilize such benefits. Early work has explored how indirect interactions between workers of different tasks that are embedded in certain workflows can be algorithmically controlled in order to maximize requester utility [42]. In contrast, the purpose of our algorithmic framework is to dynamically decide whether and when to deploy peer communication between workers of the *same* task for each task in requesters' batch to maximize their utility. Our framework is built on top of the work by [30], in which they used a Markov decision process to sequentially decide which task in requesters' batch needs an additional worker to work on given a budget constraint. However, our framework has a few key differences: First, in addition to choose which task needs further work, we also decide on how to design that piece of work—hiring one independent worker or two communicating workers? Second, when making inference for each task, we need to consider the possible correlation in the answers for this task. Finally, since peer communication and independent work incurs *different cost*, this decision-making problem does *not* degenerate into a finite-horizon Markov decision process. Thus, we explicitly model the problem as a *constrained* Markov decision process.

5.2 Examining Peer Communication via Real-World Experiments

In this section, we first present our experimental study, in which we carefully examine the effects of introducing peer communication between pairs of workers on the quality produced in individual tasks through a set of randomized experiments conducted on Amazon's Mechanical Turk (MTurk). In particular, we ask:

- **Question 1 (Q1)**: Do workers produce higher work quality in tasks with peer communication compared to that in tasks where workers work independently?

Previous studies on the effects of adding worker interactions in image and text classification tasks [26, 48, 155] seem to imply a positive answer for Q1. Compared to these studies, our study has two key differences that warrant a re-examination of Q1: (1) the main format of interaction in peer communication is a *synchronous, free-form chat* rather than required justification and argumentation; (2) we consider different types of tasks beyond classification, especially tasks with a large number of possible answers so workers can hardly agree with each other or argue against all alternative answers. Moreover, we are also interested in examining whether there is any "spillover" effects of the impact of peer communication on work quality. Specifically:

- **Question 2 (Q2)**: Do workers produce higher independent work quality after engaging in similar tasks with peer communication, compared to workers who always complete tasks on their own?

Both positive and negative answers might be possible for Q2: On the one hand, if communication between workers in tasks allow them to resolve misconception about the tasks or learn

useful problem-solving strategies from each other, we might expect a positive answer; on the other hand, if the benefits of peer communication are mostly due to workers being able to exchange their confidence levels on a task and eventually converge to the more confident answer [12], the answer for Q2 would likely be negative.

5.2.1 Independent Tasks vs. Discussion Tasks

In our experiments, we considered two ways to structure the tasks:

- **Independent tasks** (tasks without peer communication). In an independent task, workers are instructed to complete the task on their own.
- **Discussion tasks** (tasks with peer communication). In discussion tasks, we designed a procedure which guides workers to communicate with each other and complete the task together. Specifically, each worker is paired with another "co-worker" on a discussion task. Both workers in the pair are first asked to work on the task and submit their answers independently. Then, the pair enters a chat room, where they can see each other's independent answer. Workers are instructed to freely discuss the task with their co-workers for two minutes; for example, they can explain to each other why they believe their answers are correct. After the discussion, both workers get the opportunity to independently update and submit their final answers.

5.2.2 Experimental Treatments

The most straight-forward experimental design would include two treatments, where workers in one treatment are asked to work on a sequence of independent tasks while workers in the other treatment complete a sequence of discussion tasks. However, if we adopt such a design, the different nature of independent and discussion tasks (e.g., discussion tasks require more

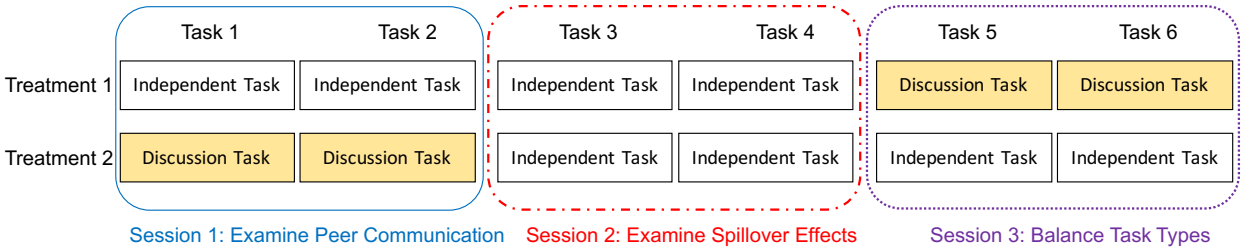


Figure 5.1: The two experimental treatments. This design enables us to examine whether peer communication improves the quality of crowd work (by comparing work quality in Session 1) and if so, does the improvement spill over to the following independent tasks (by comparing work quality in Session 2), while not creating significant differences between the two treatments (by adding Session 3 to make the two treatments containing equal number of independent and discussion tasks).

time and effort from workers but can be more interesting to workers) implies the possibility of observing severe self-selection biases in the experiments (i.e., workers may self-select into the treatment that they can complete tasks faster or find more enjoyable).

To overcome the drawback of this simple design, we design our experiments in a way that each treatment consists of the same number of independent tasks *and* discussion tasks, so neither treatment appears to be obviously more time-consuming or enjoyable. Figure 5.1 illustrates the two treatments used in our experiments. In particular, we bundle 6 tasks in each HIT (i.e., Human Intelligence Task on MTurk). When a worker accepted our HIT, she was told that there are 4 independent tasks and 2 discussion tasks in the HIT. There are two treatments in our experiments:

- *Treatment 1*: Workers are asked to complete 4 independent tasks followed by 2 discussion tasks.
- *Treatment 2*: Workers are asked to complete 2 discussion tasks followed by 4 independent tasks.

Importantly, we did *not* tell workers the ordering of the 6 tasks, which helps us to minimize the self-selection biases as the two treatments look the same to workers. We refer to the first, middle, and last two tasks in the sequence as Session 1, 2, 3 of the HIT, respectively. Thus, we can answer Q1 by comparing the work quality produced in Session 1 between the two treatments, while a comparison of work quality in Session 2 between the two treatments would allow us to answer Q2. Finally, Session 3 is used for balancing the number of independent and discussion tasks in each HIT.

5.2.3 Experimental Tasks

We conducted our experiments on three types of tasks:

- **Image labeling.** In each task, the worker is asked to identify whether the dog shown in an image is a Siberian Husky or a Malamute. Dog images we use are collected from the Stanford Dogs dataset [101].
- **Optical character recognition (OCR).** In each task, the worker is asked to transcribe a vehicle's license plate numbers from photos. The photos are taken from the dataset provided by [158].
- **Audio transcription.** In each task, the worker is asked to transcribe an audio clip which contains about 5 seconds of speech. The audio clips are collected from VoxForge[16].

We decided to conduct our experiments on these three types of tasks for two main reasons: First, these tasks are all very common types of tasks on crowdsourcing platforms [46], so experimenting with them would allow us to better understand the effects of peer communication on typical kind of crowd work. Second, in terms of the number of possible answers, these tasks span a wide spectrum from two (image labeling) to infinitely many (audio transcription),

[16] `http://www.voxforge.org`

enabling us to both confirm the effects of peer communication in tasks with just a few possible answers and explore its effects in tasks with many possible answers. As a final note, tasks we bundled in the same HIT had a certain degree of similarity[17], hence a spillover effect of peer communication on work quality is not impossible as knowledge/strategy that workers may learn in one task can potentially be transferred to another task.

5.2.4 Experimental Procedure

Enabling synchronous work among crowd workers is quite challenging, as discussed in previous research on real-time crowdsourcing [19, 22]. We address this challenge by dynamically matching pairs of workers and sending them to simultaneously start working on the same sequence of tasks. In particular, when each worker arrived at our HIT, we first checked whether there was another worker in our HIT who didn't have a co-worker yet—if yes, she would be matched to that worker and assigned to the same treatment and task sequence as that worker, and the pair then started working on their sequence of tasks together. Otherwise, the worker would be *randomly* assigned to one of the two treatments as well as a *random* sequence of tasks, and she would be asked to wait for another co-worker to join the HIT for a maximum of 3 minutes. In the case where no other workers arrived at our HIT within 3 minutes, we asked the worker to decide whether she was willing to complete all tasks in the HIT on her own (and we dropped the data for the analysis but still paid her accordingly) or get a 5-cent bonus to keep waiting for another 3 minutes.

We provided a base payment of 60 cents for all our HITs. In addition to the base payments, workers were provided with the opportunity to earn performance-based bonuses, that is, workers can earn a bonus of 10 cents in a task if the final answer they submit for that task

[17]For example, image labeling tasks are all about the key concept of distinguishing Siberian Husky from Malamute, OCR tasks have similar image quality, and audio transcription tasks contain similar accents.

is correct. Our experiment HITs were open to U.S. workers only, and each worker was only allowed to take one HIT for each type of tasks.

5.2.5 Experimental Results

In total, we have 388, 382, and 250 workers who successfully formed pairs and completed the image labeling, OCR, and audio transcription tasks in our experiments, respectively. We then answer Questions 1 and 2 separately for each type of task by analyzing experimental data from Session 1 and 2 in the HIT, respectively.[18]

Work Quality Metrics

For all three types of tasks, we evaluate the work quality using the notion of *error*. In the image labeling task, we define error as the binary classification error—the error is 0 for correct labels and 1 for incorrect labels. For OCR and audio transcription tasks, we define error as the edit distance between the worker's answer and the correct answer, divided by the number of characters in the correct answer. Naturally, for all tasks, a lower rate of error implies higher work quality.

Q1: Peer Communication Improves Work Quality

In Figure 5.2, We plot the average error rate for workers' final answers in the first two tasks (i.e., Session 1) of Treatment 1 and 2 using white and black bars, respectively. Visually, it

[18]On a side note, analyzing the data collected in Session 3 leads to conclusions that are consistent with our findings reported below, and including such data only strengthens our results. However, since we have decided not to use it in the experiment design phase, we do not include the data in the analysis. The reason of the decision is that workers' conditions in Session 3 of the two treatments differ to each other both in terms of whether they have communicated with other workers about the work in previous tasks and whether they can communicate with other workers in the current tasks, making it difficult to draw any causal conclusions on the effect of peer communication.

is clear that for all three types of tasks, the work quality is higher in discussion tasks (i.e., Session 1 of Treatment 2 HITs) when workers are able to communicate with others about the work, compared to that in independent tasks (i.e., Session 1 of Treatment 1 HITs) where workers need to complete the work on their own. Indeed, we observe a substantial 37%, 32%, and 47% deduction in the average error rate for image labeling, OCR, and audio transcription tasks when peer communication is enabled. We further conduct two-sample t-tests to check whether these changes are statistically significant, and p-values are 2.42×10^{-4}, 5.02×10^{-3}, and 1.95×10^{-11} respectively, suggesting that introducing peer communication in crowd work can significantly improve the work quality produced for various types of tasks.

We then look into the chat logs to gain some insights on how and what workers have communicated with each other during the discussion. On average, the length of the discussions in image labeling, OCR and audio transcription tasks are 4.2, 5.1 and 5.4 turns[19], yet the amount of discussion is not correlated to the quality of worker's final answers after discussion. Furthermore, by looking into the content of discussions, we find several types of information workers are exchanging during their communication:

- *Providing Justification*: e.g., "triangle ears that stand erect are traits of a Siberian Husky" (image labeling)
- *Communicating Confidence*: e.g., "I'm pretty sure about UR to start, but not very sure after that" (OCR); "I had no idea what the last word was" (audio transcription)
- *Exchanging Strategy*: e.g., "If you can zoom in on it you will see what I mean" (OCR); "He pronounces 'was' with a v-sound instead of the w-sound" (audio transcription)
- *Expressing Agreement*: e.g., "I agree"; "Listening to it again, I think you are right" (audio transcription)

[19]We count each chunk of sentences a worker entered in the chat room as a "turn."

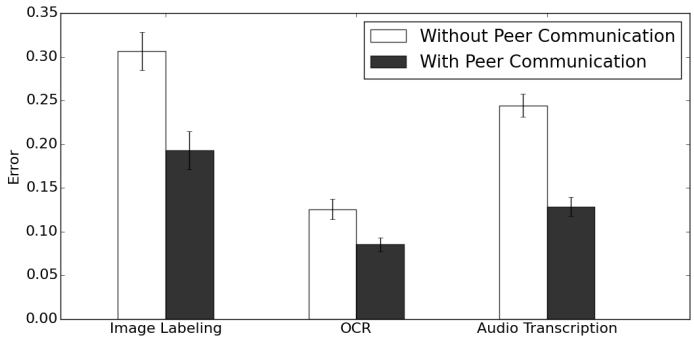


Figure 5.2: Comparisons of work quality produced in tasks with or without peer communication. Error bars indicate the mean ± one standard error.

- *Collaborative Work*: the pair of workers work together to solve the task, e.g., guessing a digit on the car plate for the OCR task that neither worker can recognize independently

Interestingly, as an anecdotal observation, we notice that in image labeling tasks the majority of workers tend to provide justifications for their answers. In OCR and audio transcription tasks, instead of "defending" their own answers, many more workers choose to team up with their co-workers to solve the task together.

Q2: There are no spillover effects

We now move on to Q2: Compared to workers who always complete tasks independently, do workers who have participated in tasks with peer communication continue to produce work of higher quality in future tasks of the same type, even if they need to complete those tasks on their own? To answer this question, we compare the work quality produced in Session 2 (i.e., the middle two independent tasks) of the two treatments for all three types of tasks. For image labeling, OCR, and audio transcription tasks, the average error rates for Session 2 in Treatment 1 (workers never engage in peer communication) are 0.324, 0.175, and 0.209 respectively, while the average error rates for Session 2 in Treatment 2 (workers have previously engaged in peer communication) are 0.334, 0.168, and 0.244. Thus, we do not

observe any spillover for the effects of peer communication on work quality, that is, the quality improvement brought up by peer communication does not carry on to future independent work.

Discussions

Results of our experimental study suggest that peer communication improves the quality of crowd work for various types of tasks, even when the number of possible answers in the task is very large, yet such effect does not spill over to later independent work. Cautions should be used when generalizing these results to substantially different contexts, such as when workers can interact with each other for an extended period of time rather than just 2 minutes, when the tasks are significantly more complex or more subjective, or when workers engage in peer communication for a longer sequence of tasks. It is, thus, an important future direction to obtain an thorough understanding on how tuning various parameters of the design space (e.g., length of interactions, complexity/subjectivity of tasks) would change the effects of peer communication.

5.3 An Algorithmic Framework for Utilizing Peer Communication

Our experimental study focuses on understanding the impact of peer communication on *individual tasks*. Now, we turn to our next question, that is, for a requester who has a *large batch* of tasks, how can he better utilize peer communication to improve the *overall utility* that he can obtain across all the tasks? In particular, we address the following research question: *Given a budget and a batch of tasks to complete, whether and when should a requester deploy peer communication in each task to maximize his total utility?*

To answer this question, there are two main challenges. First, when peer communication is deployed, workers communicate with each other before submitting their answers. Therefore, their answers might be correlated. Yet, existing aggregation methods all assume each worker complete the work independently, making it necessary for us to develop new ways to address the data correlation issue. Second, deploying peer communication incurs higher cost, since it requires us to hire workers in pairs to work for longer period of time and may need additional effort for worker synchronizations. Therefore, even though peer communication produces higher work quality for individual tasks, it is not clear deploying peer communication is always beneficial for the overall utility.

In this section, we focus on the setting in which a requester aims to collect labels for a batch of binary classification tasks with a fixed budget, and the "utility" to maximize is the average accuracy the requester obtains across all classification tasks[20]. In the following, we first discuss how to deal with the data correlation issue, that is, how to infer the correct label for a task given multiple labels solicited from workers, where some of the labels may be correlated. We then describe our algorithmic framework, a constrained Markov decision process, which adaptively decides whether and when peer communication should be deployed in each task under the budget constraint while taking into account data correlation and differing cost in deploying peer communication.

5.3.1 Dealing with Data Correlation

When peer communication is used in a task, a pair of workers directly interact with each other. Naturally, their contributions (e.g., labels in image labeling tasks) might be correlated.

[20]Our discussion can be extended to classification tasks with any finite number of labels. Extending our results to general types of tasks (such as transcription tasks) requires a well-defined utility notion that can quantify the total utility for any given set of worker contributions. It is an interesting and important future direction.

For categorical tasks with a finite number of labels, we could use the covariance notion to measure the correlation of workers' contributions. Formally, let X, Y be the random variables representing the answers generated by a pair of workers for the same task. The correlation of workers' answers can be formulated using covariance $cov(X, Y)$, defined as $cov(X, Y) = E[XY] - E[X]E[Y]$. By definition, when a pair of answers X, Y are independent, the covariance should be 0.

Measuring Data Correlation

To see whether the answers from a pair of workers are correlated when they work together on a task with peer communication, we examine workers' answers in Session 1 of both treatments in our experiments on image labeling tasks. For each of the 20 images in the experiments (with labels in $\{0, 1\}$), we calculate the covariance between pairs of labels generated in independent tasks (Session 1 of Treatment 1) and discussion tasks (Session 1 of Treatment 2)[21], in which we use the empirical average to replace the expectation in the definition. The results are shown in Figure 5.3. Perhaps not surprisingly, data collected in independent tasks is mostly independent (with covariance close to 0), while data collected in discussion tasks is correlated to various degrees.

We also calculate the covariance of workers' answers in Session 2 of both treatments to see if the data correlation caused by peer communication has any spillover effect. We find the covariance is close to 0 for both treatments, indicating the correlation in data caused by peer communication does not carry on to later independent work.

[21]Recall that in our experiment, we always send a pair of workers to work on the same sequence of tasks. Thus, an independent task is also completed by a pair of workers except that they don't communicate with each other. This allows us to directly calculate the covariance for labels generated in independent tasks.

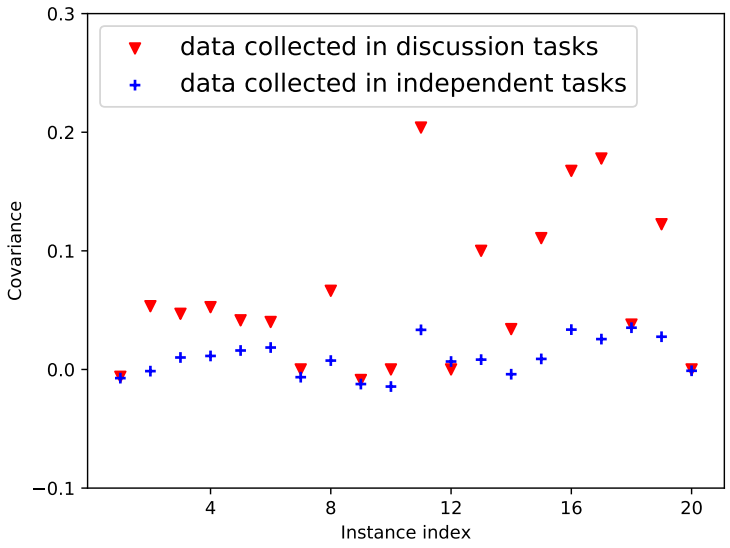


Figure 5.3: Covariance for data collected in independent tasks and discussion tasks in Session 1 in the image labeling HITs.

Meta-Workers and Meta-Labels

The above observations confirm that workers' answers are indeed correlated when peer communication is deployed. To deal with data correlation, we introduce the notions of *meta-workers* and *meta-labels*. In particular, we denote a pair of workers who talk with each other through the peer communication procedure as a meta-worker, and the pair of labels they generate as a meta-label. As no communication happens between different pairs of workers, we assume each meta-label is drawn independently.

Formally, for a binary classification task, let the true label $z \in \{0, 1\}$. When peer communication is deployed in a task, we obtain a pair of labels, which can be $\{1, 1\}$, $\{0, 1\}$, or $\{0, 0\}$, and we use the *meta-label* s_{11}, s_{01}, and s_{00} to denote them, respectively. Moreover, denote s_1 and s_0 as the label 1 and 0 obtained from a single worker who works independently.

To simplify the discussion, we assume workers are homogeneous. We propose a model to characterize the correlation in data produced in tasks with peer communication as

follows: Denote p as the probability of independent workers providing correct labels, i.e., $p = P(s_1|z = 1) = P(s_0|z = 0)$. Additionally, we denote p_+, p_0, p_- as the probability for workers in tasks with peer communication to contribute two correct labels, one correct and one incorrect label, and two incorrect labels[22]:

$$p_+ = P(s_{11}|z = 1) = P(s_{00}|z = 0)$$
$$p_- = P(s_{00}|z = 1) = P(s_{11}|z = 0)$$
$$p_0 = P(s_{01}|z = 1) = P(s_{01}|z = 0)$$

This model provides a principled way to capture different levels of correlation. For example, when the pair of labels are independent, and the probability for each worker in the pair to submit a correct label is still p, we should have $p_+ = p^2, p_- = (1-p)^2$, and $p_0 = 2p(1-p)$. When the correlation between a pair of labels is 1 (i.e., the two labels are always the same), we have $p_0 = 0$.

Utilizing Meta-labels

The idea of introducing meta-workers and meta-labels are intuitive but powerful. Below we use maximum likelihood aggregation as an example to demonstrate how the concepts of meta-workers and meta-labels can be incorporated in standard aggregation methods, which provides us with key insights on how to utilize these concepts in our algorithmic framework (we will detail this in Section 4.2).

[22]This is the extension to the standard one-coin model in crowdsourcing literatures. Extending the discussion to model the confusion matrix (e.g., using two different probability values for $P(s_{11}|z = 1)$ and $P(s_{00}|z = 0)$) is straightforward. We do not include the discussion here due to space constraints.

For a task with unknown true label $z \in \{0, 1\}$, given a set of N labels (or meta-labels) $\boldsymbol{L} = \{l_1, ..., l_N\}$, where $l_i \in \{s_{11}, s_1, s_{01}, s_0, s_{00}\}$, the maximum likelihood estimator for the value of z is defined as:

Definition 5.3.1. *Let the ground truth of the task be z. Given a set of labels $\boldsymbol{L} = \{l_1, \ldots, l_N\}$. $\hat{z}$ is a maximum likelihood estimator if*

$$\hat{z} = \begin{cases} 1 & \text{if } P(\boldsymbol{L}|z = 1) \geq P(\boldsymbol{L}|z = 0), \\ 0 & \text{otherwise.} \end{cases}$$

We assume p, p_+, p_0, and p_- are all known. Note that in our algorithmic framework as explained in Section 4.2, we adopt a Bayesian setting to learn how to aggregate the data over time without prior knowledge on values of these parameters. However, when such prior knowledge is available, a weighted majority voting rule can lead to maximum likelihood estimation:

Lemma 5.3.1. *Given a set of labels $\boldsymbol{L}$. Let $n_{11}, n_1, n_{01}, n_0, n_{00}$ denote the number of labels $s_{11}, s_1, s_{01}, s_0, s_{00}$ in $\boldsymbol{L}$. Consider the following weighted majority voting rule that generates an aggregation $\hat{z}$*

$$\hat{z} = \begin{cases} 1 & \text{if } w_{11}n_{11} + w_1 n_1 \geq w_{00}n_{00} + w_0 n_0 \\ 0 & \text{if } w_{11}n_{11} + w_1 n_1 < w_{00}n_{00} + w_0 n_0 \end{cases}$$

This weighted majority voting rule leads to maximum likelihood estimation when the weights are set as: $w_{11} = w_{00} = \ln \frac{p_+}{p_-}$, and $w_1 = w_0 = \ln \frac{p}{1-p}$.

Proof. We can write the probabilities on both sides as follows:

$$P(\boldsymbol{L}|z=1) = p_+^{n_{11}} p^{n_1} p_0^{n_{01}} (1-p)^{n_0} p_-^{n_{00}}$$
$$P(\boldsymbol{L}|z=0) = p_-^{n_{11}} (1-p)^{n_1} p_0^{n_{01}} p^{n_0} p_+^{n_{00}}$$

Therefore, we have

$$\frac{P(\boldsymbol{L}|z=1)}{P(\boldsymbol{L}|z=0)} = \left(\frac{p_+}{p_-}\right)^{n_{11}} \left(\frac{p}{1-p}\right)^{n_1} \left(\frac{1-p}{p}\right)^{n_0} \left(\frac{p_-}{p_+}\right)^{n_{00}}$$

Note that, in maximum likelihood estimator, $\hat{z}=1$ if $P(\boldsymbol{L}|z=1)/P(\boldsymbol{L}|z=0) \geq 1$. Therefore, $\hat{z}=1$ if

$$\left(\frac{p_+}{p_-}\right)^{n_{11}} \left(\frac{p}{1-p}\right)^{n_1} \geq \left(\frac{p}{1-p}\right)^{n_0} \left(\frac{p_+}{p_-}\right)^{n_{00}}$$

The proof is completed by taking logarithm on both sides. □

As a sanity check, we can see that when a pair of labels are independent (i.e., $p_+ = p^2$ and $p_- = (1-p)^2$), we have $w_{11} = w_{00} = 2w_1 = 2w_0$, implying that the weight of $\{1,1\}$ label is twice as the weight of $\{1\}$ label, and this is essentially a simple majority voting.

Note that in the maximum likelihood aggregation, the number of meta-label s_{01} does *not* play a role in the aggregation process. In other words, we may interpret the generation of meta-labels as follows: with probability p_+ (or p_-), a meta-worker generates a correct label s_{11} (or incorrect label s_{00}), while with probability p_0 she generates *no* label at all. The above weighted majority voting rule then simply indicates that different weights need to be used for labels generated by independent workers or meta-workers. Following a similar idea, in the following algorithmic framework, we only take the meta-label s_{00} and s_{11} into consideration and discard the meta-label s_{01} when inferring the correct labels of a task from a collection of labels and meta-labels.

5.3.2 Our Algorithmic Framework

With the notions of meta-workers and meta-labels in place, we have a principled way to deal with correlated data in peer communication. However, we still need to address the second challenge of balancing the quality and cost. In particular, while introducing peer communication leads to a significant improvement in work quality for individual microtasks, such improvement comes with extra cost, such as the financial payment incurred to recruit more workers (e.g., at least two workers are needed for peer communication to happen), the compensation for longer task completion time due to discussions, and the additional administrative costs for synchronizing the work pace of worker pairs. As a result, a requester needs to face the quality-cost tradeoff when deploying peer communication.

We now describe our algorithmic framework, built on the constrained Markov decision process (CMDP), that adaptively decides for a requester with a limited budget, whether and when peer communication should be deployed in each of his tasks with the goal of maximizing his total utility (i.e., the average accuracy for all classification tasks), while taking into account data correlation and differing costs for deploying peer communication.

Problem Setup.

Our problem setup is inspired by the method by [30] to optimally allocate budget among task instances in crowdsourcing data collection. Our setup differs from theirs in two fundamental ways due to the presence of peer communication strategy. First, they don't and don't need to consider the issue of data correlation. Second, in their setting, the cost for acquiring labels is fixed, while we need to deal with the differing costs when peer communication is deployed in a task. Therefore, instead of modeling the decision-making problem as a Markov decision

process framework (as in [30]), we adopt a constrained Markov decision process framework and include the meta-label concept in our formulation.

Formally, suppose a requester gets a budget of $\mathcal{B}$ and a batch of K binary classification tasks, and he needs to estimate the label for each of these tasks. The goal of the requester is to maximize the average accuracy of the estimated labels across all tasks through spending the budget to solicit labels from crowd workers and then aggregating the collected labels. We describe the setting in which workers are homogeneous (however, their performance might be different when working independently or when working with peer communication). Extensions to settings with heterogeneous workers are straightforward as described by [30].

Assume the K tasks are independent from each other, and $Z_k \in \{0, 1\}$ represents the true label for task k ($1 \leq k \leq K$). We use the notations $\theta_k \in [0, 1]$, $\alpha_s \in [0, 1]$, and $\alpha_p \in [0, 1]$ to model the label generation process, where θ_k characterizes the difficulty of task k, α_s and α_p characterize workers' performance when working independently and working with peer communication. In particular, we denote $p_{k,s,1}$ (or $p_{k,s,0}$) as the probability for a single worker (who works independently) to provide label 1 (or 0) for task k. We define $p_{k,s,1} = \alpha_s\theta_k + (1 - \alpha_s)(1 - \theta_k)$ and $p_{k,s,0} = 1 - p_{k,s,1}$. To obtain intuition for the parameters of the model, assume $\alpha_s = 1$, we can see that θ_k captures the difficulty of task k: When θ_k is close to 0.5, workers are effectively making random guess (hence the task is difficult), and when θ_k is close to 0 or 1, independent workers can consistently provide the same label (hence the task is easy). Similarly, α_s can then be interpreted as the worker skill and a larger α_s implies a higher skill. We assume θ_k is consistent with the label Z_k, which means $Z_k = 1$ if and only if $\theta_k \geq 0.5$.

Recall that we denote a meta-worker as a pair of workers in tasks with peer communication. We use α_p to denote the skill of meta-workers, and the probability for a meta-worker

to generate a meta-label s_{01} for task k is denoted as q_k. Conditioned on a meta-worker contributing a meta-label other than s_{01}, α_p is similarly defined as α_s. That is, when $p_{k,p,1}$ and $p_{k,p,0}$ is the probability for a meta-worker to generate meta-labels s_{11} and s_{00}, we have $p_{k,p,1} = (1 - q_k)(\alpha_p\theta_k + (1 - \alpha_p)(1 - \theta_k))$ and $p_{k,p,0} = 1 - p_{k,p,1} - q_k$.

After describing the data generation model, we formulate the online decision problem faced by the requester. The requester recruits workers to label his tasks in a sequential manner. Specifically, at each time step t, the requester decides on a task k_t to work on, and he can solicit label(s) from crowd workers on this task using one of the two strategies (the strategy is denoted as x_t): first, the requester can recruit a *single* worker to work on the task ($x_t = 0$), and thus obtain a label for that task; second, the requester may recruit a meta-worker (i.e., a *pair* of workers following the peer communication procedure) to work on the task ($x_t = 1$), and thus obtain a meta-label for the task. We denote c_s as the cost for recruiting a single worker and c_p ($c_p > c_s$) as the cost of recruiting a meta-worker through peer communication strategy.

Naturally, the requester's activity in each time step can be summarized through the tuple (k_t, x_t). We also denote y_t as the label (or meta-label) obtained by the requester at time t for task k_t. By the time $t_{\mathcal{B}}$ that the requester exhausts his budget, his activity history is $\mathcal{H}_{\mathcal{B}} = \{(k_0, x_0, y_0), ..., (k_{t_{\mathcal{B}}}, x_{t_{\mathcal{B}}}, y_{t_{\mathcal{B}}})\}$. The requester then aggregates the data he has collected and infers the true labels for each of the K tasks such that the expected accuracy across all K tasks, conditioned on the activity history $\mathcal{H}_{\mathcal{B}}$, is maximized.

A Constrained Markov Decision Process Formulation.

We now formally model the requester's decision-making problem as a constrained Markov decision process:

- **States**: the state s_t is a $K \times 4$ matrix, where $s_t(k, \cdot)$ is a 1×4 vector with each entry representing before time t, the number of label (or meta-labels) s_0, s_1, s_{00}, s_{11} obtained for task k. Note that following the idea that we have discussed in Section 4.1.3, we consider the meta-label s_{01} to contributes zero utility to the requester and thus we do not include the count of it in the state.
- **Actions**: $a_t = (k_t, x_t)$, where k_t is the task to work on at time t, and $x_t \in \{0, 1\}$ represents the worker recruiting strategy, with 0 being recruiting a single worker working independently and 1 being recruiting a pair of workers to follow the peer communication procedure.
- **Transition probabilities**: When $a_t = (k_t, x_t = 0)$,

$$Pr(s_{t+1}|s_t, a_t) = \begin{cases} p_{k_t,s,1} & \text{if } s_{t+1} = s_t + (\mathbf{0}, \mathbf{e_{k_t}}, \mathbf{0}, \mathbf{0}) \\ p_{k_t,s,0} & \text{if } s_{t+1} = s_t + (\mathbf{e_{k_t}}, \mathbf{0}, \mathbf{0}, \mathbf{0}) \\ 0 & \text{otherwise} \end{cases}$$

where $\mathbf{e_{k_t}}$ is a $K \times 1$ vector with value 1 at the k_t-th entry and 0 at all other entries. On the other hand, when $a_t = (k_t, x_t = 1)$,

$$Pr(s_{t+1}|s_t, a_t) = \begin{cases} p_{k_t,p,1} & \text{if } s_{t+1} = s_t + (\mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{e_{k_t}}) \\ 1 - p_{k_t,p,1} - q_{k_t} & \text{if } s_{t+1} = s_t + (\mathbf{0}, \mathbf{0}, \mathbf{e_{k_t}}, \mathbf{0}) \\ q_{k_t} & s_{t+1} = s_t \\ 0 & \text{otherwise} \end{cases}$$

- **Rewards**: We adopt the same reward function as that used by [30]. Specifically, we assume the parameters $\theta_k, \alpha_s, \alpha_p$ are sampled from three separate Beta prior distributions, and we update the posteriors of these distributions through variational approximation where hyper-parameters are decided by moment matching. Doing so, we can then define the

reward as $R(s_t, a_t) = \mathbb{E}(h(P_{k_t}^{t+1}) - h(P_{k_t}^t))$, where P_k^t is the probability of the parameter θ_k taking on a value of at least 0.5 given the posterior of θ_k at time t, $h(x) = \max(x, 1-x)$, and the expectation is taken over all possible label y_t observed after action a_t.

- **Constraint**: Different from the setting in the work by [30], as different actions imply different costs, we need to explicitly characterize the budget constraint for our problem. Formally, the requester needs to ensure the budget constraint is satisfied. $\sum_{t=0}^{t_\mathcal{B}} c_s \mathbf{1}(x_t = 0) + c_p \mathbf{1}(x_t = 1) \leq \mathcal{B}$, where $\mathbf{1}(.)$ is the indicator function.

Proposed Algorithm

We adopt the method of Lagrangian multipliers to solve the above constrained optimization problem, which converts the problem of maximizing the total reward (i.e., $\sum_{t=0}^{t_\mathcal{B}} R(s_t, a_t)$) under the budget constraint into a simpler problem of maximizing the auxiliary function $\sum_{t=0}^{t_\mathcal{B}} R(s_t, a_t) - \lambda \sum_{t=0}^{t_\mathcal{B}} (c_s \mathbf{1}(x_t = 0) + c_p \mathbf{1}(x_t = 1))$. Notice this optimization problem is equivalent to solve a (unconstrained) Markov decision process where reward in each step is redefined as $R^{'}(s_t, a_t) = R(s_t, a_t) - \lambda(c_s \mathbf{1}(x_t = 0) + c_p \mathbf{1}(x_t = 1))$. We use the optimistic knowledge gradient technique introduced by [30] to solve the optimal policy of this MDP, which produces a single-step look-ahead policy that maximizes the highest reward at each step. Note that in theory, we can compute the optimal value of λ by solving the dual of the constrained MDP. In practice, we have experimented with multiple different λ values and find that the choice of λ has limited influence on the performance of our algorithmic approach.

5.3.3 Evaluations

We evaluate the effectiveness of our algorithmic approach using both real-world data and synthetic data.

Experiments on Real Data

Using the real data that we collected in image labeling tasks of our experimental study, we compare the performance of our algorithm with a couple of baseline algorithms. In our evaluation, we set the cost of recruiting a single worker as $c_s = 1.0$ and the cost of recruiting a pair of workers to work on a task with peer communication (i.e., a meta-worker) $c_p = 2.5$. Note that we have examined a range of different values of c_p from 1.5 to 3.5 and the results are qualitatively similar. The prior distribution for θ_k is set as Beta$(1, 1)$, where the prior distributions for α_s and α_p are all set to be Beta$(4, 1)$. For this evaluation, we only considered the final labels that workers in our experimental study submit in the *first two tasks* of the image labeling HIT[23]. Thus, when $a_t = (k_t, 0)$, we randomly sampled a label from Treatment 1 workers who had completed task k_t in their first two (independent) tasks, and when $a_t = (k_t, 1)$, we randomly sampled a label from Treatment 2 workers who had completed task k_t in their first two (discussion) tasks.

The performance of our algorithmic approach is compared against the following baseline approaches:

- *Round robin*: in each round, the requester decides which task to work on in a round robin fashion, and he always recruit a single worker to work on that task independently.
- *Single workers only*: in each round, the requester recruits a single worker to work on a task independently, and this task is optimally decided (effectively by considering only actions with $x_t = 0$ in our algorithm).
- *Peer communication only*: in each round, the requester recruits a pair of workers to work on a task with peer communication, and this task is optimally decided (effectively by considering only actions with $x_t = 1$ in our algorithm).

[23]Recall that we set out to examine the effects of peer communication using the first two tasks in each HIT.

- *Our algorithm [No correlation]*: in each round, the requester uses our algorithm to decide whether to deploy peer communication and which task to work. The only difference is that this baseline treats the two labels from peer communication as two independent labels while our algorithm incorporates the concept of meta-labels to deal with data correlation.

We conduct this evaluation on a range of budget level from 20 to 400 with an interval of 20. At each budget level, we implement each of the decision-making strategies for 100 times, and we report the average level of overall accuracy the requester obtains across the 20 tasks when she exhausts the budget in Figure 5.4.

As shown in Figure 5.4, our proposed algorithm outperforms all baseline strategies. In particular, we make a few observations as follows. First, comparing the performance of our algorithm and that of the "No Correlation" strategy, it is clear that incorporating meta-labels to deal with data correlation has improved the requester's overall utility. In fact, even for the "peer communication only" strategy, we also implement two versions, and the version for which the idea of meta-labels is used also outperforms the other version treating two labels generated by pairs of workers as independent. In the following discussion, unless otherwise specified, we adopt the meta-worker ideas in our implementation when peer communication is used. Second, strategies involving peer communication converge to a better overall accuracy than strategies without peer communication does. This is due to the fact that for some tasks in our experiment, the majority of workers who work independently provide incorrect answers while the majority of workers with peer communication provide correct answers. Third, adaptively determining whether and when to deploy peer communication outperforms fixed recruiting strategies, as illustrated by the superior performance of our algorithm over both the "single workers only" and "peer communication only" strategies. Finally, adaptively deciding which task to label next significantly improves the total utility than random task

assignment does, e.g., through observing the significantly worse performance of the baseline "round robin" strategy.

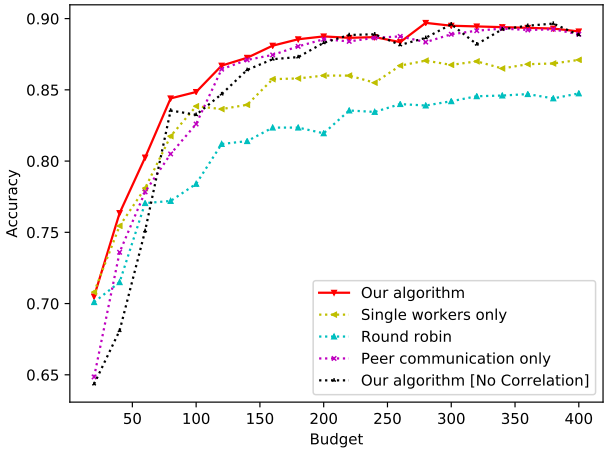
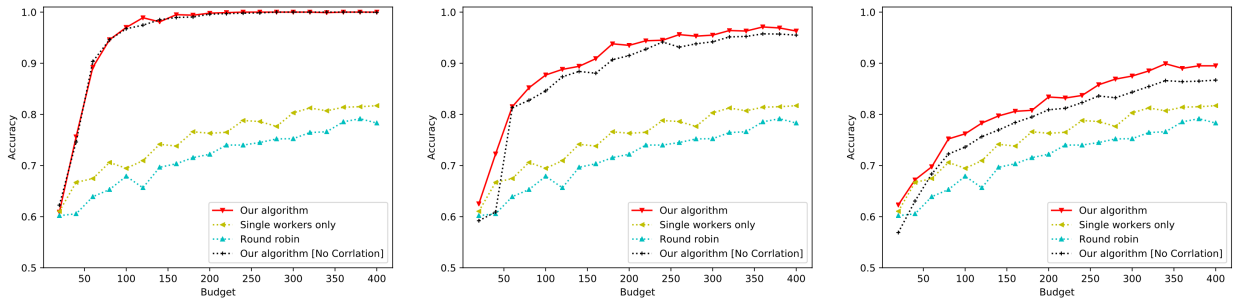


Figure 5.4: Evaluating the performance of the proposed approach on real datasets.

Experiments on Synthetic Data

To the best of our knowledge, our dataset is the only dataset that deploys peer communication for crowdsourcing data collection. Therefore, to further investigate the properties of our proposed algorithm, we generate synthetic data to evaluate our algorithm. In particular, we explore how the performance of our algorithm changes along the following three dimensions: 1) the level of data correlation of workers' answers in tasks with peer communication, 2) the performance gap between workers who work independently and workers who discuss with others via peer communication, and 3) the cost differences of hiring a single worker and hiring a pair of workers for peer communication. In the base setup, we set θ_k to be uniformly drawn from $[0.5, 1]$, α_s drawn from a normal distribution with mean 0.7 and variance 0.01, α_p drawn from a normal distribution with mean 0.9 and variance 0.1. We also set $c_s = 1$ and $c_p = 2.5$.



(a) No Correlation (b) Weak Correlation (c) Strong Correlation

Figure 5.5: The performance comparison under different levels of correlation in peer communication.

We first modify the level of data correlation of workers' answers in peer communication. This can be done by changing the value of q_k, i.e., the probability of a pair of workers in peer communication to generate the meta-label s_{01}. In strong correlation, we set $q_k = 0$, which means the two workers are entirely correlated (always generating the same label). In no correlation, we set q_k to the value such that the two labels in a meta-label are independently generated (i.e., $q_k = 2\sqrt{p_{k,p,1}p_{k,p,0}}$, which can be derived using our model discussed in Section 5.3.1). In weak correlation, q_k is uniformly drawn between the above two values. We compare the performance between our algorithm and "our algorithm [no correlation]", which treats the two labels from a meta-label as independent labels. As shown in Figure 5.5, the performance gap becomes larger as the correlation becomes stronger[24]. This validates the benefits of incorporating meta-labels in our framework when there is data correlation.

We then change the cost of deploying peer communication c_p to be from $\{1.5, 2.5, 3.5\}$. As shown in Figure 5.6, our algorithm performance decreases as c_p increases. However, even when the cost of peer communication is pretty large (i.e., $c_p = 3.5$), utilizing peer communication is still beneficial. Next, we vary the performance gap between single workers (who work independently) and meta-workers by fixing the mean of α_p to be 0.9 and set the mean

[24]As a side note, the overall performance is lower in strong correlation since we fixed α_p in all three plots; two independent labels brings more information than two correlated labels. Since our goal is to measure the gap between two algorithms, we didn't tune the parameter to normalize the algorithm performance.

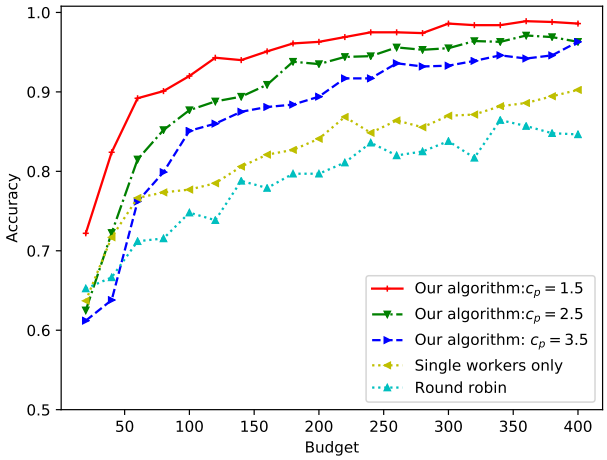


Figure 5.6: Modify the cost of peer communication.

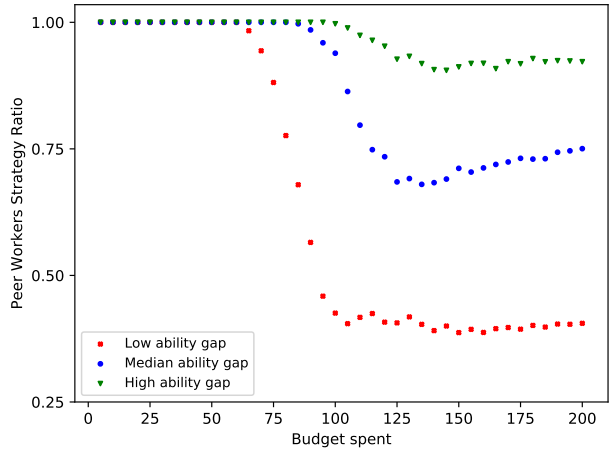


Figure 5.7: Ratio of peer communication strategies deployed.

of α_s to be 0.8, 0.7, and 0.6. The results are qualitatively similar to changing c_p (e.g., larger skill gap corresponds to smaller c_p). To provide more insights for our algorithm, we demonstrate this result in a different plot. In particular, we fix the budget to be 200 and run our algorithm 100 times. We record the worker recruiting strategy (hiring single workers or deploying peer communication) our algorithm takes at every step, and then calculate the ratio of peer communication strategy as a function of the budget spent so far. As shown in Figure 5.7, our algorithm always starts by deploying peer communication. When the marginal rewards for hiring peer communication is not high enough to justify the higher cost, our

algorithm gradually switches to hire single workers. These two figures demonstrate that our algorithm brings in benefits under a wide range of settings and has stronger benefits when c_p is small or when the performance gap between single workers and meta-workers (with peer communication) are higher.

Chapter 6

Conclusion and Future Direction

In summary, this dissertation studies human-centered machine learning from two perspectives – understanding human behavior from empirical perspective and designing efficient and socially responsible algorithms when humans are involved in from theoretical perspective. While this dissertation provides several solutions to some specific problems, it has only scratched the surface of this new emerging area. We conclude this dissertation by outlining several future research directions to better understand human-centered machine learning.

Behavioral Experiments as A Lens of Understanding Human Behavior. Theoretical analyses in algorithmic-based systems often assume stylized/simple human behavior models. For example, in strategic classification, humans are assumed to be rational and aim to maximize their payoff. However, these models often fail to explain human's behavior in many real-world scenarios. Future work includes conducting more extensive behavioral experiments to understand and model human behavior in a wide range of contexts. Examining the impact of such behavior models in various social contexts of algorithm design is also a natural next

step. The goal is to provide insights on how to better model and incorporate real-world human behavior in algorithm design.

Learning, Fairness and Privacy with Realistic Human Behavior. What are the societal impacts when deploying machine learning algorithms with humans in the loop? It is well-known that learning with human-generated data often suffer the issues of fairness or privacy, especially when the data are representing people's socioeconomic status or are user-specific. On the other hand, human-generated data closely relates to how human behave in the process. To explore how human behavior affect the design of fair or private learning algorithms, it is important to incorporate realistic human behavior in the design of learning algorithms and investigate the impacts of learning algorithms to humans. To this end, it is important to develop human models that capture the most salient behavior aspects of humans and then incorporate these human models in *fair or private* learning algorithm design.

Design Information in Human-Centered Machine Learning. AI-assisted decision making can be viewed as a process of an AI system (the sender) providing information for humans (the receiver) to make final decisions. One of our recent works [135] demonstrate that the predictive information from AI could impact human ethical preferences. Different ethical preferences also leads to different decisions or responses when human see information from AI. In this line of future research, with anticipated human response, one interesting questions is what information structure is desired for an AI system to include in interaction with its users in order to promote desired behaviors and outcomes.

In [55], we give efficient algorithms on how to learn an optimal information policy if the AI has no knowledge about human's preferences and humans are responding to the information in a Bayesian rational manner. However, in practice, humans may not be Bayesian rational, it is thus important to consider more realistic human behavior models, especially the one

pertinent to capture human's cognitive strengths/limitations on processing information. With these models at hand, one natural question is whether we can develop an efficient algorithm to obtain the optimal information policy.

Perhaps a theoretical model can have impact only when it is deployed in the real-world. However, there are many challenge in realizing such deployment. One particular challenge is on characterizing *desired* information structure whenever humans are involved in. Since the sender (i.e., AI) often represents the advantageous party (e.g., the government, the company, the platform, etc) that has access to more information, when the interests of the sender do not align with the interests of the receiver, optimizing the sender's utility could lead to potential negative social impacts to the receivers, who are often the general public. In other words, with ill-specified objective in information design, the sender could utilize the information advantage and create significant negative impacts. This question is of potential importance as we are moving into the era of AI-assisted decision making, where human utilize the information provided by AI algorithms to make decisions. It is therefore also important to consider the societal impacts of and the potential regulations on information design.

References

[1] Jacob Abernethy, Yiling Chen, Chien-Ju Ho, and Bo Waggoner. "Low-cost learning via active data procurement." In *Proceedings of the Sixteenth ACM Conference on Economics and Computation*. 2015, pp. 619–636.

[2] Alan Aipe and Ujwal Gadiraju. "SimilarHITs: Revealing the Role of Task Similarity in Microtask Crowdsourcing." In *Proceedings of the 29th on Hypertext and Social Media*. 2018.

[3] Zeyuan Allen-Zhu and Elad Hazan. "Variance reduction for faster non-convex optimization." In *International Conference on Machine Learning (ICML)*. 2016, pp. 699–707.

[4] Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. "Machine bias." In *ProPublica, May* 23 (2016), p. 2016.

[5] Andreas Aristidou, Giorgio Coricelli, and Alexander Vostroknutov. "Incentives or Persuasion? An Experimental Investigation." In *GSBE Research Memoranda* 012 (2019).

[6] Pak Hung Au and King King Li. "Bayesian persuasion and reciprocity: theory and experiment." In *Available at SSRN 3191203* (2018).

[7] Jean-Yves Audibert and Sébastien Bubeck. "Minimax policies for adversarial and stochastic bandits." In *Conference on Learning Theory*. 2009.

[8] Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. "Finite-time analysis of the multiarmed bandit problem." In *Machine learning* 47.2-3 (2002), pp. 235–256.

[9] Peter Auer, Nicolo Cesa-Bianchi, Yoav Freund, and Robert E Schapire. "Gambling in a rigged casino: The adversarial multi-armed bandit problem." In *Foundations of Computer Science, 1995. Proceedings., 36th Annual Symposium on*. IEEE. 1995, pp. 322–331.

[10] Peter Auer, Nicolo Cesa-Bianchi, Yoav Freund, and Robert E Schapire. "The non-stochastic multiarmed bandit problem." In *SIAM journal on computing* 32.1 (2002), pp. 48–77.

[11] Kay W Axhausen and Tommy Gärling. "Activity-based approaches to travel analysis: conceptual frameworks, models, and research problems." In *Transport reviews* 12.4 (1992), pp. 323–341.

[12] Bahador Bahrami, Karsten Olsen, Peter E Latham, Andreas Roepstorff, Geraint Rees, and Chris D Frith. "Optimally interacting minds." In *Science* 329.5995 (2010), pp. 1081–1085.

[13] Abhijit V Banerjee. "A simple model of herd behavior." In *The quarterly journal of economics* 107.3 (1992), pp. 797–817.

[14] Alexander Bartik and Scott Nelson. "Credit reports as resumes: The incidence of pre-employment credit screening." In (2016).

[15] Robert Bartlett, Adair Morse, Richard Stanton, and Nancy Wallace. "Consumer-Lending Discrimination in the Era of FinTech." In *Unpublished working paper. University of California, Berkeley* (2018).

[16] Yahav Bechavod, Katrina Ligett, Aaron Roth, Bo Waggoner, and Steven Z Wu. "Equal opportunity in online classification with partial feedback." In *Advances in Neural Information Processing Systems*. 2019, pp. 8972–8982.

[17] Dirk Bergemann and Stephen Morris. "Information design: A unified perspective." In *Journal of Economic Literature* 57.1 (2019), pp. 44–95.

[18] Michael S Bernstein, Greg Little, Robert C Miller, Björn Hartmann, Mark S Ackerman, David R Karger, David Crowell, and Katrina Panovich. "Soylent: a word processor with a crowd inside." In *Proceedings of the 23nd annual ACM symposium on User interface software and technology*. ACM. 2010, pp. 313–322.

[19] Michael S. Bernstein, Joel Brandt, Robert C. Miller, and David R. Karger. "Crowds in Two Seconds: Enabling Realtime Crowd-powered Interfaces." In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology (UIST)*. 2011.

[20] Omar Besbes, Yonatan Gur, and Assaf Zeevi. "Non-stationary stochastic optimization." In *Operations research* 63.5 (2015), pp. 1227–1244.

[21] Omar Besbes, Yonatan Gur, and Assaf Zeevi. "Stochastic multi-armed-bandit problem with non-stationary rewards." In *Advances in Neural Information Processing Systems (NIPS)*. 2014, pp. 199–207.

[22] Jeffrey P. Bigham et al. "VizWiz: Nearly Real-time Answers to Visual Questions." In *Proceedings of the 23Nd Annual ACM Symposium on User Interface Software and Technology (UIST)*. 2010.

[23] Sushil Bikhchandani, David Hirshleifer, and Ivo Welch. "A theory of fads, fashion, custom, and cultural change as informational cascades." In *Journal of political Economy* 100.5 (1992), pp. 992–1026.

[24] Sébastien Bubeck, Rémi Munos, and Gilles Stoltz. "Pure exploration in finitely-armed and continuous-armed bandits." In *Theoretical Computer Science* 412.19 (2011), pp. 1832–1852.

[25] Carrie J. Cai, Shamsi T. Iqbal, and Jaime Teevan. "Chain Reactions: The Impact of Order on Microtask Chains." In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. 2016.

[26] Joseph Chee Chang, Saleema Amershi, and Ece Kamar. "Revolt: Collaborative Crowdsourcing for Labeling Machine Learning Datasets." In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI)*. 2017.

[27] Nick Chater, Joshua B Tenenbaum, and Alan Yuille. "Probabilistic models of cognition: Conceptual foundations." In *Trends in cognitive sciences* 10.7 (2006), pp. 287–291.

[28] Wei Chen, Wei Hu, Fu Li, Jian Li, Yu Liu, and Pinyan Lu. "Combinatorial multi-armed bandit with general reward functions." In *Advances in Neural Information Processing Systems*. 2016, pp. 1659–1667.

[29] Wei Chen, Yajun Wang, Yang Yuan, and Qinshi Wang. "Combinatorial multi-armed bandit and its extension to probabilistically triggered arms." In *The Journal of Machine Learning Research* 17.1 (2016), pp. 1746–1778.

[30] Xi Chen, Qihang Lin, and Dengyong Zhou. "Optimistic knowledge gradient policy for optimal budget allocation in crowdsourcing." In *International Conference on Machine Learning (ICML)*. 2013.

[31] Lydia B Chilton, Greg Little, Darren Edge, Daniel S Weld, and James A Landay. "Cascade: Crowdsourcing taxonomy creation." In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM. 2013, pp. 1999–2008.

[32] Sharath R. Cholleti, Sally A. Goldman, Avrim Blum, David G. Politte, and Steven Don. "Veritas: Combining expert opinions without labeled data." In *Proceedings 20th IEEE international Conference on Tools with Artificial intelligence (ICTAI)*. 2008.

[33] Alexandra Chouldechova. "Fair prediction with disparate impact: A study of bias in recidivism prediction instruments." In *Big data* 5.2 (2017), pp. 153–163.

[34] Geoffroy de Clippel and Xu Zhang. *Non-bayesian persuasion*. Tech. rep. Technical report, Working Paper, 2019.

[35] Derrick Coetzee, Seongtaek Lim, Armando Fox, Bjorn Hartmann, and Marti A Hearst. "Structuring interactions for large-scale synchronous peer learning." In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing*. ACM. 2015, pp. 1139–1152.

[36] Richard Combes, Alexandre Proutière, and Alexandre Fauquette. "Unimodal Bandits with Continuous Arms: Order-optimal Regret without Smoothness." In *Proceedings of the ACM on Measurement and Analysis of Computing Systems* 4.1 (2020), pp. 1–28.

[37] Bo Cowgill and Catherine E Tucker. "Economics, Fairness and Algorithmic Bias." In *preparation for: Journal of Economic Perspectives* (2019).

[38] Bo Cowgill and Eric Zitzewitz. "Incentive Effects of Equity Compensation: Employee Level Evidence from Google." In *Dartmouth Department of Economics working paper* (2009).

[39] Nick Craswell, Onno Zoeter, Michael Taylor, and Bill Ramsey. "An experimental comparison of click position-bias models." In *Proceedings of the 2008 international conference on Web Search and Data Mining (WSDM)*. ACM. 2008, pp. 87–94.

[40] Catherine Crouch and Eric Mazur. "Peer instruction: Ten years of experience and results." In *Am. J. Phys.* 69.9 (Sept. 2001), pp. 970–977.

[41] Alexander D'Amour, Hansa Srinivasan, James Atwood, Pallavi Baljekar, D Sculley, and Yoni Halpern. "Fairness is not static: deeper understanding of long term fairness via simulation studies." In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*. 2020, pp. 525–534.

[42] Peng Dai, Christopher H. Lin, Mausam, and Daniel S. Weld. "POMDP-based Control of Workflows for Crowdsourcing." In *Artif. Intell.* 202.1 (Sept. 2013), pp. 52–85.

[43] A. P. Dawid and A. M. Skene. "Maximum likeihood estimation of observer error-rates using the EM algorithm." In *Applied Statistics* 28 (1979), pp. 20–28.

[44] Arthur P. Dempster, Nan M. Laird, and Donald B. Rubin. "Maximum likelihood from incomplete data via the EM algorithm." In *Journal of the Royal Statistical Society: Series B* 39 (1977), pp. 1–38.

[45] Louis Deslauriers, Ellen Schelew, and Carl Wieman. "Improved learning in a large-enrollment physics class." In *science* 332.6031 (2011), pp. 862–864.

[46] Djellel Eddine Difallah, Michele Catasta, Gianluca Demartini, Panagiotis G Ipeirotis, and Philippe Cudré-Mauroux. "The dynamics of micro-task crowdsourcing: The case of amazon mturk." In *Proceedings of the 24th international conference on world wide web*. International World Wide Web Conferences Steering Committee. 2015, pp. 238–247.

[47] ($\alpha - \beta$). Bolin Ding, Yiding Feng, Chien-Ju Ho, **Wei Tang**, and Haifeng Xu. "Competitive Information Design for Pandora's Box." In *arXiv preprint arXiv:2103.03769* (2022).

[48] Ryan Drapeau, Lydia B. Chilton, Jonathan Bragg, and Daniel S. Weld. "MicroTalk: Using Argumentation to Improve Crowdsourcing Accuracy." In *Fourth AAAI Conference on Human Computation and Crowdsourcing (HCOMP)*. 2016.

[49] Xiaoni Duan, Chien-Ju Ho, and Ming Yin. "Does exposure to diverse perspectives mitigate biases in crowdwork? an explorative study." In *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing (HCOMP)*. 2020.

[50] Xiaoni Duan, Chien-Ju Ho, and Ming Yin. "The Influences of Task Design on Crowdsourced Judgement: A Case Study of Recidivism Risk Evaluation." In *Proceedings of the ACM Web Conference 2022*. 2022.

[51] Shaddin Dughmi and Haifeng Xu. "Algorithmic bayesian persuasion." In *SIAM Journal on Computing* 0 (2019), STOC16–68.

[52] Ward Edwards. "Conservatism in human information processing." In *Formal representation of human judgment* (1968).

[53] Hadi Elzayn, Shahin Jabbari, Christopher Jung, Michael Kearns, Seth Neel, Aaron Roth, and Zachary Schutzman. "Fair algorithms for learning in allocation problems." In *Proceedings of the Conference on Fairness, Accountability, and Transparency*. 2019, pp. 170–179.

[54] Yuval Emek, Michal Feldman, Iftah Gamzu, Renato PaesLeme, and Moshe Tennenholtz. "Signaling schemes for revenue maximization." In *ACM Transactions on Economics and Computation (TEAC)* 2.2 (2014), pp. 1–19.

[55] $(\alpha - \beta)$. Yiding Feng, **Wei Tang**, and Haifeng Xu. "Online Bayesian Recommendation with No Regret." In *arXiv preprint arXiv:2202.06135* (2022).

[56] Peter Frazier, David Kempe, Jon Kleinberg, and Robert Kleinberg. "Incentivizing Exploration." In *Proceedings of the Fifteenth ACM Conference on Economics and Computation (EC)*. 2014.

[57] Guillaume R Fréchette, Alessandro Lizzeri, and Jacopo Perego. *Rules and commitment in communication: An experimental analysis*. Tech. rep. National Bureau of Economic Research, 2019.

[58] Noufel Frikha, Stéphane Menozzi, et al. "Concentration bounds for stochastic approximations." In *Electronic Communications in Probability* 17 (2012).

[59] Andreas Fuster, Paul Goldsmith-Pinkham, Tarun Ramadorai, and Ansgar Walther. "Predictably unequal? the effects of machine learning on credit markets." In (2018).

[60] Xavier Gabaix. "Behavioral inattention." In *Handbook of Behavioral Economics: Applications and Foundations 1*. Vol. 2. Elsevier, 2019, pp. 261–343.

[61] Aurélien Garivier and Eric Moulines. "On Upper-Confidence Bound Policies for Switching Bandit Problems." In *Algorithmic Learning Theory*. Ed. by Jyrki Kivinen, Csaba Szepesvári, Esko Ukkonen, and Thomas Zeugmann. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 174–188.

[62] Rong Ge, Furong Huang, Chi Jin, and Yang Yuan. "Escaping from saddle points—online stochastic gradient for tensor decomposition." In *Conference on Learning Theory*. 2015, pp. 797–842.

[63] Scott Gehlbach and Konstantin Sonin. "Government control of the media." In *Journal of public Economics* 118 (2014), pp. 163–171.

[64] Andrew Gelman, Jeffrey Fagan, and Alex Kiss. "An analysis of the New York City police department's "stop-and-frisk" policy in the context of claims of racial bias." In *Journal of the American statistical association* 102.479 (2007), pp. 813–823.

[65] Arpita Ghosh and Patrick Hummel. "Learning and Incentives in User-generated Content: Multi-armed Bandits with Endogenous Arms." In *Proceedings of the 4th Conference on Innovations in Theoretical Computer Science (ITCS)*. 2013.

[66] Stephen Gillen, Christopher Jung, Michael Kearns, and Aaron Roth. "Online learning with an unknown fairness metric." In *Advances in Neural Information Processing Systems*. 2018, pp. 2600–2609.

[67] Sharad Goel, Justin M Rao, Ravi Shroff, et al. "Precinct or prejudice? Understanding racial disparities in New York City's stop-and-frisk policy." In *The Annals of Applied Statistics* 10.1 (2016), pp. 365–394.

[68] Itay Goldstein and Yaron Leitner. "Stress tests and information disclosure." In *Journal of Economic Theory* 177 (2018), pp. 34–69.

[69] Mary L Gray, Siddharth Suri, Syed Shoaib Ali, and Deepti Kulkarni. "The crowd is a collaborative network." In *Proceedings of the 19th ACM conference on computer-supported cooperative work & social computing*. ACM. 2016, pp. 134–147.

[70] Thomas L. Griffiths and Joshua B. Tenenbaum. "Optimal Predictions in Everyday Cognition." In *Psychological Science* 17.9 (2006), pp. 767–773.

[71] Neha Gupta, David Martin, Benjamin V Hanrahan, and Jacki O'Neill. "Turk-life in India." In *Proceedings of the 18th International Conference on Supporting Group Work*. ACM. 2014, pp. 1–11.

[72] Swati Gupta and Vijay Kamble. "Individual fairness in hindsight." In *Proceedings of the 2019 ACM Conference on Economics and Computation*. 2019, pp. 805–806.

[73] Danna Gurari and Kristen Grauman. "CrowdVerge: Predicting If People Will Agree on the Answer to a Visual Question." In *Proceedings of the 2017 Conference on Human Factors in Computing Systems (CHI)*. 2017.

[74] Hoda Heidari, Vedant Nanda, and Krishna P Gummadi. "On the Long-term Impact of Algorithmic Decision Policies: Effort Unfairness and Feature Segregation through Social Learning." In *arXiv preprint arXiv:1903.01209* (2019).

[75] Bruce M Hill, David Lane, and William Sudderth. "A strong law for some generalized urn processes." In *The Annals of Probability* (1980), pp. 214–226.

[76] Chien-Ju Ho and Kuan-Ta Chen. "On formal models for social verification." In *Proceedings of the ACM SIGKDD Workshop on Human Computation*. 2009.

[77] Chien-Ju Ho, Rafael Frongillo, and Yiling Chen. "Eliciting categorical data for optimal aggregation." In *Advances In Neural Information Processing Systems (NIPS)* (2016).

[78] Chien-Ju Ho, Shahin Jabbari, and Jennifer Wortman Vaughan. "Adaptive Task Assignment for Crowdsourced Classification." In *The 30th International Conference on Machine Learning (ICML)*. 2013.

[79] Chien-Ju Ho, Aleksandrs Slivkins, Siddharth Suri, and Jennifer Wortman Vaughan. "Incentivizing High Quality Crowdwork." In *Proceedings of the 24th International Conference on World Wide Web (WWW)*. 2015.

[80] Chien-Ju Ho, Aleksandrs Slivkins, and Jennifer Wortman Vaughan. "Adaptive Contract Design for Crowdsourcing Markets: Bandit Algorithms for Repeated Principal-Agent Problems." In *ACM Conference on Economics and Computation (EC)*. 2014.

[81] Chien-Ju Ho and Jennifer Wortman Vaughan. "Online Task Assignment in Crowdsourcing Markets." In *AAAI Conference on Artificial Intelligence (AAAI)*. 2012.

[82] Chien-Ju Ho and Ming Yin. "Working in Pairs: Understanding the Effects of Worker Interactions in Crowdwork." In *arXiv preprint arXiv:1810.09634* (2018).

[83] Chien-Ju Ho, Yu Zhang, Jennifer Wortman Vaughan, and Mihaela Van Der Schaar. "Towards social norm design for crowdsourcing markets." In *Human Computation Workshop (HCOMP)*. 2012.

[84] John Joseph Horton and Lydia B. Chilton. "The labor economics of paid crowdsourcing." In *Proceedings of the 11th ACM conference on Electronic commerce (EC)*. 2010.

[85] Jane Yung-jen Hsu, Kwei-Jay Lin, Tsung-Hsiang Chang, Chien-ju Ho, Han-Shen Huang, and Wan-rong Jih. "Parameter learning of personalized trust models in broker-based distributed trust management." In *Information Systems Frontiers* (2006), pp. 321–333.

[86] Lily Hu and Yiling Chen. "A short-term intervention for long-term fairness in the labor market." In *Proceedings of the 2018 World Wide Web Conference*. 2018, pp. 1389–1398.

[87] Panagiotis G Ipeirotis, Foster Provost, and Jing Wang. "Quality management on amazon mechanical turk." In *Proceedings of the ACM SIGKDD workshop on human computation*. 2010, pp. 64–67.

[88] Lilly C Irani and M Silberman. "Turkopticon: Interrupting worker invisibility in amazon mechanical turk." In *Proceedings of the SIGCHI conference on human factors in computing systems*. ACM. 2013, pp. 611–620.

[89] Srikanth Jagabathula, Lakshminarayanan Subramanian, and Ashwin Venkataraman. "Identifying unreliable and adversarial workers in crowdsourced labeling tasks." In *The Journal of Machine Learning Research* 18.1 (2017), pp. 3233–3299.

[90] Rong Jin and Zoubin Ghahramani. "Learning with multiple labels." In *Advances in Neural Information Processing Systems (NIPS)*. 2003.

[91] Thorsten Joachims, Laura Granka, Bing Pan, Helene Hembrooke, and Geri Gay. "Accurately interpreting clickthrough data as implicit feedback." In *ACM SIGIR Forum*. Vol. 51. 1. ACM. 2017, pp. 4–11.

[92] Matthew Joseph, Michael Kearns, Jamie H Morgenstern, and Aaron Roth. "Fairness in learning: Classic and contextual bandits." In *Advances in Neural Information Processing Systems*. 2016, pp. 325–333.

[93] D. Kahneman and A. Tversky. "On the psychology of prediction." In *Psychological Review* 80 (1973), pp. 237–251.

[94] Daniel Kahneman and Amos Tversky. "Prospect theory: An analysis of decisions under risk." In *Econometrica* (1979), pp. 263–291.

[95] Emir Kamenica. "Bayesian persuasion and information design." In *Annual Review of Economics* 11 (2019), pp. 249–272.

[96] Emir Kamenica and Matthew Gentzkow. "Bayesian persuasion." In *American Economic Review* 101.6 (2011), pp. 2590–2615.

[97] Emir Kamenica and Matthew Gentzkow. "Bayesian persuasion." In *American Economic Review* 101.6 (2011), pp. 2590–2615.

[98] Sampath Kannan, Aaron Roth, and Juba Ziani. "Downstream effects of affirmative action." In *Proceedings of the Conference on Fairness, Accountability, and Transparency*. 2019, pp. 240–248.

[99] David R. Karger, Sewoong Oh, and Devavrat Shah. "Budget-Optimal Crowdsourcing using Low-rank Matrix Approximations." In *Proc. 49th Annual Conference on Communication, Control, and Computing (Allerton)*. 2011.

[100] David. R. Karger, Sewoong Oh, and Devavrat Shah. "Iterative learning for reliable crowdsourcing systems." In *The 25th Annual Conference on Neural Information Processing Systems (NIPS)*. 2011.

[101] Aditya Khosla, Nityananda Jayadevaprakash, Bangpeng Yao, and Li Fei-Fei. "Novel Dataset for Fine-Grained Image Categorization." In *First Workshop on Fine-Grained Visual Categorization, IEEE Conference on Computer Vision and Pattern Recognition*. Colorado Springs, CO, June 2011.

[102] Juho Kim, Phu Tran Nguyen, Sarah Weir, Philip J Guo, Robert C Miller, and Krzysztof Z Gajos. "Crowdsourcing step-by-step information extraction to enhance existing how-to videos." In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM. 2014, pp. 4017–4026.

[103] Aniket Kittur, Boris Smus, Susheel Khamkar, and Robert E. Kraut. "CrowdForge: Crowdsourcing Complex Work." In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology (UIST)*. 2011.

[104] Jon Kleinberg, Jens Ludwig, Sendhil Mullainathan, and Cass R Sunstein. "Discrimination in the Age of Algorithms." In *Journal of Legal Analysis* 10 (2018).

[105] Jon Kleinberg and Sigal Oren. "Time-inconsistent planning: a computational problem in behavioral economics." In *Proceedings of the fifteenth ACM conference on Economics and computation*. 2014, pp. 547–564.

[106] Jon Kleinberg, Sigal Oren, and Manish Raghavan. "Planning with multiple biases." In *Proceedings of the 2017 ACM Conference on Economics and Computation*. 2017, pp. 567–584.

[107] Robert Kleinberg and Nicole Immorlica. "Recharging bandits." In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science*. 2018, pp. 309–319.

[108] Robert Kleinberg, Aleksandrs Slivkins, and Eli Upfal. "Multi-armed bandits in metric spaces." In *Proceedings of the fortieth annual ACM symposium on Theory of computing*. 2008, pp. 681–690.

[109] Ilan Kremer, Yishay Mansour, and Motty Perry. "Implementing the "Wisdom of the Crowd"." In *Proceedings of the Fourteenth ACM Conference on Electronic Commerce (EC)*. 2013, pp. 605–606.

[110] Anand Kulkarni, Matthew Can, and Björn Hartmann. "Collaboratively Crowdsourcing Workflows with Turkomatic." In *Proceedings of the ACM 2012 Conference on Computer Supported Cooperative Work CSCW*. 2012.

[111] Tze Leung Lai and Herbert Robbins. "Asymptotically efficient adaptive allocation rules." In *Advances in applied mathematics* 6.1 (1985), pp. 4–22.

[112] Edith Law, Ming Yin, Joslin Goh, Kevin Chen, Michael A. Terry, and Krzysztof Z. Gajos. "Curiosity Killed the Cat, but Makes Crowdwork Better." In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI)*. 2016.

[113] Nir Levine, Koby Crammer, and Shie Mannor. "Rotting bandits." In *Advances in neural information processing systems*. 2017, pp. 3074–3083.

[114] Qi Li, Fenglong Ma, Jing Gao, Lu Su, and Christopher J Quinn. "Crowdsourcing high quality labels with a tight budget." In *Proceedings of the ninth acm international conference on web search and data mining*. ACM. 2016, pp. 237–246.

[115] Greg Little, Lydia B. Chilton, Max Goldman, and Robert C. Miller. "TurKit: Human Computation Algorithms on Mechanical Turk." In *Proceedings of the 23Nd Annual ACM Symposium on User Interface Software and Technology (UIST)*. 2010.

[116] Lydia T Liu, Sarah Dean, Esther Rolf, Max Simchowitz, and Moritz Hardt. "Delayed Impact of Fair Machine Learning." In *International Conference on Machine Learning*. 2018, pp. 3150–3158.

[117] Lydia T Liu, Ashia Wilson, Nika Haghtalab, Adam Tauman Kalai, Christian Borgs, and Jennifer Chayes. "The disparate equilibria of algorithmic decision making when individuals invest rationally." In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*. 2020, pp. 381–391.

[118] Yang Liu. "Fair Optimal Stopping Policy for Matching with Mediator." In *Uncertainty in Artificial Intelligence*. 2017.

[119] Yang Liu and Chien-Ju Ho. "Incentivizing High Quality User Contributions: New Arm Generation in Bandit Learning." In *AAAI Conference on Artificial Intelligence (AAAI)*. 2018.

[120] Yang Liu, Goran Radanovic, Christos Dimitrakakis, Debmalya Mandal, and David C Parkes. "Calibrated fairness in bandits." In *Proceedings of the 4th Workshop on Fairness, Accountability, and Transparency in Machine Learning* (2017).

[121] George Loewenstein. "Out of control: Visceral influences on behavior." In *Organizational behavior and human decision processes* 65.3 (1996), pp. 272–292.

[122] Frank Lyman. "Think-Pair-Share: An Expanding Teaching Technique." In *MAA-CIE Cooperative News* (1987).

[123] Stefan Magureanu, Richard Combes, and Alexandre Proutiere. "Lipschitz Bandits: Regret Lower Bound and Optimal Algorithms." In *Conference on Learning Theory*. 2014, pp. 975–999.

[124] George J. Mailath and Larry Samuelson. "Learning under Diverse World Views: Model-Based Inference." In *American Economic Review* 110.5 (May 2020), pp. 1464–1501. DOI: 10.1257/aer.20190080.

[125] Yishay Mansour, Aleksandrs Slivkins, and Vasilis Syrgkanis. "Bayesian incentive-compatible bandit exploration." In *Proceedings of the Sixteenth ACM Conference on Economics and Computation (EC)*. ACM. 2015, pp. 565–582.

[126] David Martin, Benjamin V Hanrahan, Jacki O'Neill, and Neha Gupta. "Being a turker." In *Proceedings of the 17th ACM conference on Computer supported cooperative work & social computing*. ACM. 2014, pp. 224–235.

[127] Winter Mason and Duncane Watts. "Financial Incentives and the "Performance of Crowds"." In *Proceedings of the 1st Human Computation Workshop (HCOMP)*. 2009.

[128] Donald McCloskey and Arjo Klamer. "One quarter of GDP is persuasion." In *The American Economic Review* 85.2 (1995), pp. 191–195.

[129] Daniel McFadden et al. "Conditional logit analysis of qualitative choice behavior." In (1973).

[130] Daniel McFadden. "Econometric models of probabilistic choice." In *Structural analysis of discrete data with econometric applications* 198272 (1981).

[131] Daniel McFadden. "Economic choices." In *American economic review* 91.3 (2001), pp. 351–378.

[132] Stephen Morris. "The Common Prior Assumption in Economic Theory." In *Economics and Philosophy* 11.2 (1995), pp. 227–253. DOI: 10.1017/S0266267100003382.

[133] Hussein Mouzannar, Mesrob I Ohannessian, and Nathan Srebro. "From fair decision making to social equality." In *Proceedings of the Conference on Fairness, Accountability, and Transparency*. 2019, pp. 359–368.

[134] Lev Muchnik, Sinan Aral, and Sean J Taylor. "Social influence bias: A randomized experiment." In *Science* 341.6146 (2013), pp. 647–651.

[135] Saumik Narayanan, Guanghui Yu, **Wei Tang**, Chien-Ju Ho, and Ming Yin. "How Does Predictive Information Affect Human Ethical Preferences?" In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society* (2022 (To appear)).

[136] Gergely Neu, Andras Antos, András György, and Csaba Szepesvári. "Online Markov decision processes under bandit feedback." In *Advances in Neural Information Processing Systems (NIPS)*. 2010, pp. 1804–1812.

[137] John von Neumann and Oscar Morgenstern. *Theory of Games and Economic Behavior*. Princeton University Press, 1944.

[138] Edward Newell and Derek Ruths. "How One Microtask Affects Another." In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. 2016.

[139] Jon Noronha, Eric Hysen, Haoqi Zhang, and Krzysztof Z Gajos. "Platemate: crowdsourcing nutritional analysis from food photographs." In *Proceedings of the 24th annual ACM symposium on User interface software and technology*. ACM. 2011, pp. 1–12.

[140] Ziad Obermeyer, Brian Powers, Christine Vogeli, and Sendhil Mullainathan. "Dissecting racial bias in an algorithm used to manage the health of populations." In *Science* 366.6464 (2019), pp. 447–453.

[141] Ronald Ortner, Daniil Ryabko, Peter Auer, and Rémi Munos. "Regret bounds for restless Markov bandits." In *International Conference on Algorithmic Learning Theory (ALT)*. Springer. 2012, pp. 214–228.

[142] Yiangos Papanastasiou, Kostas Bimpikis, and Nicos Savva. "Crowdsourcing exploration." In *Management Science* (2017).

[143] Vishakha Patil, Ganesh Ghalme, Vineet Nair, and Y Narahari. "Achieving fairness in the stochastic multi-armed bandit problem." In *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 34. 04. 2020, pp. 5379–5386.

[144] Ciara Pike-Burke, Shipra Agrawal, Csaba Szepesvari, and Steffen Grunewalder. "Bandits with Delayed, Aggregated Anonymous Feedback." In *International Conference on Machine Learning*. 2018, pp. 4105–4113.

[145] Drazen Prelec. "The probability weighting function." In *Econometrica* (1998), pp. 497–527.

[146] Vikas Raykar, Shipeng Yu, Linda Zhao, Gerardo Valadez, Charles Florin, Luca Bogoni, and Linda Moy. "Learning from crowds." In *Journal of Machine Learning Research* 11 (2010), pp. 1297–1322.

[147] Luis Rayo and Ilya Segal. "Optimal information disclosure." In *Journal of political Economy* 118.5 (2010), pp. 949–987.

[148] Daniela Retelny et al. "Expert Crowdsourcing with Flash Teams." In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology (UIST)*. 2014.

[149] Matthew Richardson, Ewa Dominowska, and Robert Ragno. "Predicting clicks: estimating the click-through rate for new ads." In *Proceedings of the 16th international conference on World Wide Web (WWW)*. ACM. 2007, pp. 521–530.

[150] Marc Oliver Rieger and Mei Wang. "Cumulative prospect theory and the St. Petersburg paradox." In *Economic Theory* 28.3 (2006), pp. 665–679.

[151] Herbert Robbins and Sutton Monro. "A stochastic approximation method." In *The annals of mathematical statistics* (1951), pp. 400–407.

[152] Jakob Rogstadius, Vassilis Kostakos, Aniket Kittur, Boris Smus, Jim Laredo, and Maja Vukovic. "An assessment of intrinsic and extrinsic motivation on task performance in crowdsourcing markets." In *ICWSM* 11 (2011), pp. 17–21.

[153] Niloufar Salehi, Lilly C Irani, Michael S Bernstein, Ali Alkhatib, Eva Ogbe, Kristy Milland, et al. "We are dynamo: Overcoming stalling and friction in collective action for crowd workers." In *Proceedings of the 33rd annual ACM conference on human factors in computing systems*. ACM. 2015, pp. 1621–1630.

[154] Matthew J Salganik, Peter Sheridan Dodds, and Duncan J Watts. "Experimental study of inequality and unpredictability in an artificial cultural market." In *Science* 311.5762 (2006), pp. 854–856.

[155] Mike Schaekermann, Joslin Goh, Kate Larson, and Edith Law. "Resolvable vs. Irresolvable Disagreement: A Study on Worker Deliberation in Crowd Work." In *Proceedings of the 21st ACM Conference on Computer Supported Cooperative Work & Social Computing (CSCW)*. 2018.

[156] Sven Schmit and Carlos Riquelme. "Human Interaction with Recommendation Systems." In *International Conference on Artificial Intelligence and Statistics*. 2018, pp. 862–870.

[157] Rajiv Sethi and Muhamet Yildiz. "Communication With Unknown Perspectives." In *Econometrica* 84.6 (2016), pp. 2029–2069.

[158] Nihar Bhadresh Shah and Denny Zhou. "Double or Nothing: Multiplicative Incentive Mechanisms for Crowdsourcing." In *The 29th Annual Conference on Neural Information Processing Systems (NIPS)*. 2015.

[159] Aaron D. Shaw, John J. Horton, and Daniel L. Chen. "Designing incentives for inexpert human raters." In *Proceedings of the ACM 2011 conference on Computer supported cooperative work (CSCW)*. 2011.

[160] Ruben Sipos, Arpita Ghosh, and Thorsten Joachims. "Was This Review Helpful to You?: It Depends! Context and Voting Patterns in Online Content." In *Proceedings of the 23rd International Conference on World Wide Web (WWW)*. 2014, pp. 337–348.

[161] Aleksandrs Slivkins. "Contextual bandits with similarity information." In *The Journal of Machine Learning Research* 15.1 (2014), pp. 2533–2568.

[162] Aleksandrs Slivkins and Eli Upfal. "Adapting to a Changing Environment: the Brownian Restless Bandits." In *Conference on Learning Theory*. 2008, pp. 343–354.

[163] Kenneth A Small. "A discrete choice model for ordered alternatives." In *Econometrica: Journal of the Econometric Society* (1987), pp. 409–424.

[164] Lones Smith and Peter Sørensen. "Pathological Outcomes of Observational Learning." In *Econometrica* 68.2 (2000), pp. 371–398.

[165] Michelle K Smith, William B Wood, Wendy K Adams, Carl Wieman, Jennifer K Knight, Nancy Guild, and Tin Tin Su. "Why peer discussion improves student performance on in-class concept questions." In *Science* 323.5910 (2009), pp. 122–124.

[166] Ola Svenson. "Process descriptions of decision making." In *Organizational behavior and human performance* 23.1 (1979), pp. 86–112.

[167] Cem Tekin and Mingyan Liu. "Online algorithms for the multi-armed bandit problem with markovian rewards." In *Proceedings of the 48th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*. 2010.

[168] Joshua Tenenbaum. "Bayesian Modeling of Human Concept Learning." In *Advances in Neural Information Processing Systems*. MIT Press, 1999.

[169] Kenneth E Train. *Discrete choice methods with simulation*. Cambridge university press, 2009.

[170] Amos Tversky and Daniel Kahneman. "Advances in prospect theory: Cumulative representation of uncertainty." In *Journal of Risk and uncertainty* 5.4 (1992), pp. 297–323.

[171] Amos Tversky and Daniel Kahneman. "Judgment under Uncertainty: Heuristics and Biases." In *Science* 185.4157 (1974), pp. 1124–1131. DOI: 10.1126/science.185.4157.1124.

[172] Amos Tversky and Peter Wakker. "Risk attitudes and decision weights." In *Econometrica: Journal of the Econometric Society* (1995), pp. 1255–1280.

[173] **Wei Tang** and Chien-Ju Ho. "Bandit Learning with Biased Human Feedback." In *AAMAS*. 2019, pp. 1324–1332.

[174] **Wei Tang** and Chien-Ju Ho. "On the Bayesian Rational Assumption in Information Design." In *HCOMP*. 2021.

[175] **Wei Tang**, Chien-Ju Ho, and Yang Liu. "Bandit Learning with Delayed Impact of Actions." In *NeurIPS* (2021).

[176] **Wei Tang**, Chien-Ju Ho, and Yang Liu. "Differentially Private Contextual Dynamic Pricing." In *AAMAS*. 2020, pp. 1368–1376.

[177] **Wei Tang**, Chien-Ju Ho, and Yang Liu. "Linear models are robust optimal under strategic behavior." In *AISTATS*. 2021, pp. 2584–2592.

[178] **Wei Tang**, Chien-Ju Ho, and Yang Liu. "Optimal Query Complexity of Secure Stochastic Convex Optimization." In *NeurIPS*. 2020.

[179] **Wei Tang**, Chien-Ju Ho, and Ming Yin. "Leveraging peer communication to enhance crowdsourcing." In *The World Wide Web Conference*. 2019, pp. 1794–1805.

[180] Jacob Whitehill, Ting-fan Wu, Jacob Bergsma, Javier R. Movellan, and Paul L. Ruvolo. "Whose vote should count more: Optimal integration of labels from labelers of unknown expertise." In *Advances in Neural Information Processing Systems (NIPS)*. 2009.

[181] George Wu and Richard Gonzalez. "Curvature of the probability weighting function." In *Management science* 42.12 (1996), pp. 1676–1690.

[182] Ming Yin, Yiling Chen, and Yu-An Sun. "The Effects of Performance-Contingent Financial Incentives in Online Labor Markets." In *Proceedings of the 27th AAAI Conference on Artificial Intelligence (AAAI)*. 2013.

[183] Ming Yin, Mary L Gray, Siddharth Suri, and Jennifer Wortman Vaughan. "The communication network within the crowd." In *Proceedings of the 25th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee. 2016, pp. 1293–1303.

[184] Guanghui Yu and Chien-Ju Ho. "Environment Design for Biased Decision Makers." In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*. 2022.

[185] Lixiu Yu, Paul André, Aniket Kittur, and Robert Kraut. "A comparison of social, learning, and financial strategies on crowd engagement and output quality." In *Proceedings of the 17th ACM conference on Computer supported cooperative work & social computing*. ACM. 2014, pp. 967–978.

[186] Xueru Zhang, Mohammadmahdi Khaliligarekani, Cem Tekin, et al. "Group Retention when Using Machine Learning in Sequential Decision Making: the Interplay between User Dynamics and Fairness." In *Advances in Neural Information Processing Systems*. 2019, pp. 15243–15252.

[187] Xueru Zhang, Ruibo Tu, Yang Liu, Mingyan Liu, Hedvig Kjellström, Kun Zhang, and Cheng Zhang. "How do fair decisions fare in long-term qualification?" In (2020).

[188] Yudian Zheng, Guoliang Li, Yuanbing Li, Caihua Shan, and Reynold Cheng. "Truth inference in crowdsourcing: Is the problem solved?" In *Proceedings of the VLDB Endowment* 10.5 (2017), pp. 541–552.

Appendix A

Additional Proofs from Chapter 2

A.1 Useful Lemmas

In this section, we list some useful lemmas in our analysis.

Lemma A.1.1. ***(Stirling's approximation for gamma function quotient)*** *Given* $\alpha > 0, \beta > 0$ *and when* $x \to \infty$*, we have*

$$\frac{\Gamma(x+\beta)}{\Gamma(x+\alpha)} = x^{\beta-\alpha}.$$

Proof. From Stirling's Approximation, and let $\gamma = \beta - \alpha$

$$\begin{aligned}\frac{\Gamma(x+1+\beta)}{\Gamma(x+1+\alpha)} &= \frac{\sqrt{2\pi(x+\beta)}(\frac{x+\beta}{e})^{x+\beta}}{\sqrt{2\pi(x+\alpha)}(\frac{x+\alpha}{e})^{x+\alpha}} \\ &= (1+\frac{\gamma}{x+\alpha})^{x+\alpha+1/2}(1+\frac{\beta}{x})^{\gamma}(\frac{x}{e})^{\gamma}.\end{aligned}$$

Since $\lim_{x\to\infty}(1+y/x)^x = e^y$ and $\lim_{x\to\infty}(1+y/x) = 1$, we have

$$\lim_{x\to\infty}\frac{\Gamma(x+1+\beta)}{\Gamma(x+1+\alpha)} = x^{\beta-\alpha}.$$

□

Lemma A.1.2. ***(Concentration property for bounded random variable)*** *Given any bounded random variable Y, and a $L-$Lipschitz function $g(\cdot)$, then for $\forall\lambda\in\mathbb{R}$,*

$$\mathbb{E}[\exp(\lambda g(Y))] \le \exp(\lambda^2 L^2/2).$$

A.2 Proofs and Simulations in Bandits with Avg-Herding Feedback Model

A.2.1 Proof of Lemma 2.3.1

Proof. Our goal is to prove $\mathbb{P}(\lim_{t\to\infty}\rho_t \in \mathcal{S}_\theta) = 1$ for $\mathcal{S}_\theta := \{\rho : \rho - F(\theta,\rho) = 0\}$. Since $F(\theta,\rho)$ is continuous, for all $\epsilon > 0$, we define the following two sets:

$$U_\epsilon := \{\rho : F(\theta,\rho) - \rho > \epsilon\},$$
$$D_\epsilon := \{\rho : F(\theta,\rho) - \rho < -\epsilon\}.$$

If we can show that $\mathbb{P}(\lim_{t\to\infty}\rho_t \notin U_\epsilon) = 1$ and $\mathbb{P}(\lim_{t\to\infty}\rho_t \notin D_\epsilon) = 1$ for any arbitrarily small ϵ, the proof is completed. We first establish the following fact: If there exists some

$t_0 \geq 0$ such that $\rho_{t_0} \in U_\epsilon$, we must have

$$\mathbb{P}(\lim_{t\to\infty} \rho_t \notin U_\epsilon) = 1. \tag{A.1}$$

To prove (A.1), consider the first exit time τ of $\{\rho_t\}$ from U_ϵ, namely, τ is the smallest time round such that $\rho_t \notin U_\epsilon$ (or ∞ if $\{\rho_t\}$ never leaves U_ϵ). Let $\tau_t = \min\{\tau, t\}$ denote the minimum of τ and $t \geq t_0$. It is easy to see that $\forall k \geq t_0 + 1$, the event $\{\tau_t \geq k\}$ is $\mathcal{F}_{k-1}-$ measurable, thus

$$\begin{aligned}
1 \geq \mathbb{E}[\rho_{\tau_t}] \geq \mathbb{E}[\rho_{\tau_t} - \rho_{t_0}] &= \mathbb{E}[\rho_{t_0+1} - \rho_{t_0} + \rho_{t_0+2} - \rho_{t_0+1} + ... + \rho_{\tau_t} - \rho_{\tau_t - 1}] \\
&= \mathbb{E}\bigg[\sum_{k=t_0+1}^{t} (\rho_k - \rho_{k-1}) \mathbb{1}\{\tau_t \geq k\}\bigg] \\
&\geq \mathbb{E}\bigg[\sum_{k=t_0+1}^{t} \mathbb{E}\big[\rho_k - \rho_{k-1} | \mathcal{F}_{k-1}\big] \mathbb{1}\{\tau = \infty\}\bigg],
\end{aligned}$$

where $\mathbb{1}\{\mathcal{E}\}$ is the indicator function of event $\mathcal{E}$. We aslo have

$$\begin{aligned}
\mathbb{E}[\rho_k - \rho_{k-1} | \mathcal{F}_{k-1}] &= \mathbb{E}[\rho_k - \rho_{k-1} | \rho_{k-1}] \\
&= \mathbb{E}\big[\eta_k\big(F(\theta, \rho_{k-1}) - \rho_{k-1}\big) | \rho_{k-1}\big] \\
&\qquad \text{By the update rule defined in (2.2) and } \mathbb{E}[\xi_k | \mathcal{F}_{k-1}] = 0 \\
&\geq \epsilon/k. \qquad\qquad \text{By the fact that } \rho_{k-1} \in U_\epsilon
\end{aligned}$$

Then for $\forall t \geq t_0$, $\forall t_0 \geq 0$,

$$\epsilon \sum_{k=t_0+1}^{t} \frac{\mathbb{P}(\tau = \infty)}{k} \leq 1.$$

Since $\sum_k 1/k$ is divergent, then the probability that $\tau = \infty$ must be zero, i.e., $\mathbb{P}(\tau = \infty) = 0$. This completes the proof of (A.1).

Similarly, we can also prove $\mathbb{P}(\lim_{t\to\infty}\rho_t \notin D_\epsilon) = 1$. Since ϵ is arbitrarily selected, we have $\mathbb{P}(\lim_{t\to\infty}|\rho_t - F(\theta,\rho_t)| \le \epsilon) = 1$, which completes the proof. □

A.2.2 Proof of Corollary 2.3.2

Proof. Since G is strongly convex with respect to ρ, we have $\nabla^2_\rho G > 0$, i.e., $\nabla_\rho F \le 1$. By Banach fixed-point theorem, we know that $F(\theta,\rho)$ has a unique fixed point ρ^*_θ in $(0,1)$.

Define $h(\theta,\rho) = \rho - F(\theta,\rho)$. We now proceed to prove that this fixed point ρ^*_θ is globally asymptotically stable for $h(\theta,\rho)$. Consider a Lyapunov function $V(\theta,\rho) : \rho \to \frac{1}{2}(\rho-\rho^*_\theta)^2$ for the ODE defined in (2.2). We have $V(\theta,\rho) \ge 0$ for $\rho \in (0,1)$. And $V(\theta,\rho) = 0$ if and only if $\rho = \rho^*_\theta$. Furthermore, we have

$$\frac{d}{d_t}V(\theta,\rho) = (\rho-\rho^*_\theta)\frac{d}{d_t}\rho = (\rho^*_\theta - \rho)h(\theta,\rho).$$

By assumption, $F(\theta,\rho)$ is a contraction mapping function, it is easy to see that $h(\theta,\rho)$ is strictly increasing in ρ, i.e., $\partial h(\theta,\rho)/\partial\rho > 0$. So we have $h(\theta,\rho) \ge (\le) 0$ for $\rho \ge (\le)\rho^*_\theta$, which means $dV(\theta,\rho)/d_t \le 0$ for all $\rho \in (0,1)$ and $dV(\theta,\rho)/d_t < 0$ for all $\rho \in (0,1)\setminus\rho^*_\theta$. This proves that ρ^*_θ is the asymptotically stable point of $h(\theta,\rho)$. □

A.2.3 Proof of Theorem 2.3.3

We can decompose $z_t := |\rho_t - \rho^*|$ for each $t \ge 0$ into two parts. the *empirical iterate error* $|\rho_t - \rho^*| - \mathbb{E}[|\rho_t - \rho^*|$ and the *expectation error* $\mathbb{E}[|\rho_t - \rho^*|]$:

$$z_t := |\rho_t - \rho^*| = (|\rho_t - \rho^*| - \mathbb{E}[|\rho_t - \rho^*|]) + \mathbb{E}[|\rho_t - \rho^*|]. \tag{A.2}$$

To derive the probabilistic tail bound for $|\rho_t - \rho^*|$, we bound the empirical iterate error and expectation error separately. Below, we first give the high probability bound for the empirical iterate error:

Lemma A.2.1. *Given the average feedback dynamics $\{\rho_t\}_{t\geq 0}$ (ρ_0 is the prior information) defined in (2.2), and under the assumptions of (A1 - A2), and G is strongly convex, then for any $\delta > 0$, we have:*

$$\mathbb{P}(|\rho_t - \rho^*| - \mathbb{E}[|\rho_t - \rho^*|] \geq \delta) \leq \exp\left(\frac{-\delta^2}{2\sum_{i=1}^{t}\eta_i^2\big(\prod_{j=i}^{t-1}(1 - 2\bar{\lambda}\eta_{j+1} + \eta_{j+1}^2(L_h^\rho)^2)\big)}\right),$$

where $L_h^\rho = 1 - L_F^\rho$.

Proof. Notice that by introducing a telescoping sum of martingale differences, we could rewrite $|\rho_t - \rho^*| - \mathbb{E}[|\rho_t - \rho^*|]$ as follows

$$\begin{aligned}|\rho_t - \rho^*| - \mathbb{E}[|\rho_t - \rho^*|] &= \sum_{i=1}^{t}\mathbb{E}[|z_t||\mathcal{F}_i] - \mathbb{E}[|z_t||\mathcal{F}_{i-1}] \\ &= \sum_{i=1}^{t} g_i(\rho_{i-1}, \xi_i) - \mathbb{E}[g_i(\rho_{i-1}, \xi_i)|\mathcal{F}_{i-1}],\end{aligned}$$

where $g_i(\rho_i, \xi) = \mathbb{E}[|\rho_t - \rho^*||\rho_i, \mathcal{F}_{i-1}]$. Let $L_h^\rho = 1 - L_F^\rho$ and $\mathcal{G}_i = g_i - \mathbb{E}[g_i|\mathcal{F}_{i-1}]$. Recall that $\mathcal{F}_i := \sigma(\xi_j, j \leq i)$ and $\{\mathcal{F}_i\}_{i\in\mathbb{N}}$ denotes the natural filtration.

Let $H(\theta, \rho, \xi) = \rho - F(\theta, \rho) + \xi$. We then reduce the proof of empirical iterate error by establishing a Lipschitz continuous property of $\mathcal{G}_i$ on martingale difference ξ_i. We also write out the superscript of ρ_j as $\rho_j^{\rho,i}$ to explicitly express the dependency of $\{\rho_t\}$ on a given initial starting time i such that $\rho_i = \rho$. Recall that $h(\theta, \rho) = \mathbb{E}[H(\theta, \rho, \xi)]$. By introducing a

martingale difference $\Delta_{j+1}^{\rho,i} = H(\theta, \rho_j^{\rho,i}, \xi_{j+1}) - h(\theta, \rho_j^{\rho,i})$, we then have

$$\begin{aligned}
|\rho_{j+1}^{\rho,i} - \rho_{j+1}^{\rho',i}|^2 &= |\rho_j^{\rho,i} - \rho_j^{\rho',i} - \eta_{j+1}(H(\theta, \rho_j^{\rho,i}, \xi_{j+1}) - H(\theta, \rho_j^{\rho',i}, \xi_{j+1}))|^2 \\
&= (\rho_j^{\rho,i} - \rho_j^{\rho',i})^2 - 2\eta_{j+1}(\rho_j^{\rho,i} - \rho_j^{\rho',i})\big(H(\theta, \rho_j^{\rho,i}, \xi_{j+1}) - H(\theta, \rho_j^{\rho',i}, \xi_{j+1})\big) \\
&\quad + \eta_{i+1}^2\big(H(\theta, \rho_j^{\rho,i}, \xi_{j+1}) - H(\theta, \rho_j^{\rho',i}, \xi_{j+1})\big)^2 \\
&= (\rho_j^{\rho,i} - \rho_j^{\rho',i})^2 - 2\eta_{j+1}(\rho_j^{\rho,i} - \rho_j^{\rho',i})(h(\theta, \rho_j^{\rho,i}) - h(\theta, \rho_j^{\rho',i})) \\
&\quad - 2\eta_{j+1}(\rho_j^{\rho,i} - \rho_j^{\rho',i})(\Delta_{j+1}^{\rho,i} - \Delta_{j+1}^{\rho',i}) + \eta_{j+1}^2\big(H(\theta, \rho_j^{\rho,i}, \xi_{j+1}) - H(\theta, \rho_j^{\rho',i}, \xi_{j+1})\big)^2.
\end{aligned}$$

Applying the Lipschitz continuity of $H(\cdot)$ w.r.t. ρ, by the strongly convex property of $G(\cdot)$, the above equation gives us following

$$\begin{aligned}
|\rho_{j+1}^{\rho,i} - \rho_{j+1}^{\rho',i}|^2 &\leq (\rho_j^{\rho,i} - \rho_j^{\rho',i})^2 - 2\bar{\lambda}\eta_{j+1}(\rho_j^{\rho,i} - \rho_j^{\rho',i})^2 \\
&\quad - 2\eta_{j+1}(\rho_j^{\rho,i} - \rho_j^{\rho',i})(\Delta_{j+1}^{\rho,i} - \Delta_{j+1}^{\rho',i}) + \eta_{i+1}^2(L_h^\rho)^2(\rho_j^{\rho,i} - \rho_j^{\rho',i})^2 \\
&= (\rho_j^{\rho,i} - \rho_j^{\rho',i})^2(1 - 2\bar{\lambda}\eta_{j+1} + \eta_{j+1}^2(L_h^\rho)^2) - 2\eta_{j+1}(\rho_j^{\rho,i} - \rho_j^{\rho',i})(\Delta_{j+1}^{\rho,i} - \Delta_{j+1}^{\rho',i}).
\end{aligned}$$

Then taking induction on j from i to t, we have

$$\begin{aligned}
|\rho_t^{\rho,i} - \rho_t^{\rho',i}|^2 &\leq (\rho - \rho')^2 \prod_{j=i}^{t-1}(\eta_{j+1}^2(L_h^\rho)^2 - 2\bar{\lambda}\eta_{j+1} + 1) \\
&\quad - 2\prod_{j=i}^{t-1}(1 - 2\bar{\lambda}\eta_{j+1} + \eta_{j+1}^2(L_h^\rho)^2)\sum_{j=1}^{t-1}\frac{\eta_{j+1}}{\prod_{l=i}^{j}(1 - 2\bar{\lambda}\eta_{l+1} + \eta_{l+1}^2(L_h^\rho)^2)}(\rho_j^{\rho,i} - \rho_j^{\rho',i})(\Delta_{j+1}^{\rho,i} - \Delta_{j+1}^{\rho',i})
\end{aligned}$$

Taking the expectation on both sides, applying the tower property of expectation and by the fact that $\mathbb{E}[\Delta_j^{\rho,i}] = 0$, we have

$$\mathbb{E}[|\rho_t^{\rho,i} - \rho_t^{\rho',i}|^2] \leq (\rho - \rho')^2 \prod_{j=i}^{t-1}(\eta_{j+1}^2(L_h^\rho)^2 - 2\bar{\lambda}\eta_{j+1} + 1).$$

Back to our error decomposition, we have the following Lipschitz bound for the function $g_i(\cdot)$,

$$\begin{aligned}
|g_i(\rho,\xi) - g_i(\rho,\xi')| &= \big|\mathbb{E}\big[|\rho_t^{\rho+\eta_i H(\theta,\rho,\xi),i} - \rho^*|\big] - \mathbb{E}\big[|\rho_t^{\rho+\eta_i H(\theta,\rho,\xi'),i} - \rho^*|\big]\big| \\
&\le \mathbb{E}\big[|\rho_t^{\rho+\eta_i H(\theta,\rho,\xi),i} - \rho_t^{\rho+\eta_i H(\theta,\rho,\xi'),i}|\big] \\
&\le \eta_i\big|\xi - \xi'\big|\bigg(\prod_{j=i}^{t-1}(\eta_{j+1}^2(L_h^\rho)^2 - 2\bar{\lambda}\eta_{j+1} + 1\bigg)^{1/2}.
\end{aligned}$$

The above inequality shows $g_i(\cdot)$ is a Lipschitz continuous function defined on random variable ξ given $\mathcal{F}_{i-1}$ with the Lipschitz constant equaling to $L_{g_i} = \eta_i\big(\prod_{j=i}^{t-1}(\eta_{j+1}^2(L_h^\rho)^2 - 2\bar{\lambda}\eta_{j+1}+1\big)^{1/2}$. Thus,

$$\begin{aligned}
\mathbb{P}(|\rho_t - \rho^*| - \mathbb{E}[|\rho_t - \rho^*|] \ge \delta) &= \mathbb{P}(\sum_{i=1}^{t}\mathcal{G}_i \ge \delta) \\
&\le \exp(-\gamma\delta)\mathbb{E}[\exp(\gamma\sum_{i=1}^{t}\mathcal{G}_i)] \\
&= \exp(-\gamma\delta)\mathbb{E}[\exp(\gamma\sum_{i=1}^{t-1}\mathcal{G}_i)]\mathbb{E}[\exp(\gamma\mathcal{G}_t)|\mathcal{F}_{t-1}].
\end{aligned}$$

Now it shows that $\mathcal{G}_t$ is a L_{g_t}-Lipschitz function conditional on $\mathcal{F}_{t-1}$. By invoking a martingale concentration bound in Lemma A.1.2, we have:

$$\mathbb{E}[\exp(\gamma\mathcal{G}_t)|\mathcal{F}_{t-1}] \le \exp\left(\frac{\gamma^2 L_{g_t}^2}{2}\right).$$

By induction on i, we have

$$\mathbb{P}(|\rho_t - \rho^*| - \mathbb{E}[|\rho_t - \rho^*|] \ge \delta) \le \exp(-\gamma\delta)\exp\left(\frac{\gamma^2\sum_{i=1}^{t}L_{g_i}^2}{2}\right).$$

We can then finish the proof by optimizing with respect to γ.

□

We now proceed to bound the expectation error $\mathbb{E}[|\rho_t - \rho^*|]$:

Lemma A.2.2. *Given the ratio dynamics $\{\rho_t\}_{t\geq 0}$ (ρ_0 is the prior information) defined in (2.2), and under the assumptions of (A1 -A2), and G is strongly convex, then we have*

$$\mathbb{E}[|\rho_t - \rho^*|] \leq \exp(-\bar{\lambda} S_t)|\rho_0 - \rho^*| + \sqrt{\sum_{i=0}^{t-1} \eta_{i+1}^2 \exp(-2\bar{\lambda}(S_t - S_{i+1}))},$$

where $S_t = \sum_{i=1}^{t} \eta_i$.

Proof. We define the following

$$\begin{aligned} z_{t+1} := \rho_{t+1} - \rho^* &= \rho_t - \rho^* - \eta_{t+1} H(\theta, \rho_t, \xi_{t+1}) \\ &= \rho_t - \rho^* - \eta_{t+1}(h(\theta, \rho_t) - \Delta Y_{t+1}), \end{aligned}$$

where $\Delta Y_{t+1} = h(\theta, \rho_t) - H(\theta, \rho_t, \xi_{t+1}) = \mathbb{E}[H(\theta, \rho_t, \xi_{t+1})|\mathcal{F}_t] - H(\theta, \rho_t, \xi_{t+1})$. Rewriting above equation as follows

$$z_{t+1} = \rho_t - \rho^* - \eta_{t+1}(\rho_t - \rho^*) \int_0^1 (\partial h(\theta, \rho^* + \alpha(\rho_t - \rho^*))/\partial \rho) d\alpha + \eta_{t+1} \Delta Y_{t+1}.$$

Let $\mathcal{J}_t = \int_0^1 (\partial h(\theta, \rho^* + \alpha(\rho_t - \rho^*))/\partial \rho) d\alpha$, then we have

$$\begin{aligned} z_{t+1} &= \rho_t - \rho^* - \eta_{t+1}(\rho_t - \rho^*)\mathcal{J}_t + \eta_{t+1}\Delta Y_{t+1} \\ &= z_t(1 - \eta_{t+1}\mathcal{J}_t) + \eta_{t+1}\Delta Y_{t+1}. \end{aligned}$$

Taking a square operator on both sides, expanding and then taking expectation, we can get

$$\begin{aligned}\mathbb{E}[|z_{t+1}|^2] &= \mathbb{E}[|z_t(1-\eta_{t+1}\mathcal{J}_t)+\eta_{t+1}\Delta Y_{t+1}|^2] \\ &= \mathbb{E}[|z_t(1-\eta_{t+1}\mathcal{J}_t)|^2] + 2\mathbb{E}[z_t(1-\eta_{t+1}\mathcal{J}_t)\eta_{t+1}\Delta Y_{t+1}] + \mathbb{E}[|\eta_{t+1}\Delta Y_{t+1}|^2] \\ &= (1-\eta_{t+1}\mathcal{J}_t)^2\mathbb{E}[|z_t|^2] + \eta_{t+1}^2\mathbb{E}[|\Delta Y_{t+1}|^2],\end{aligned}$$

where the last equality is due to the martingale difference property of ΔY_{t+1}. Notice that by the property of strongly convex G, namely, there exists a global stable equilibrium point of h, we have $|1-\eta_{t+1}\mathcal{J}_t| \le \exp(-\bar{\lambda}\eta_{t+1})$. Thus,

$$\mathbb{E}[|z_{t+1}|^2] \le \exp(-2\bar{\lambda}\eta_{t+1})\mathbb{E}[|z_t|^2] + \eta_{t+1}^2.$$

Finally, taking the induction from $t=1$ will give us the following

$$\mathbb{E}[|z_t|^2] \le z_0^2\exp(-2\bar{\lambda}S_t) + \sum_{i=0}^{t-1}\eta_{i+1}^2\exp(-2\bar{\lambda}(S_t-S_{i+1})),$$

where $S_t = \sum_{i=1}^t \eta_i$. □

Combining the above empirical iterate error and expectation error completes the proof of Theorem 2.3.3.

Convergence Analysis Let $M_t = \prod_{i=1}^t(1-2\bar{\lambda}/i+(L_h^\rho)^2/i^2)$, then $\sum_{i=1}^t L_i = M_t\sum_{i=1}^t \eta_i^2/M_i$. By $1+x < e^x$, it is immediate to see that $M_t \le \prod_{i=1}^t \exp\big(-2\bar{\lambda}/i + (L_h^\rho)^2/i^2\big) = \exp\big(\sum_{i=1}^t(-2\bar{\lambda}/i+(L_h^\rho)^2/i^2)\big) = \exp\big(-2\bar{\lambda}\ln t + (L_h^\rho)^2\pi^2/6\big) = \exp\big((L_h^\rho)^2\pi^2/6\big)t^{-2\bar{\lambda}}$. Thus,

- when $\bar{\lambda}\in(0,1/2]$, we have $i^2\prod_{j=1}^i(1-2\bar{\lambda}/j+(L_h^\rho)^2/j^2) > i^2(1-2\bar{\lambda})\prod_{j=2}^i(1-1/j) = i(1-2\bar{\lambda})$. Thus, $\sum_{i=1}^t\eta_i^2/M_i$ is summable, and $\sum_{i=1}^t\eta_i^2/M_i \le \sum_{i=1}^t\frac{1}{i^2\prod_{j=1}^i(1-2\bar{\lambda}/j)} \le C_0$,

where $C_0 = \lim_{t\to\infty} \sum_{i=1}^{t} \frac{1}{i^2 \prod_{j=1}^{i}(1-2\bar{\lambda}/j)}$. For simplicity, let $C_1 = C_0 \exp((L_h^\rho)^2\pi^2/6)$, then we have $\sum_{i=1}^{t} L_i \leq C_1 t^{-2\bar{\lambda}} = \mathcal{O}(t^{-2\bar{\lambda}})$;

- when $\bar{\lambda} \in (1/2, \infty)$, it can be proved by comparisons with integrals that $\sum_{i=1}^{t} \eta_i^2/M_i \leq Ct^{(2\bar{\lambda}-1)}$, where C is a constant which is only dependent on $\bar{\lambda}$. Thus, let $C_2 = \frac{(2\bar{\lambda}+1)\exp((L_h^\rho)^2\pi^2/6)}{4(2^{2\bar{\lambda}-1}-1)}$, we'll have $\sum_{i=1}^{t} L_i \leq C_2 t^{-1} = \mathcal{O}(t^{-1})$.

For the expectation error δ_t, we know that $S_t = \Theta(\ln t)$. Thus, we have $\sum_{i=0}^{t-1} \eta_{i+1}^2 \exp(-2\bar{\lambda}(S_t - S_{i+1})) = \sum_{k=1}^{t}(1/k^2)\exp(-2\bar{\lambda}\sum_{i=k}^{t} 1/i) \leq \sum_{k=1}^{t}(1/k^2)\exp(-2\bar{\lambda}\ln t/k) \leq t^{-2\bar{\lambda}}\sum_{k=1}^{t} 1/k^{2-2\bar{\lambda}}$. By comparing the sums with integrals,

- when $\bar{\lambda} \in (0, 1/2)$, we have $\lim_{t\to\infty} t^{-2\bar{\lambda}}\sum_{k=1}^{t} 1/k^{2-2\bar{\lambda}} = \mathcal{O}(t^{-\bar{\lambda}})$;
- when $\bar{\lambda} = 1/2$, we have $\lim_{t\to\infty} t^{-2\bar{\lambda}}\sum_{k=1}^{t} 1/k^{2-2\bar{\lambda}} = \Theta(t^{-1}\ln t)$;
- when $\bar{\lambda} \in (1/2, \infty)$, we have $\lim_{t\to\infty} t^{-2\bar{\lambda}}\sum_{k=1}^{t} 1/k^{2-2\bar{\lambda}} = \mathcal{O}(1/\sqrt{t})$.

Hence, we have $\delta_t \to 0$ when $t \to \infty$.

A.2.4 Proof of Theorem 2.3.5

We first prove that a small deviation of $\rho_{k,t}$ leads to a small deviation of the quality estimator $\hat{\theta}_{k,t}$, as summarized in the following lemma:

Lemma A.2.3. *Assume there exist $D_\theta > 0$ and $D_\rho \in (0,1)$, such that for any $0 < \theta_2 < \theta_1 < 1$, $D_\theta(\theta_1 - \theta_2) \leq F(\theta_1, \rho) - F(\theta_2, \rho)$; and for any $0 < \rho_2 < \rho_1 < 1$, $D_\rho(\rho_1 - \rho_2) \leq F(\theta, \rho_1) - F(\theta, \rho_2)$. Let $\hat{\theta}_1$ and $\hat{\theta}_2$ be the quality estimates for ρ_1 and ρ_2, as specified in Equation (2.3). Then the following holds*

$$|\hat{\theta}_1 - \hat{\theta}_2| \leq \frac{1 - D_\rho}{D_\theta}|\rho_1 - \rho_2|.$$

Proof. Since the quality estimate $\hat{\theta}$ is chosen such that $F(\hat{\theta}, \rho) = \rho$, we have

$$F(\hat{\theta}_1, \rho_1) - F(\hat{\theta}_2, \rho_2) = \rho_1 - \rho_2$$

From Corollary 2.3.2, we know that when $\rho_1 = \rho_2$, $\hat{\theta}_1 = \hat{\theta}_2$. Therefore the lemma statement is true. Below we discuss the case $\rho_1 > \rho_2$. We first argue that when $\rho_1 > \rho_2$, $\hat{\theta}_1 > \hat{\theta}_2$. Assume by contradiction that $\hat{\theta}_1 < \hat{\theta}_2$. We have

$$\begin{aligned} F(\hat{\theta}_1, \rho_1) - F(\hat{\theta}_2, \rho_2) &= F(\hat{\theta}_1, \rho_1) - F(\hat{\theta}_2, \rho_1) + F(\hat{\theta}_2, \rho_1) - F(\hat{\theta}_2, \rho_2) \\ &\leq -D_\theta(\hat{\theta}_2 - \hat{\theta}_1) + L_F^\rho(\rho_1 - \rho_2), \end{aligned}$$

where L_F^ρ is the Lipschitz constant of F. Since $F(\hat{\theta}_1, \rho_1) - F(\hat{\theta}_2, \rho_2) = \rho_1 - \rho_2$,

$$\hat{\theta}_2 - \hat{\theta}_1 \leq \frac{L_F^\rho - 1}{D_\theta}(\rho_1 - \rho_2). \tag{A.3}$$

Since $F(\theta, \rho)$ is contraction mapping w.r.t. ρ, i.e., $L_F^\rho < 1$. The above inequality leads to contradiction.

Now we focus on the case when $\hat{\theta}_1 > \hat{\theta}_2$, we can get

$$\begin{aligned} F(\hat{\theta}_1, \rho_1) - F(\hat{\theta}_2, \rho_2) &= F(\hat{\theta}_1, \rho_1) - F(\hat{\theta}_2, \rho_1) + F(\hat{\theta}_2, \rho_1) - F(\hat{\theta}_2, \rho_2) \\ &\geq D_\theta(\hat{\theta}_1 - \hat{\theta}_2) + D_\rho(\rho_1 - \rho_2). \end{aligned}$$

Again, we get

$$\hat{\theta}_1 - \hat{\theta}_2 \leq \frac{1 - D_\rho}{D_\theta}(\rho_1 - \rho_2).$$

Following the same argument, when $\rho_1 < \rho_2$, we must have $\hat{\theta}_2 > \hat{\theta}_1$, and moreover:

$$\hat{\theta}_2 - \hat{\theta}_1 \leq \frac{1 - D_\rho}{D_\theta}(\rho_2 - \rho_1).$$

Combining above two cases will complete the proof. □

Armed with the above small deviation result, we now proceed to prove the regret bound in Theorem 2.3.5.

Proof. We will restrict to prove the regret bound for the case $\bar{\lambda} \in (0, 1/2)$, and the proof also holds when $\bar{\lambda} \in [1/2, \infty)$. By the small deviation connection between $\rho_{k,t}$ and $\hat{\theta}_{k,t}$, the following holds

$$P(|\rho_{k,t} - \rho_k^*| \geq \delta) \geq \mathbb{P}\bigg(|\hat{\theta}_{k,t} - \theta_k| \geq \frac{1 - D_\rho}{D_\theta}\delta\bigg).$$

By the convergence analysis of $\rho_{k,t}$, we have the following concentration inequality for the estimator $\hat{\theta}_{k,t}$

$$\mathbb{P}(|\hat{\theta}_{k,t} - \theta_k| \geq \delta) \leq \exp\left(-\frac{D_\theta^2}{2C_1(1 - D_\rho)^2}\delta^2 n_{k,t}^{2\bar{\lambda}}\right),$$

where C_1 is a constant dependent on $\bar{\lambda}$ (defined in the above convergence analysis) and $n_{k,t}$ is the number of pulls of arm k till up to round t. Therefore, for each arm k at time t, we have the following

$$|\hat{\theta}_{k,t} - \theta_k| \leq \sqrt{\frac{\beta \ln t}{n_{k,t}^{2\bar{\lambda}}}},$$

with probability at least $1 - t^{-\beta C'}$, where $C' = \frac{D_\theta^2}{2C_1(1-D_\rho)^2}$. From this, it is immediate to get the following two useful bounds: With probability at least $1 - t^{-\beta C'}$, we have

$$\text{UCB}_{k,t} > \theta_k. \tag{A.4}$$

Furthermore, given $n_{k,t} \geq (4\beta \ln t/\Delta_k^2)^{1/(2\bar{\lambda})}$, where $\Delta_k = \theta^* - \theta_k$, we have

$$\hat{\theta}_{k,t} < \theta_k + \Delta_k/2. \tag{A.5}$$

The above two bounds implies the optimistic property of our constructed UCB algorithm. Particularly, (A.4) implies UCB value should be probably as large as the true arm quality. And (A.5) implies that given enough samples (at least $(4\beta \ln t/\Delta_k^2)^{1/(2\bar{\lambda})}$), then the quality estimator $\hat{\theta}_{k,t}$ would not exceed the true arm quality by more than $\Delta_k/2$. In words, above two bounds can be used to get following guarantee on finding out a suboptimal arm

$$\mathbb{P}(I_t = k | n_{k,t} \geq (4\beta \ln t/\Delta_k^2)^{1/(2\bar{\lambda})}) \leq t^{-2\beta C'}. \tag{A.6}$$

Above property is due to:

$$\begin{aligned}
\text{UCB}_{k,t} = \hat{\theta}_{k,t} + \sqrt{\beta \ln t / n_{k,t}^{2\bar{\lambda}}} &\leq \hat{\theta}_{k,t} + \Delta_k/2 \\
&< \theta_k + \Delta_k/2 + \Delta_k/2 \\
&= \theta^* < \hat{\theta}_{I^*,t} + \sqrt{\beta \ln t / n_{I^*,t}^{2\bar{\lambda}}} \\
&= \text{UCB}_{I^*,t}.
\end{aligned}$$

The first inequality is coming from $n_{k,t} \geq (4\beta \ln t/\Delta_k^2)^{1/(2\bar{\lambda})}$ and second inequality coming from (A.5), the third equality coming from $\Delta_k = \theta^* - \theta_k$ and the forth inequality is due to (A.4).

Till now, we can bound the number of pulls for suboptimal arm k

$$
\begin{aligned}
\mathbb{E}[n_{k,T}] &= 1 + \mathbb{E}[\sum_{t=K}^{T} \mathbb{1}(I_{t+1} = k)] \\
&= 1 + \mathbb{E}[\sum_{t=K}^{T} \mathbb{1}(I_{t+1} = k, n_{k,t} < (4\beta \ln t/\Delta_k^2)^{1/(2\bar{\lambda})})] + \mathbb{E}[\sum_{t=K}^{T} \mathbb{1}(I_{t+1} = k, n_{k,t} \geq (4\beta \ln t/\Delta_k^2)^{1/(2\bar{\lambda})}] \\
&\leq (4\beta \ln T/\Delta_k^2)^{1/(2\bar{\lambda})} + \mathbb{E}[\sum_{t=K}^{T} \mathbb{1}(I_{t+1} = k, n_{k,t} \geq (4\beta \ln t/\Delta_k^2)^{1/(2\bar{\lambda})})] \\
&= (4\beta \ln T/\Delta_k^2)^{1/(2\bar{\lambda})} + \sum_{t=K}^{T} \mathbb{P}(I_{t+1} = k, n_{k,t} \geq (4\beta \ln t/\Delta_k^2)^{1/(2\bar{\lambda})}) \\
&= (4\beta \ln T/\Delta_k^2)^{1/(2\bar{\lambda})} + \sum_{t=K}^{T} \mathbb{P}(I_{t+1} = k | n_{k,t} \geq (4\beta \ln t/\Delta_k^2)^{1/(2\bar{\lambda})}) \mathbb{P}(n_{k,t} \geq (4\beta \ln t/\Delta_k^2)^{1/(2\bar{\lambda})}) \\
&\leq (4\beta \ln T/\Delta_k^2)^{1/(2\bar{\lambda})} + \sum_{t=K}^{T} t^{-2\beta C'} \\
&\leq (\frac{4 \ln T}{C' \Delta_k^2})^{\frac{1}{2\bar{\lambda}}} + \pi^2/6.
\end{aligned}
$$

where the first equality is for adding 1 initial pull for every arm. For the first inequality, suppose the indicator $\mathbb{1}(I_{t+1} = k, n_{k,t} < N)$ takes value 1 at more than $N-1$ time rounds, where $N = (4 \ln t/\Delta_k^2)^{1/(2\bar{\lambda})}$. And let t' be the time step where $\mathbb{1}(I_{t+1} = k, n_{k,t} < N) = 1$ for $(N-1)^{\text{th}}$ round. Thus, including the initial pull, arm k has been pulled at least N rounds until time t'. Then for any $t > t'$, $n_{k,t} > N$ which implies $n_{k,t} > (4 \ln t/\Delta_k^2)^{1/(2\bar{\lambda})}$. Thus, the indication cannot be 1 for any $t > t'$, contradicting the assumption that the indicator takes values 1 for more than $N-1$ rounds. The second inequality is coming from the tail bound for number of pulls for suboptimal to bound the first conditional term. The second probability is bounded by 1. The last inequality is by choosing $\beta = 1/C' = \frac{2C_1(1-D_\rho)^2}{D_\theta^2}$, where C_1 is defined as above in convergence analysis.

We note that the above analysis also holds true when $\bar{\lambda} \in [1/2, \infty)$, and the key difference is that concentration inequality for estimator $\hat{\theta}_{k,t}$ will become following

$$\mathbb{P}(|\hat{\theta}_{k,t} - \theta_k| \geq \delta) \leq \exp\left(-\frac{D_\theta^2}{2C_2(1-D_\rho)^2}\delta^2 n_{k,t}\right),$$

and $\beta = \frac{2C_2(1-D_\rho)^2}{D_\theta^2}$ to ensure the number pulls for suboptimal arms with a logarithmic times.

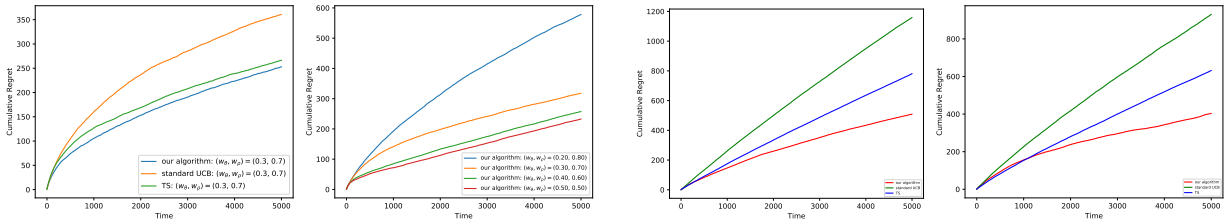
Then the expected regret is obtained by summing for all suboptimal arms: $\mathbb{E}[R(T)] \leq \sum_{k \neq I^*} \mathbb{E}[n_{k,t}]\Delta_k$.

When $\bar{\lambda} \geq 1/2$ (which includes the unbiased feedback setting with $\bar{\lambda} = 1$) dividing the arms into two groups, group 1 contains "almost optimal" arms with $\Delta_k < \sqrt{\ln T/T}$, while group 2 contains "bad" arms with $\Delta_k \geq \sqrt{\ln T/T}$. Then the regret $\mathbb{E}[R(T)] \leq \sum_{k \in \text{Group } 1} \mathbb{E}[n_{k,t}]\Delta_k + \sum_{k \in \text{Group } 2} \mathbb{E}[n_{k,t}]\Delta_k \leq \sqrt{\ln T/T} \sum_{k \in \text{Group } 1} \mathbb{E}[n_{k,t}] + \sum_{k \in \text{Group } 2} (\frac{4 \ln T}{C\Delta_k} + \pi^2/6)\Delta_k \leq T\sqrt{\ln T/T} + 4\sqrt{T \ln T}$, which shows a regret of $\mathcal{O}(\sqrt{T \ln T})$ over T rounds.

When $\bar{\lambda} \to 0$, i.e., $\partial F(\theta, \rho)/\partial \rho \to 1$, which means the information gain on updating estimator $\hat{\theta}_{k,t}$ will become negligible , thus becomes hard to differentiate the arms, which reflects suffering regret in above result. □

A.2.5 Experiments.

We conduct a simple simulation to evaluate our Algorithm Avg-UCB. For each experiment, we perform 50 independent trials up to time $T = 5000$ and report the average cumulative regret. For each independent trial, there are $K = 5$ arms with quality drawn uniformly at random from the unit range $(0, 1)$. We use the classic UCB and TS (Thompson Sampling), the two most popular and robust bandit algorithms, as the comparison baselines. In these baseline algorithms, the learner treats the biased feedback as the unbiased estimates of the



(a) (b) (c) (d)

Figure A.1: (a) & (b): Performance of Algorithm 1 on feedback function defined in (A.7). (a): Performance compared with UCB and TS; (b): Performance on different w_θ. (c) & (d): Performance of Algorithm 1 on feedback function defined in (A.7). (c): $k = 0.7, b = 0.4, \bar{\lambda} = 0.4535$; (d): $k = 0.8, b = 0.4, \bar{\lambda} = 0.5250$.

true rewards. For the UCB algorithm, we set the exploration paramter $\beta = 2$ as the default setting..

Evaluate the performance. We start with evaluating the performance of our algorithm compared with the UCB and TS. We use the feedback function as provided in Example 2.3.1, i.e., $F(\theta, \rho) = w_\theta\theta + w_\rho\rho$, for any $w_\theta, w_\rho \geq 0$ and $w_\theta + w_\rho = 1$. In this function, it is easy to see that $\bar{\lambda} = w_\theta$. Note that when $w_\theta \in [1/2, 1]$, our algorithm will recover the standard UCB algorithm. Thus, we set $w_\theta = 0.3$ and compute $\beta = 1.2$ for Algorithm 1. Figure A.1a, which shows the regret of three algorithms across time, demonstrates that our algorithm does achieve better performance than baseline algorithms that are oblivious of biased feedback.

Evaluate the performance with different $\bar{\lambda}$. In this experiment, we again use the Example 2.3.1 as the user's feedback function. As showed in our regret bound, $\bar{\lambda}$ reflects the learnability of the hidden parameter θ from the noise feedback, i.e., when $\bar{\lambda}$ (recall that in this particular feedback function, $\bar{\lambda} = w_\theta$) is larger, the learner can be more aggressive to learn θ. Thus, in Figure A.1b, we compare the performance of our algorithm on different w_θ, i.e., we set $w_\theta = [0.20, 0.30, 0.40, 0.50]$, where we set β all equal to 1.2. The result shows that the regret get increased as the w_θ increases, this confirms our derived regret bound in

Theorem 2.3.5 of the previous section, where larger $\bar{\lambda}$ (smaller $\bar{\lambda}'$) leads better regrets in the non-asymptotic regime.

Non-convex G. We further investigate how to adapt our algorithm when G is non-convex with respect to ρ, i.e., there exists some $\rho \in (0,1)$ such that $\nabla_\rho^2 G = \nabla_\rho\big(\rho - F(\theta,\rho)\big) < 0$. We construct $F(\theta,\rho)$ from a generalized logistic function. Given the arm's history information ρ_t:

- if the user's private experience is positive, the probability for him to provide positive feedback is $f(\rho) = \frac{1}{1+\exp(-k(\rho-b))}$;
- if the user's private experience is negative, the probability for him to provide positive feedback is $f(1-\rho) = \frac{1}{1+\exp(-k((1-\rho)-b))}$;

Thus, the probability for a user to provide positive feedback is characterized by the following

$$F(\theta,\rho) = \frac{\theta}{1+\exp(-k(\rho-b))} + \frac{1-\theta}{1+\exp(-k((1-\rho)-b))}, \tag{A.7}$$

where k and b are the parameters which control the steepness of the curve and the midpoint of f, they're chosen to ensure the output of $f(\cdot)$ falls in $(0,1)$ for all $\theta \in (0,1)$. Recall that due to the non-convexity of G, we cannot directly apply our algorithm, since $\bar{\lambda}$ is smaller than 0. To adapt our algorithm in non-convex setting, we use the local convexity of equilibrium points ρ^* of $\rho - F(\theta,\rho)$ and we compute

$$\bar{\lambda} = 1 - \sup_{\forall\theta\in(0,1)} \max_{\rho^*\in\mathcal{S}_\theta} \nabla_{\rho^*} F(\theta,\rho).$$

In Figure A.1c, we set $k = 0.7, b = 0.4$, and compute $\bar{\lambda} = 0.4535$, while for Figure A.1d, we set $k = 0.8, b = 0.4$, and compute $\bar{\lambda} = 0.5250$ in order for matching the derived two regions of

our regret bound. By exploiting the local convexity of equilibrium point of function G, the result demonstrates our algorithm is still robust on finding out optimal arm.

A.3 Proofs in Bandits with Beta-Herding Feedback Model

A.3.1 Proof of Lemma 2.4.1

Proof. Let $S_t = \sum_{i=1}^{t} x_i$, where x_i is the realization of the feedback random variable X_i. For simplicity, let $\{n_0\rho_0, n_0(1-\rho_0)\} = \{a, b\}$ (in our current setting, $n_0 = \rho_0 = 0$, which implies $a = b = 0$, but our results hold even if they are nonzero.).

Before we characterize the asymptotic behavior of $\{X_t\}_{t\geq 0}$, we observe that $\{X_t\}_{t\geq 0}$ satisfies a so-called *exchangeable* property. This is summarized in following definition.

Definition A.3.1. *A sequence $\{X_t\}$ of random variables is exchangeable if for all $t \geq 2$*

$$X_1, ..., X_t \stackrel{\Delta}{=} X_{\pi(1)}, ..., X_{\pi(t)}, \forall \pi \in S(t).$$

where $S(t)$ is the symmetric group, the group of permutations.

We have the following

Lemma A.3.1. *Given the above defined learning process, the stochastic random process $\{X_t\}_{t\geq 0}$ is exchangeable.*

Proof. Suppose the principal has received a total of t feedback from the agent, then the probability that there are l positive feedback is given by

$$\frac{\prod_{i=0}^{l-1}(m\theta+a+i)\prod_{j=0}^{t-l-1}(m(1-\theta)b+j)}{\prod_{i=0}^{t-1}(m+a+b+i)}.$$

This shows the exchangeability of $\{X_t\}_{t\geq 0}$. □

Based on the exchangeable property of $\{X_t\}_{t\geq 0}$, we can establish that the asymptotic positive feedback ratio ρ_∞ converges almost surely to a random variable. Suppose $S_t = l$, by the above exchangeability property, we have

$$\mathbb{P}(X_1 = x_1, ..., X_t = x_t) = \frac{\prod_{i=0}^{l-1}(m\theta+a+i)\cdot\prod_{j=0}^{t-1-l}(m(1-\theta)+b+j)}{\prod_{i=0}^{t-1}(m+a+b+i)}.$$

Moreover,

$$\begin{aligned}
\mathbb{P}(S_t = l) &= \binom{t}{l}\frac{\prod_{i=0}^{l-1}(m\theta+a+i)\cdot\prod_{j=0}^{t-1-l}(m(1-\theta)+b+j)}{\prod_{i=0}^{t-1}(m+a+b+i)} \\
&= \binom{t}{l}\frac{\frac{\Gamma(m\theta+a+l)}{\Gamma(m\theta+a)}\cdot\frac{\Gamma(m(1-\theta)+b+t-l)}{\Gamma(m(1-\theta)+b)}}{\frac{\Gamma(m+a+b+t)}{\Gamma(m+a+b)}} \\
&= \frac{\Gamma(m+a+b)}{\Gamma(m\theta+a)\cdot\Gamma(m(1-\theta)+b)}\frac{\Gamma(l+a+m\theta)}{\Gamma(l+1)}\frac{\Gamma(t-l+b+m(1-\theta))}{\Gamma(t-l+1)}\frac{\Gamma(t+1)}{\Gamma(t+a+b+m)} \\
&= \frac{1}{B(m\theta+a, m(1-\theta)+b)}\frac{\Gamma(l+a+m\theta)}{\Gamma(l+1)}\frac{\Gamma(t-l+b+m(1-\theta))}{\Gamma(t-l+1)}\frac{\Gamma(t+1)}{\Gamma(t+a+b+m)}.
\end{aligned}$$

where $\Gamma(\cdot)$ and $B(\cdot)$ are Gamma function and Beta function, respectively.

By Stirling's approximation for gamma function quotient in Lemma A.1.1, we have

$$\mathbb{P}(S_t = l) = \frac{1}{B(m\theta+a, m(1-\theta)+b)}\cdot l^{a+m\theta-1}\cdot(t-l)^{b+m(1-\theta)-1}\cdot t^{1-a-b-m}. \tag{A.8}$$

Denote $l = \rho t$ for some $0 < \rho < 1$. Then we have

$$\mathbb{P}(\frac{S_t}{t} \leq \rho) = \mathbb{P}(\frac{S_t}{t} = 0) + \mathbb{P}(\frac{S_t}{t} = \frac{1}{t}) + ... + \mathbb{P}(\frac{S_t}{t} = \frac{\lfloor t\rho \rfloor}{t}).$$

Therefore,

$$\int_0^\rho \mathbb{P}(\frac{S_t}{t} = u)du = \lim_{t\to\infty} \frac{1}{t}[\mathbb{P}(\frac{S_t}{t} = 0) + \mathbb{P}(\frac{S_t}{t} = \frac{1}{t}) + ... + \mathbb{P}(\frac{S_t}{t} = \frac{\lfloor t\rho \rfloor}{t})]$$
$$\mathbb{P}(\frac{S_t}{t} \leq \rho) = t\int_0^\rho \mathbb{P}(\frac{S_t}{t} = u)du.$$

Replacing l with ρt in Equation (A.8), we have

$$\mathbb{P}(\frac{S_t}{t} \leq \rho) = \frac{1}{B(m\theta + a, m(1-\theta) + b)} \int_0^\rho u^{a+m\theta-1}(1-u)^{b+m(1-\theta)-1}du,$$

which completes the proof. □

A.3.2 Proof of Lemma 2.4.2

Proof. Let $f(x|\theta)$ be the probability mass function of random variable X and x_t be the realization of X_t. Consider the stochastic process specified in Equation (2.1), the probability mass function can be computed as $f(x_t|\theta) = (\frac{m\theta+S_{t-1}+a}{m+a+b+t-1})^{x_t} \cdot (1 - \frac{m\theta+S_{t-1}+a}{m+a+b+t-1})^{1-x_t}$, where $x_t = 1$ or $x_t = 0$, $S_t = \sum_{i=1}^t X_i$. Let $l(x_t|\theta)$ be the log-likelihood of $f(x_t|\theta)$, namely,

$$l(x_t|\theta) = x_t \log\left(\frac{m\theta + S_{t-1} + a}{m + a + b + t - 1}\right) + (1 - x_t)\log\left(1 - \frac{m\theta + S_{t-1} + a}{m + a + b + t - 1}\right).$$

According to the definition of Fisher information for a single observation, and by the chain rule for multiple observations, we have

$$\begin{aligned}\sum_{i=1}^{t}\mathcal{I}_i(\theta) &= \sum_{i=1}^{t}-\mathbb{E}[l^{''}(x_i|\theta)] \\ &= \sum_{i=1}^{t}\frac{m^2}{m+a+b+i-1}(\mathbb{E}[\frac{1}{m\theta+a+S_{i-1}}]+\mathbb{E}[\frac{1}{m(1-\theta)+b+i-1-S_{i-1}}]).\end{aligned}$$

Since $\{X_t\}_{t\geq 1}$ are exchangeable random variables, then

$$\begin{aligned}\mathbb{E}[\frac{1}{m\theta+a+S_t}] &= \sum_{l=0}^{t}\binom{t}{l}\frac{\prod_{i=0}^{l-1}(m\theta+a+i)\cdot\prod_{j=0}^{t-l-1}(m(1-\theta)+b+j)}{\prod_{i=0}^{t-1}(m+a+b+i)}\cdot\frac{1}{m\theta+a+l} \\ &= \sum_{l=0}^{t}\frac{1}{B(m\theta+a,m(1-\theta)+b)}l^{a+m\theta-1}(t-l)^{b+m(1-\theta)-1}t^{1-a-b-m}\cdot\frac{1}{m\theta+a+l} \\ &= \sum_{l=0}^{t}\frac{1}{B(m\theta+a,m(1-\theta)+b)}(l/t)^{a+m\theta-1}(1-l/t)^{b+m(1-\theta)-1}t^{-1}\cdot\frac{1}{m\theta+a+l} \\ &\leq \frac{1}{B(m\theta+a,m(1-\theta)+b)}t^{-1}\sum_{l=0}^{t}(l/t)^{a+m\theta-1}(1-l/t)^{b+m(1-\theta)-1} \\ &= \mathcal{O}(t^{-1}).\end{aligned}$$

Similarly, we also have

$$\mathbb{E}[\frac{1}{m(1-\theta)+b+t-1-S_{t-1}}] = \mathcal{O}(t^{-1}).$$

Thus,

$$\begin{aligned}\lim_{t\to\infty}\sum_{i=1}^{t}\mathcal{I}_i(\theta) &= \lim_{t\to\infty}\sum_{i=1}^{t}\frac{m^2}{m+a+b+i-1}\mathcal{O}(i^{-1})\\ &\leq m^2\lim_{t\to\infty}\sum_{i=1}^{t}\frac{1}{i}\mathcal{O}(i^{-1})\\ &= \mathcal{O}(1),\end{aligned}$$

where $\mathcal{O}(1)$ is a constant. This completes the proof. □

A.3.3 Proof of Theorem 2.4.3

Proof. We prove this by contradiction. Consider a bandit model which has two arms. Without loss of generality, assume arm 1 is optimal and arm 2 is suboptimal, i.e., $\theta_1 > \theta_2$, and suppose there exists an algorithm $\mathcal{A}$ which can achieve sublinear regret, i.e., $\mathbb{E}(R_{\mathcal{A}}(T)) = o(T)$. Let k_t denote the arm chosen by algorithm $\mathcal{A}$ at time t. One must have $\lim_{t\to\infty}\mathbb{P}(k_t = 1) = 1$. Let $\hat{\theta}_1^t, \hat{\theta}_2^t$ be the algorithm's estimators on θ_1, θ_2 given the history information accumulated till time round t. The ability to almost surely choose arm 1 by algorithm $\mathcal{A}$ when $t \to \infty$ indicates that we are able to differentiate the two arms, i.e.,

$$\lim_{t\to\infty}\mathbb{P}(\hat{\theta}_1^t > \hat{\theta}_2^t) = 1$$

However, as shown in Lemma 2.4.2, since the fisher information on the estimator are always bounded even when given infinitely many observations. It implies the estimators are not consistent, and that $\lim_{t\to\infty}\mathbb{P}(\hat{\theta}_1^t < \hat{\theta}_2^t) > 0$. This leads to the contradiction and completes the proof. □

A.3.4 Proof of Theorem 2.4.4

Proof. We prove the upper regret bound by separately bounding the regret of two phases. The proof in the first learning phase is similar to the proof of upper regret bound in avg-herding feedback model. We include the proof here for completeness. By Chernoff Inequality, we have

$$|\hat{\theta}'_{k,t-1} - \theta'_k| < \sqrt{\frac{\beta \ln t}{n_{k,t}}},$$

with probability at least $1 - 2\beta/t^2$. This means, with probability at most $2\beta/t^2$,

$$U_{k,t} < \theta'_k. \tag{A.9}$$

Given that $n_{i,t} \geq 4\beta \ln t/\Delta_k^2$, with probability at least $1 - 2\beta/t^2$,

$$\hat{\theta}'_{k,t} < \theta'_k + \Delta_k/2. \tag{A.10}$$

The above inequality means, the quality estimator would not exceed the true quality by more than $\Delta_k/2$ with high probability. Thus, given a suboptimal arm k which has been pulled for $n_{k,t} > 4\beta \ln t/\Delta_k^2$ times, with probability at most $4\beta/t^2$, we have $U_{I^*,t} < U_{k,t}$, i.e.,

$$\mathbb{P}(I_{t+1} = k | n_{k,t} \geq 4\beta/\Delta_k^2) \leq 4\beta/t^2.$$

The above high probability bound is coming from $U_{k,t} = \hat{\theta}'_{k,t-1} + \sqrt{\beta \ln t/n_{k,t}} \leq \hat{\theta}'_{k,t-1} + \Delta_k/2 < \theta'_k + \Delta_k = \theta^{*\prime} \leq \hat{\theta^*}' + \sqrt{\beta \ln t/n_{k^*,t}} = U_{k^*,t}$ given both (A.9) and (A.10) hold true. Then, one

can bound the expected number of pulls of arm k up to round T^α:

$$\begin{aligned}
\mathbb{E}[n_{k,T^\alpha}] &= 1 + \mathbb{E}[\sum_{t=K}^{T^\alpha} \mathbb{1}(I_{t+1} = k)] \\
&= 1 + \mathbb{E}[\sum_{t=K}^{T^\alpha} \mathbb{1}(I_{t+1} = k, n_{k,t} < 4\beta \ln t/\Delta_k^2)] + \mathbb{E}[\sum_{t=K}^{T^\alpha} \mathbb{1}(I_{t+1} = k, n_{k,t} \geq 4\beta \ln t/\Delta_k^2)] \\
&\leq 4\beta \ln T^\alpha/\Delta_k^2 + \sum_{t=K}^{T^\alpha} \mathbb{P}(I_{t+1} = k, n_{k,t} \geq 4\beta \ln t/\Delta_k^2) \\
&= 4\alpha\beta \ln T/\Delta_k^2 + \sum_{t=K}^{T^\alpha} \mathbb{P}(I_{t+1} = k | n_{k,t} \geq 4\beta \ln t/\Delta_k^2)\mathbb{P}(n_{k,t} \geq 4\beta \ln t/\Delta_k^2) \\
&\leq 4\alpha\beta \ln T/\Delta_k^2 + 8\beta.
\end{aligned}$$

Thus, the regret in the first learning phase could be bounded as follows

$$\mathbb{E}[R(T^\alpha)] = \sum_{k \neq I^*} \mathbb{E}[n_{k,T^\alpha}]\Delta_k \leq \sum_{k \neq I^*} \frac{4\alpha\beta \ln T}{\Delta_k} + 8\beta\Delta_k. \tag{A.11}$$

In the second phase, the algorithm recommends the arm I_τ for the reminder of the rounds. Denote the regret accumulated in the second phase be recommendation regret, i.e., $\mathbb{E}[r_\tau]$. To bound the recommendation regret, we note that the algorithms is essentially running UCB(β) in the first phase and then select the most played arm (MPA) in the second phase. The regret caused in the second phase has been derived by [24] and we rephrase it as follows.

Lemma A.3.2. *If we select most played arm (MPA) in the second phase after adopting UCB(β) in the first phase, for $\beta > 1$, and $\tau \geq K(K+2)$, then*

$$\mathbb{E}[r_\tau] \leq \sqrt{4K\beta \ln \tau/(\tau - K)} + \frac{K}{\beta - 1}(\tau/K - 1)^{2-2\beta}. \tag{A.12}$$

Combining the above two upper regret bounds and summing for all suboptimal arms and all rounds will give us the final regret bound on two-level policy. □

Appendix B

Additional Proofs from Chapter 3

B.1 Lagrangian Formulation

While our setting follows standard bandit settings and aims to maximize the utility, it can be extended to incorporate fairness constraints as commonly seen in the discussion of algorithmic fairness. For example, consider the notion of group fairness, which aims to achieve approximate parity of certain measures across groups. Let $\pi_i(f_i(t)) \in [0,1]$ be the fairness measure for group i (which could reflect the socioeconomic status of the group). One common approach is to impose constraints to avoid the group disparity. Let $\tau \in [0,1]$ be the tolerance parameter, the fairness constraints at t can be written as: $|\pi_i(f_i(t)) - \pi_j(f_j(t))| \leq \tau, \forall i,j \in [K]$. $\pi_i(\cdot)$ is unknown a priori and is dependent on the historical impact. Incorporating the fairness constraints would transform the goal of the institution as a constrained optimization problem:

$$\max_{\mathbf{p}\in\mathcal{P}} \quad \sum_{t=1}^{T} U_t(\mathbf{p}(t)) \quad \text{s.t.} \quad |\pi_i(f_i(t)) - \pi_j(f_j(t))| \leq \tau, \forall i,j \in [K], \forall t \in [T].$$

We can then utilize the Lagrangian relaxation: impose the fairness requirement as soft constraints and obtain an unconstrained optimization problem with a different utility function. As long as we also observe (bandit) feedback on the fairness measures at every time step, the techniques developed in this work can be extended to include fairness constraints.

To simplify the presentation, we fix a time t and drop the dependency on t in the notations.

Definition B.1.1. *The Lagrangian $\mathcal{L} : \mathcal{P} \times \Lambda^2 \to \mathbb{R}$ where $\Lambda \subseteq \mathbb{R}_+^{\binom{K}{2}}$ of our problem can be formulated as:*

$$\mathcal{L}(\mathbf{p}, \lambda) := \sum_{k=1}^{K} p_k r_k(f_k) - \sum_{c=1}^{\binom{K}{2}} \lambda_c^+ \left(\pi_{i_c}(f_{i_c}) - \pi_{j_c}(f_{j_c}) - \tau\right) - \sum_{c=1}^{\binom{K}{2}} \lambda_c^- \left(\pi_{j_c}(f_{j_c}) - \pi_{i_c}(f_{i_c}) - \tau\right),$$

where $\lambda^+, \lambda^- \in \Lambda$. The notation $(i_c, j_c) \in \{(i,j)_{1 \le i < j \le K}\}$ is a pair of combination and $c \in [K(K-1)/2]$ is the index of each pair of this combination.

The problem then reduces to jointly maximize over $\mathbf{p} \in \mathcal{P}$ and minimize over $\lambda^+, \lambda^- \in \Lambda$. Rearranging and with a slight abuse of notations, we have the following equivalent optimization problem:

$$\max_{\mathbf{p} \in \mathcal{P}} \min_{\lambda^+, \lambda^-} \sum_{k=1}^{K} p_k(t) r_k(f_k(t)) + \lambda_k \pi_k(f_k(t)) + \tau \sum_{c=1}^{\binom{K}{2}} (\lambda_c^+ + \lambda_c^-), \tag{B.1}$$

where $\lambda_k := -\sum_{c:i_c=k}(\lambda_c^+ - \lambda_c^-) + \sum_{c:j_c=k}(\lambda_c^+ - \lambda_c^-)$. Due to the uncertainty of reward function $r_k(\cdot)$ and fairness measure $\pi_k(\cdot)$ (recall that our fairness criteria is defined as the parity of socio-economic status cross different groups, which we can only observe the realization drawn from an unknown distribution), we treat the above optimization problem as a hyperparameter optimization: similar to choosing hyperparameters (the Lagrange multipliers: λ^+ and λ^-) based on a validation set in machine learning tasks. Therefore, given a fixed set of λ^+ and

λ^-, the problem in (B.1) can be reduced to the following:

$$\max_{\mathbf{p}\in\mathcal{P}} \sum_{k=1}^{K} p_k(t) \cdot r_k(f_k(t)) + \lambda_k \cdot \pi_k(f_k(t)). \tag{B.2}$$

B.2 Negative Results

In this section, we show that an online algorithm which ignores its action's impact would suffer linear regret. We consider two general bandit algorithms: TS (Thompson Sampling) and a mean-converging family of algorithms (which includes UCB-like algorithms). These are the two most popular and robust bandit algorithms that can be applied to a wide range of scenarios. We prove the negative results respectively. In particular, we construct problem instances that could result in linear regret if the deployed algorithm ignore the action's impact.

Example B.2.1. *Considering the following Bernoulli bandit instance with two arms, indexed by arm 1 and arm 2, i.e., $K = 2$. For any $\epsilon \in [0, 1/2)$, define the expected reward of each arm as follows:*

- *arm 1:* $r_1(p) = p/(1-\epsilon) \cdot \mathbb{1}(p \le 1-\epsilon) + (2-\epsilon-p) \cdot \mathbb{1}(p \ge 1-\epsilon), \quad \forall p \in [0,1]$
- *arm 2:* $r_2(p) = p/(2\epsilon) \cdot \mathbb{1}(p \le \epsilon) + (-\frac{1}{2}p + \frac{1}{2}(1+\epsilon)) \cdot \mathbb{1}(p \ge \epsilon), \quad \forall p \in [0,1]$

It is easy to see that $\mathbf{p}^* = \{1-\epsilon, \epsilon\}$ is the optimal strategy for the above bandit instance.

We first prove the negative result of Thompson Sampling using the above example. The Thompson Sampling algorithm can be summarized as below.

Algorithm 6 Thompson Sampling

$S_i = 0, F_i = 0$.
for $t = 1, 2, \ldots,$ **do**
    For each arm $i = 1, 2$, sample $\theta_i(t)$ from the $\texttt{Beta}(S_i + 1, F_i + 1)$ distribution.
    Play arm $a_t := \arg\max_i \theta_i(t)$ and observe reward $\tilde{r}_t$.
    If $\tilde{r}_t = 1$, then $S_{a_t} = S_{a_t} + 1$, else $F_{a_t} = F_{a_t} + 1$.

Lemma B.2.1. *For the reward structure defined in Example B.2.1, Thompson Sampling would suffer linear regret if it doesn't consider the action's impact it deploys at every time round, namely, it takes the sample mean as the true mean reward of each arm.*

Before we proceed, we first prove the following strong law of large numbers in Beta distribution. We note that the below two lemmas are not new results and can be found in many statistical books.

Lemma B.2.2. *Consider the Beta distribution* $\texttt{Beta}(a\alpha + 1, b\alpha + 1)$ *whose pdf is defined as* $f(x, \alpha) = \frac{[x^a(1-x)^b]^\alpha}{B(a\alpha+1, b\alpha+1)}$, *where* $B(\cdot)$ *is the beta function, then for any positive* (a, b) *such that* $a + b = 1$, *when* $\alpha \to \infty$, *the limit of* $f(x, \alpha)$ *can be characterized by Dirac delta function* $\delta(x - a)$.

Lemma B.2.3. *Let* $h : [0, 1] \to \mathbb{R}^+$ *be any bounded measurable non-negative function with a unique maximum at* x^*, *and suppose* h *is continuous at* x^*. *For* $\lambda > 0$ *define* $h_\lambda(x) = C_\lambda h^\lambda(x)$ *where* C_λ *normalizes such that* $\int_0^1 h_\lambda(x)dx = 1$. *Consider any continuous function* f *defined on* $[0, 1]$ *and* $\epsilon > 0$, *then we have* $\lim_{\lambda\to\infty} \int_{h(x)\le h(x^*)-\epsilon} h_\lambda(x)f(x)dx = 0$ *and* $\lim_{\lambda\to\infty} \int_0^1 h_\lambda(x)f(x)dx = f(x^*)$.

We now ready to prove Lemma B.2.1.

Proof. We prove this by contradiction. Let $\mathrm{Reg}(T)$ denote the expected regret incurred by TS up to time round T, and $N_t(\mathbf{p}) = \sum_{s=1}^t \mathbb{1}(\mathbf{p}(s) = \mathbf{p})$ denote the number of rounds when the

algorithm deploys the (mixed) strategy $\mathbf{p} \in \Delta_K$. Furthermore, let $S_i(t)$(resp. $F_i(t)$) denote the received 1_s(resp. 0_s) of arm i up to time round t. Recall that in Thompson Sampling, we have $\mathbb{P}(a_t = 1) = \mathbb{P}\big(\theta_1(t) > \theta_2(t)\big)$. By the reward function defined in Example B.2.1, it's immediate to see that

$$S_1(T) \geq (1-\epsilon)N_T(\mathbf{p}^*); \quad F_1(T) \leq T - N_T(\mathbf{p}^*); \quad S_2(T) \geq 0.5\epsilon N_T(\mathbf{p}^*); \quad F_2(T) \geq 0.5\epsilon N_T(\mathbf{p}^*).$$

Now suppose Thompson Sampling achieves sublinear regret, i.e., $\mathrm{Reg}(T) = o(T)$, which implies following

$$\lim_{T\to\infty} \frac{T - N_T(\mathbf{p}^*)}{T} = 0.$$

Thus, by the strong law of large numbers and invoking Lemma B.2.2, the sample $\theta_1(T+1) \sim \texttt{Beta}(S_1(T), F_1(T))$ and $\theta_2(T+1) \sim \texttt{Beta}(S_2(T), F_2(T))$ will converge as follows:

$$\lim_{T\to\infty} \theta_1(T+1) = 1; \quad \lim_{T\to\infty} \theta_2(T+1) = 0.5.$$

Then it's almost surely that $\lim_{T\to\infty} \mathbb{P}(a_{T+1} = 1) = \lim_{T\to\infty} \mathbb{P}\big(\theta_1(T+1) > \theta_2(T+1)\big) = 1$. This leads to following holds for sure

$$S_1(s+1) = S_1(s) + 1, \forall s > T.$$

Thus, consider the regret incurred from the $(T+1)-$th round to $(2T)-$th round, the regret will be

$$\mathrm{Reg}(2T) - \mathrm{Reg}(T) = \sum_{s=T+1}^{2T} U(\mathbf{p}(s)) = 0.5T\epsilon,$$

where the second equality follows that $\mathbf{p}(s) = (1, 0)$ holds almost surely from $T + 1$ to $2T$. This shows that $\lim_{T\to\infty} \frac{\mathbb{E}[\mathrm{Reg}(2T)]}{2T} = \epsilon/4$, which contradicts that the algorithm achieves the sublinear regret. □

We now show that a general class of algorithms, which are based on *mean-converging*, will suffer linear regret if it ignores the action's impact. This family of algorithms includes UCB algorithm in classic MAB problems.

Definition B.2.2 (Mean-converging Algorithm [156])**.** *Define* $I_k(t) = \{s : a_s = k, s < t\}$ *as the set of time rounds such the arm* k *is chosen. Let* $\bar{r}_k(t) = \frac{1}{|I_k(t)|} \sum_{s\in I_k(t)} \tilde{r}_s$ *be the empirical mean of arm* k *up to time* t*. The mean-converging algorithm* $\mathcal{A}$ *assigns* $s_k(t)$ *for each arm* k *if following holds true:*

- $s_k(t)$ *is the function of* $\{\tilde{r}_s : s \in I_k(t)\}$ *and time* t*;*
- $\mathbb{P}(s_k(t) = \bar{r}_k(t)) = 1 \quad$ *if* $\quad \liminf_t \frac{|I_k(t)|}{t} > 0.$

Lemma B.2.4. *For the reward structure defined in Example B.2.1, the mean-converging Algorithm will suffer linear regret if it mistakenly take the sample mean as the true mean reward of each arm.*

Proof. We prove above lemma by contradiction. Let $N_t^{\mathcal{A}}(\mathbf{p})$ denote the number of plays with deploying the strategy $\mathbf{p}$ by algorithm $\mathcal{A}$ till time t. Suppose a mean-converging Algorithm $\mathcal{A}$ achieves sublinear regret, then it must have $\lim_{T\to\infty} N_T^{\mathcal{A}}(\mathbf{p}^*)/T > 0$ and $\lim_{T\to\infty} \big(T - N_T^{\mathcal{A}}(\mathbf{p}^*)\big)/T = o(T)$. By the definition of mean-converging algorithm and recall the reward structure defined in Example B.2.1, the score $s_T(1)$ assigned to arm 1 by the algorithm $\mathcal{A}$ must be converging to 1, and the score of $s_T(2)$ assigned to arm 2 must be converging to 0.5. By the strong law of large numbers, it suffices to show that

$\mathbb{P}(\mathbf{p}(t) = \{1, 0\}) = 1, \forall t \geq T + 1$, which implies the algorithm $\mathcal{A}$ would suffer linear regret after T time rounds and thus completes the proof. □

B.3 Missing Proofs for Action-Dependent Bandits

B.3.1 The naive method that directly utilize techniques from Lipschitz bandits

We first give a naive approach which directly applies Lipschitz bandit technique to our action-dependent setting. Recall that each meta arm $\mathbf{p}$ specifies the probability $p_k \in [0, 1]$ for choosing each base arm k. We *uniformly* discretize each p_k into intervals of a fixed length ϵ, with carefully chosen ϵ such that $1/\epsilon$ is an positive integer. Let $\mathcal{P}_\epsilon$ be the space of discretized meta arms, i.e., for each $\mathbf{p} = \{p_1, \ldots, p_K\} \in \mathcal{P}_\epsilon$, $\sum_{k=1}^{K} p_k = 1$ and $p_k \in \{0, \epsilon, 2\epsilon, \ldots, 1\}$ for all k. We then run standard bandit algorithms on the finite set $\mathcal{P}_\epsilon$.

There is a natural trade-off on the choice of ϵ, which controls the complexity of arm space and the discretization error. show that, with appropriately chosen ϵ, this approach can achieve sublinear regret (with respect to the optimal arm in the non-discretized space $\mathcal{P}$).

Lemma B.3.1. *Let $\epsilon = \Theta\big(\big(\frac{\ln T}{T}\big)^{\frac{1}{K+1}}\big)$. Running a bandit algorithm which achieves optimal regret $\mathcal{O}(\sqrt{|\mathcal{P}_\epsilon| T \ln T})$ on the strategy space $\mathcal{P}_\epsilon$ attains the following regret (w.r.t. the optimal arm in non-discretized $\mathcal{P}$):* $\mathrm{Reg}(T) = \mathcal{O}\big(T^{\frac{K}{K+1}} (\ln T)^{\frac{1}{K+1}}\big)$.

Proof. As mentioned, we *uniformly* discretize the interval $[0, 1]$ of each arm into interval of a fixed length ϵ. The strategy space will be reduced as $\mathcal{P}_\epsilon$, which we use this as an approximation for the full set $\mathcal{P}$. Then the original infinite action space will be reduces as finite $\mathcal{P}_\epsilon$, and we run an off-the-shelf MAB algorithm $\mathcal{A}$, such as UCB1 or Successive

Elimination, that only considers these actions in $\mathcal{P}_\epsilon$. Adding more points to $\mathcal{P}_\epsilon$ makes it a better approximation of $\mathcal{P}$, but also increases regret of $\mathcal{A}$ on $\mathcal{P}_\epsilon$. Thus, $\mathcal{P}_\epsilon$ should be chosen so as to optimize this tradeoff. Let $\mathbf{p}_\epsilon^* := \sup_{\mathbf{p}\in\mathcal{P}_\epsilon} \sum_{k=1}^K p_k r_k(p_k)$ denote the best strategy in discretized space $\mathcal{P}_\epsilon$. At each round, the algorithm $\mathcal{A}$ can only hope to approach expected reward $U(\mathbf{p}_\epsilon^*)$, and together with additionally suffering *discretization error*:

$$\texttt{DE}_\epsilon := U(\mathbf{p}^*) - U(\mathbf{p}_\epsilon^*).$$

Then the expected regret of the entire algorithm is:

$$\begin{aligned}
\mathrm{Reg}(T) &= T \cdot U(\mathbf{p}^*) - \texttt{Reward}(\mathcal{A}) \\
&= T \cdot U(\mathbf{p}_\epsilon^*) - \texttt{Reward}(\mathcal{A}) + T(U(\mathbf{p}^*) - U(\mathbf{p}_\epsilon^*)) \\
&= \mathbb{E}[\mathrm{Reg}_\epsilon(T)] + T \cdot \texttt{DE}_\epsilon,
\end{aligned}$$

where $\texttt{Reward}(\mathcal{A})$ is the total reward of the algorithm, and $\mathrm{Reg}_\epsilon(T)$ is the regret relative to $U(\mathbf{p}_\epsilon^*)$. If $\mathcal{A}$ attains optimal regret $\mathcal{O}(\sqrt{KT\ln T})$ on any problem instance with time horizon T and K arms, then,

$$\mathrm{Reg}(T) \leq \mathcal{O}(\sqrt{\left|\mathcal{P}_\epsilon\right| T \ln T}) + T \cdot \texttt{DE}_\epsilon.$$

Thus, we need to choose ϵ to get the optimal trade-off between the size of $\mathcal{P}_\epsilon$ and its discretization error. Recall that $r_k(\cdot)$ is Lipschitz-continuous with the constant of L_k, thus, we could bound the $\texttt{DE}_\epsilon$ by restricting $\mathbf{p}_\epsilon^*$ to be nearest w.r.t $\mathbf{p}^*$. Let $L^* = \max_{k\in[K]}(1 + L_k)$, then it's easy to see that

$$\texttt{DE}_\epsilon = \Omega(KL^*\epsilon).$$

Thus, the total regret can be bounded above from:

$$\mathrm{Reg}(T) \leq \mathcal{O}\left(\sqrt{(1/\epsilon+1)^{K-1}T\ln T}\right) + \Omega(TKL^*\epsilon).$$

By choosing $\epsilon = \Theta\left(\left(\frac{\ln T}{T(L^*)^2}\right)^{\frac{1}{K+1}}\right)$ we obtain:

$$\mathrm{Reg}(T) \leq \mathcal{O}(cT^{\frac{K}{K+1}}(\ln T)^{\frac{1}{K+1}}).$$

where $c = \Theta\left(K(L^*)^{\frac{K-1}{K+1}}\right)$. □

B.3.2 Missing Discussions and Proofs of Theorem 3.4.1

Step 1: Bounding the error of $|\overline{U}(\mathbf{p}) - U(\mathbf{p})|$. For any $\mathbf{p} = \{p_1, \ldots, p_K\}$, define the empirical reward $\overline{U}_t(\mathbf{p}) = \sum_{k=1}^K p_k \bar{r}_t(p_k)$. The first step of our proof is to bound $\mathbb{P}(|\overline{U}_t(\mathbf{p}) - U(\mathbf{p})| \leq \delta)$ for each meta arm $\mathbf{p} = \{p_1, \ldots, p_K\}$ with high probability.[25] Using the Hoeffding's inequality, we obtain

$$\begin{aligned}\mathbb{P}\big(|\overline{U}_t(\mathbf{p}) - U(\mathbf{p})| \geq \delta\big) &= \mathbb{P}\bigg(\bigg|\sum_k \frac{\sum_{s\in\mathcal{T}_t(p_k)}\hat{r}_s(p_k)}{n_t(p_k)} - \sum_k p_k r(p_k)\bigg| \geq \delta\bigg) \\ &\leq 2\exp\bigg(-\frac{2\delta^2}{\sum_k \frac{1}{n_t(p_k)}}\bigg) \leq 2\exp\bigg(-\frac{2\delta^2 n_t(p_{\min}(\mathbf{p}))}{K}\bigg),\end{aligned}$$

where $p_{\min}(\mathbf{p}) := \arg\min_{p_k\in\mathbf{p}} n_t(p_k)$. By choosing $\delta = \sqrt{\frac{K\ln t}{n_t(p_{\min}(\mathbf{p}))}}$ in the above inequality, for each meta arm $\mathbf{p}$ at time t, we have that $|\overline{U}_t(\mathbf{p}) - U(\mathbf{p})| \leq \sqrt{K\ln t / n_t(p_{\min}(\mathbf{p}))}$, with the probability at least $1 - 2/t^2$.

Step 2: Bounding the probability on deploying suboptimal meta arm. With the above high probability bound we obtain in Step 1, we can construct an `UCB` index for each

[25] We use δ to denote the estimation error, as ϵ has been used as the discretization parameter.

meta arm $\mathbf{p} \in \mathcal{P}_\epsilon$:

$$\texttt{UCB}_t(\mathbf{p}) = \overline{U}_t(\mathbf{p}) + \sqrt{\frac{K \ln t}{n_t(p_{\min}(\mathbf{p}))}}. \tag{B.3}$$

The above constructed UCB index gives the following guarantee:

Lemma B.3.2. *At any time round t, for a suboptimal meta arm $\mathbf{p}$, if it satisfies $n_t(p_{\min}(\mathbf{p})) \geq 4K \ln t/\Delta_{\mathbf{p}}^2$, then $\texttt{UCB}_t(\mathbf{p}) < \texttt{UCB}_t(\mathbf{p}_\epsilon^*)$ with the probability at least $1 - 4/t^2$. Thus, for any t,*

$$\mathbb{P}\left(\mathbf{p}(t) = \mathbf{p} | n_t(p_{\min}(\mathbf{p})) \geq 4K \ln t/\Delta_{\mathbf{p}}^2\right) \leq 4t^{-2},$$

where $\Delta_{\mathbf{p}}$ denotes the badness of meta arm $\mathbf{p}$.

Proof. We prove this lemma by considering two "events" which occur with high probability: (1) the UCB index of each meta arm will concentrate on the true mean utility of $\mathbf{p}$; (2) the empirical mean utility of each meta arm $\mathbf{p}$ will also concentrate on the true mean utility of $\mathbf{p}$. We then show that the probability of either one of the events not holding is at most $4/t^2$. By a union bound we prove above desired lemma.

$$\begin{aligned}
\texttt{UCB}_t(\mathbf{p}) &= \sum_{k=1}^{K} p_k \bar{r}_t(p_k) + \sqrt{\ln t \frac{K}{n_t(p_{\min}(\mathbf{p}))}} \\
&\overset{\text{(a)}}{\leq} \sum_{k=1}^{K} p_k \bar{r}_t(p_k) + \Delta_{\mathbf{p}}/2 < \left(\sum_{k=1}^{K} p_k r_k(p_k) + \Delta_{\mathbf{p}}/2\right) + \Delta_{\mathbf{p}}/2 && \text{By Event 1} \\
&= \sum_{k=1}^{K} p_{k,\epsilon}^* r_k(p_{k,\epsilon}^*) < \sum_{k=1}^{K} p_{k,\epsilon}^* \bar{r}_t(p_{k,\epsilon}^*) + \sqrt{\ln t \frac{K}{n_t(p_{\min}(\mathbf{p}_\epsilon^*))}} && \text{By Event 2} \\
&= \texttt{UCB}_t(\mathbf{p}_\epsilon^*),
\end{aligned}$$

where $\mathbf{p}_\epsilon^* = (p_{1,\epsilon}^*, \ldots, p_{K,\epsilon}^*)$. The first inequality (a) comes from that $n_t(p_{\min}(\mathbf{p})) \geq \frac{4K \ln t}{\Delta_{\mathbf{p}}^2}$ and the probability of third inequality or fifth inequality not holding is at most $4/t^2$. □

Intuitively, Lemma B.3.2 essentially shows that for a meta arm $\mathbf{p}$, if its $n_t(p_{\min}(\mathbf{p}))$ is sufficiently sampled with respect to $\Delta_{\mathbf{p}}$, that is, sampled at least $4K \ln t/\Delta_{\mathbf{p}}^2$ times, we know that the probability that we hit this suboptimal meta arm is very small.

Step 3: Bounding the $\mathbb{E}[n_T(p_{\min}(\mathbf{p}))]$. Ideally, we would like to bound the number of the selections on deploying the suboptimal meta arm, i.e., $N_T(\mathbf{p})$, in a logarithmic order of T. However, if we proceed to bound this by separately considering each meta arm, the final regret bound will have an order with exponent in K since the number of meta arms grows exponentially in K. Instead, we turn to bound $\mathbb{E}[n_T(p_{\min}(\mathbf{p}))]$. Recall that by the definitions of $n_T(p)$ and $p_{\min}(\mathbf{p})$, the pulls of $\mathbf{p}$ is upper bounded by its $n_T(p_{\min}(\mathbf{p}))$. This quantity will help us to reduce the exponential K to the polynomial K. This is formalized in the following lemma.

Lemma B.3.3. *For each suboptimal meta arm $\mathbf{p} \neq \mathbf{p}_\epsilon^*$, we have that $\mathbb{E}[n_T(p_{\min}(\mathbf{p}))] \leq \frac{4K \ln T}{\Delta_{\mathbf{p}}^2} + \mathcal{O}(1)$.*

Proof. To simplify notations, for each discretized arm p_k, we define the notion of *super set* $\mathcal{S}(p_k) = \{\mathbf{p} : p_k \in \mathbf{p}\}$ which contains all the meta arms that include this discretized arm. For

suboptimal meta arm $\mathbf{p} \neq \mathbf{p}^*_\epsilon$ and its $p_{\min}(\mathbf{p})$, we have

$$
\begin{aligned}
&\mathbb{E}[n_T(p_{\min}(\mathbf{p}))] \\
&\overset{(a)}{=} 1 + \mathbb{E}\Bigg[\sum_{t=\lceil K/\epsilon\rceil+1}^{T} \mathbb{1}\left(\mathbf{p}(t) = \mathbf{p}, \mathbf{p} \in \mathcal{S}(p_{\min}(\mathbf{p}))\right)\Bigg] \\
&= 1 + \mathbb{E}\Bigg[\sum_{t=\lceil K/\epsilon\rceil+1}^{T} \mathbb{1}\left(\mathbf{p}(t) = \mathbf{p}, \mathbf{p} \in \mathcal{S}(p_{\min}(\mathbf{p})); n_t(p_{\min}(\mathbf{p})) < \frac{4K\ln t}{\Delta_{\mathbf{p}}^2}\right)\Bigg] \\
&\quad + \mathbb{E}\Bigg[\sum_{t=\lceil K/\epsilon\rceil+1}^{T} \mathbb{1}\left(\mathbf{p}(t) = \mathbf{p}, \mathbf{p} \in \mathcal{S}(p_{\min}(\mathbf{p})); n_t(p_{\min}(\mathbf{p})) \geq \frac{4K\ln t}{\Delta_{\mathbf{p}}^2}\right)\Bigg] \\
&\overset{(b)}{\leq} \frac{4K\ln T}{\Delta_{\mathbf{p}}^2} + \mathbb{E}\Bigg[\sum_{t=\lceil K/\epsilon\rceil+1}^{T} \mathbb{1}\left(\mathbf{p}(t) = \mathbf{p}, \mathbf{p} \in \mathcal{S}(p_{\min}(\mathbf{p})); n_t(p_{\min}(\mathbf{p})) \geq \frac{4K\ln t}{\Delta_{\mathbf{p}}^2}\right)\Bigg] \\
&= \frac{4K\ln T}{\Delta_{\mathbf{p}}^2} + \sum_{t=\lceil K/\epsilon\rceil+1}^{T} \mathbb{P}\left(\mathbf{p}(t) = \mathbf{p}, \mathbf{p} \in \mathcal{S}(p_{\min}(\mathbf{p})); n_t(p_{\min}(\mathbf{p})) \geq \frac{4K\ln t}{\Delta_{\mathbf{p}}^2}\right) \\
&= \frac{4K\ln T}{\Delta_{\mathbf{p}}^2} + \sum_{t=\lceil K/\epsilon\rceil+1}^{T} \mathbb{P}\left(\mathbf{p}(t) = \mathbf{p}, \mathbf{p} \in \mathcal{S}(p_{\min}(\mathbf{p}))\middle| n_t(p_{\min}(\mathbf{p})) \geq \frac{4K\ln t}{\Delta_{\mathbf{p}}^2}\right)\mathbb{P}\left(n_t(p_{\min}(\mathbf{p})) \geq \frac{4K\ln t}{\Delta_{\mathbf{p}}^2}\right) \\
&\overset{(c)}{\leq} \frac{4K\ln T}{\Delta_{\mathbf{p}}^2} + \frac{2\pi^2}{3}.
\end{aligned}
$$

We add 1 in the first equality to account for 1 (step (a)) initial pull of every discretized arm by the algorithm (the initialization phase). In step (b), suppose for contradiction that the indicator $\mathbb{1}\left(\mathbf{p}(t) = \mathbf{p}, \mathbf{p} \in \mathcal{S}(p_{\min}(\mathbf{p})); n_t(p_{\min}(\mathbf{p})) < S\right)$ takes value of 1 at more than $S-1$ time steps, where $S = \frac{4K\ln T}{\Delta_{\mathbf{p}}^2}$. Let τ be the time step at which this indicator is 1 for the $(S-1)$-th time. Then the number of pulls of all meta arms in $\mathcal{S}(p_{\min}(\mathbf{p}))$ is at least L times until time τ (including the initial pull), and for all $t \geq \tau$, $n_t(p_{\min}(\mathbf{p})) \geq S$ which implies $n_t(p_{\min}(\mathbf{p})) \geq \frac{4K\ln t}{\Delta_{\mathbf{p}}^2}$. Thus, the indicator cannot be 1 for any $t \geq \tau$, contradicting the assumption that the indicator takes value of 1 more than L times. This bounds $1 + \mathbb{E}\left[\sum_{t\geq\lceil K/\epsilon\rceil+1} \mathbb{1}\left(\mathbf{p}(t) = \mathbf{p}, \mathbf{p} \in \mathcal{S}(p_{\min}(\mathbf{p})); n_t(p_{\min}(\mathbf{p})) < S\right)\right]$ by S. In step (c),

we apply the lemma B.3.2 to bound the first conditional probability term and use the fact that the probabilities cannot exceed 1 to bound the second probability term. □

We use this connection in the following step to reduce the computation of regret on pulling all suboptimal meta arms so that to calculate the regret via the summation over discretized arms.

Wrapping up: Proof of Theorem 3.4.1. We are now ready to prove Theorem 3.4.1. We first define notations that are helpful for our analysis. To circumvent the summation over all feasible suboptimal arms $\{\mathbf{p}\}$, for each discretized arm p_k, we define the notion of *super set* $\mathcal{S}(p_k) := \{\mathbf{p} : p_k \in \mathbf{p}\}$ which contains all suboptimal meta arms that include this discretized arm. With a slight abuse of notations, we also sort all meta arms in $\mathcal{S}(p_k)$ as $\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_{I(p_k)}$ in ascending order of their expected rewards, where $I(p_k) := |\mathcal{S}(p_k)|$ is the cardinality of the super set $\mathcal{S}(p_k)$. For $\mathbf{p}_l \in \mathcal{S}(p_k)$, we also define $\Delta_l^{p_k} := \Delta_{\mathbf{p}_l}$ where $l \in [I(p_k)]$, and specifically $\Delta_{\min}^{p_k} := \min_{\mathbf{p}\in\mathcal{S}(p_k)} \Delta_{\mathbf{p}} = \Delta_{I(p_k)}^{p_k}$; $\Delta_{\max}^{p_k} := \max_{\mathbf{p}\in\mathcal{S}(p_k)} \Delta_{\mathbf{p}} = \Delta_1^{p_k}$. Let $\mathrm{Reg}_\epsilon(T)$ denote the regret relative to the best strategy in the discretized space parameterized by ϵ. With these notations, we first establish the following instance-dependent regret.

Lemma B.3.4. *Following the* `UCB` *designed in (B.3), we have the following instance-dependent regret on the discretized arm space:* $\mathrm{Reg}_\epsilon(T) \le \lceil K/\epsilon \rceil \cdot (\Delta_{\max} + \mathcal{O}(1)) + \sum_{p_k:\Delta_{\min}^{p_k}>0} 8K \ln T/\Delta_{\min}^{p_k}$, *where* $\Delta_{\max} := \max_{p_k} \Delta_{\max}^{p_k}$.

Proof. Note that by definition, we can compute the regret $\mathrm{Reg}_\epsilon(T)$ as follows:

$$\mathrm{Reg}_\epsilon(T) = \sum_{\mathbf{p}\in\mathcal{P}_\epsilon} \mathbb{E}[N_T(\mathbf{p})]\Delta_{\mathbf{p}} \le \sum_{p_k} \sum_{l\in[I(p_k)]} \mathbb{E}[N_T(\mathbf{p}_l)]\Delta_l^{p_k}. \tag{B.4}$$

Observe that, by Lemma B.3.3, for each discretized arm p_k, there are two possible cases:

- There exists a meta arm $\mathbf{p}_l \in \mathcal{S}(p_k)$, and its $p_{\min}(\mathbf{p}_l) = p_k$. Then by linearity of expectation, we can bound the expectation of total number of pulls for all $\mathbf{p}_{l'} \in \mathcal{S}(p_k)$ as follows

$$\sum_{\mathbf{p}_{l'} \in \mathcal{S}(p_k)} \mathbb{E}[N_T(\mathbf{p}_{l'})] = \mathbb{E}[n_T(p_k)] \leq \frac{4K \ln T}{(\Delta_{\min}^{p_k})^2} + \mathcal{O}(1).$$

- There exists no meta arm $\mathbf{p} \in \mathcal{S}(p_k)$, and $p_{\min}(\mathbf{p})$ for each $\mathbf{p}$ is p_k. In this case, for each $\mathbf{p}_l \in \mathcal{S}(p_k)$, there always exists another discretized arm p' that is included in $\mathbf{p}_l$ such that $p' = p_{\min}(\mathbf{p}_l)$ but $p' \neq p_k$. Thus, for each $\mathbf{p}_l \in \mathcal{S}(p_k)$, together with other meta arms which also include discretized arm p' as $\mathbf{p}_l$, we have that

$$\begin{aligned} \sum_{\mathbf{p} \in \bigcup_{p' \in \mathbf{p}} \mathbf{p}} \mathbb{E}[N_T(\mathbf{p})] &= \sum_{\mathbf{p} \in \mathcal{S}(p')} \mathbb{E}[N_T(\mathbf{p})] \\ = \quad \mathbb{E}[n_T(p')] &\leq \frac{4K \ln T}{(\Delta_{\min}^{p'})^2} + \mathcal{O}(1). \end{aligned}$$

The above observations imply that even though we can not find any meta arm $\mathbf{p}$ in $\mathcal{S}(p_k)$ such that $p_{\min}(\mathbf{p}) = p_k$, we can always carry out similar analysis by finding another discretized arm $p' \in \mathbf{p}$ but $p' \neq p_k$, such that $p' = p_{\min}(\mathbf{p})$. Thus, for each discretized arm p_k, we can focus on the case where p_k is able to attain the minimum $n_t(p_k)$ for some $\mathbf{p} \in \mathcal{S}(p_k)$. For analysis convenience, instead of looking at the counter of $\mathbf{p}$, i.e., $n_t(p_{\min}(\mathbf{p}))$, we will define a counter $c(p_k)$ for each discretized arm p_k and the value of $c(p_k)$ at time t is denoted by $c_t(p_k)$. The update of $c_t(p_k)$ is as follows: For a round $t > \lceil K/\epsilon \rceil$ (here $\lceil K/\epsilon \rceil$ is the number of rounds needed for initialization), let $\mathbf{p}(t)$ be the meta arm selected in round t by the algorithm. Let $p_k = \arg\min_{p_k \in \mathbf{p}(t)} c_{t-1}(p_k)$. We increment $c(p_k)$ by one, i.e., $c_t(p_k) = c_{t-1}(p_k) + 1$. In other words, we find the discretized arm p_k with the smallest counter in $\mathbf{p}(t)$ and increment its counter. If such p_k is not unique, we pick an arbitrary discretized arm with the smallest counter. Note that the initialization gives $\sum_{p_k} c_{\lceil K/\epsilon \rceil}(p_k) = \lceil K/\epsilon \rceil$. It is easy to see that for any $p_k = p_{\min}(\mathbf{p})$, we have $n_t(p_k) = c_t(p_k)$.

With the above change of counters, Lemma B.3.2 and Lemma B.3.3 then have the implication on selecting discretized arm $p_k \notin \mathbf{p}_\epsilon^*$ given its counter $c_t(p_k)$. To see this, for each $\mathbf{p}_l \in \mathcal{S}(p_k)$, we define sufficient selection of discretized arm p_k with respect to $\mathbf{p}_l$ as p_k being selected $4K \ln T / \left(\Delta_l^{p_k}\right)^2$ times and p_k's counter $c(p_k)$ being incremented in these selected instances. Then Lemma B.3.2 tells us when p_k is sufficiently selected with respect to $\mathbf{p}_l$, the probability that the meta arm $\mathbf{p}_l$ is selected by the algorithm is very small. On the other hand, when p_k's counter $c(p_k)$ is incremented, but if p_k is under-selected with respect to $\mathbf{p}_l$, we incur a regret of at most $\Delta_j^{p_k}$ for some $j \leq l$.

Define $C_T(\Delta) := \frac{4K \ln T}{\Delta^2}$, the number of selection that is considered sufficient for a meta arm with reward Δ away from the optimal strategy $\mathbf{p}_\epsilon^*$ with respect to time horizon t. With the above analysis, we define following two situations for the counter of each discretizad arm:

$$
\begin{aligned}
c_T^{l,\texttt{suf}}(p_k) &:= \sum_{t=\lceil K/\epsilon \rceil + 1}^{T} \mathbb{1}\left(\mathbf{p}(t) = \mathbf{p}_l, c_t(p_k) > c_{t-1}(p_k) > C_T(\Delta_l^{p_k})\right), \\
c_T^{l,\texttt{und}}(p_k) &:= \sum_{t=\lceil K/\epsilon \rceil + 1}^{T} \mathbb{1}\left(\mathbf{p}(t) = \mathbf{p}_l, c_t(p_k) > c_{t-1}(p_k), c_{t-1}(p_k) \leq C_T(\Delta_l^{p_k})\right).
\end{aligned}
$$

Clearly, we have $c_T(p_k) = 1 + \sum_{l \in I(p_k)} \left(c_T^{l,\texttt{suf}}(p_k) + c_T^{l,\texttt{und}}(p_k)\right)$. With these notations, we can write (B.4) as follows:

$$
\mathrm{Reg}_\epsilon(T) \leq \mathbb{E}\left[\sum_{p_k} \left(\Delta_{\max}^{p_k} + \sum_{l \in [I(p_k)]} \left(c_T^{l,\texttt{suf}}(p_k) + c_T^{l,\texttt{und}}(p_k)\right) \cdot \Delta_l^{p_k}\right)\right]. \tag{B.5}
$$

The proof of this lemma will complete after establishing following two claims:

$$\text{Claim 1: } \mathbb{E}\Bigg[\sum_{p_k}\sum_{l\in[I(p_k)]} c_T^{l,\texttt{suf}}(p_k)\Bigg] \leq \lceil K/\epsilon\rceil \cdot \mathcal{O}(1). \tag{B.6}$$

$$\text{Claim 2: } \mathbb{E}\Bigg[\sum_{p_k}\sum_{l\in[I(p_k)]} c_T^{l,\texttt{und}}(p_k)\Delta_l^{p_k}\Bigg] \leq \sum_{p_k}\left((4K\ln T)/\Delta_{\min}^{p_k} + 4K\ln T\,(1/\Delta_{\min}^{p_k} - 1/\Delta_{\max}^{p_k})\right). \tag{B.7}$$

We now first prove the Claim 1 as in (B.6), i.e., for any $t > \lceil K/\epsilon\rceil$, we have following upper bound over counters of sufficiently selected discretized arms. To see this, by definition of $c_T^{l,\texttt{suf}}(p_k)$, it reduces to show that for any $T \geq t > \lceil K/\epsilon\rceil$,

$$\begin{aligned}
&\mathbb{E}\Bigg[\sum_{p_k}\sum_{l\in[I(p_k)]} \mathbb{1}\left(\mathbf{p}(t) = \mathbf{p}_l, c_t(p_k) > c_{t-1}(p_k) > C_T(\Delta_l^{p_k})\right)\Bigg] \\
=& \sum_{p_k}\sum_{l\in[I(p_k)]} \mathbb{P}\left(\mathbf{p}(t) = \mathbf{p}_l, p_k = p_{\min}(\mathbf{p}_l); \forall p \in \mathbf{p}_l, c_{t-1}(p) > C_T(\Delta_l^{p_k})\right) \\
\overset{\text{(a)}}{\leq}& \lceil 4K/\epsilon\rceil \cdot t^{-2},
\end{aligned}$$

where the last step (a) is due to Lemma B.3.2, thus (B.6) follows from a simple series bound.

We now proceed to analyze the discretized arms that are not sufficiently included in the meta arm chosen by the algorithm and prove the Claim 2 as in (B.7). For any under-selected discretized arm p_k, its counter $c(p_k)$ will increase from 1 to $C_T(\Delta_{\min}^{p_k})$. To simplify the notation, we set $C_T(\Delta_0^{p_k}) = 0$. Suppose that at round t, $c(p_k)$ is incremented, and $c_{t-1}(p_k) \in (C_T(\Delta_{j-1}^{p_k}), C_T(\Delta_j^{p_k})]$ for some $j \in [I(p_k)]$. Notice that we are only interested in the case that p_k is under-selected. In particular, if this is indeed the case, $\mathbf{p}(t) = \mathbf{p}_l$ for some $l \geq j$. (Otherwise, $\mathbf{p}(t)$ is sufficiently selected based on the counter value $c_{t-1}(p_k)$.) Thus, we will suffer a regret of $\Delta_l^{p_k} \leq \Delta_j^{p_k}$ (step (a)). As a result, for counter $c_t(p_k) \in (C_T(\Delta_{j-1}^{p_k}), C_T(\Delta_j^{p_k})]$, we will suffer a total regret for those playing suboptimal meta arms that include under-selected

discretized arms at most $(C_T(\Delta_j^{p_k}) - C_T(\Delta_{j-1}^{p_k})) \cdot \Delta_j^{p_k}$ in rounds that $c_t(p_k)$ is incremented (step (b)). In what follows we establish the above analysis rigorously.

$$
\begin{aligned}
&\sum_{l \in [I(p_k)]} c_T^{l,\mathtt{und}}(p_k) \Delta_l^{p_k} \\
&= \sum_{t=\lceil K/\epsilon \rceil + 1}^{T} \sum_{l \in [I(p_k)]} \mathbb{1}\left(\mathbf{p}(t) = \mathbf{p}_l, c_t(p_k) > c_{t-1}(p_k), c_{t-1}(p_k) \leq C_T(\Delta_l^{p_k})\right) \cdot \Delta_l^{p_k} \\
&= \sum_{t=\lceil K/\epsilon \rceil + 1}^{T} \sum_{l \in [I(p_k)]} \sum_{j=1}^{l} \mathbb{1}\left(\mathbf{p}(t) = \mathbf{p}_l, c_t(p_k) > c_{t-1}(p_k), c_{t-1}(p_k) \in (C_T(\Delta_{j-1}^{p_k}), C_T(\Delta_j^{p_k})]\right) \cdot \Delta_l^{p_k} \\
&\overset{(a)}{\leq} \sum_{t=\lceil K/\epsilon \rceil + 1}^{T} \sum_{l \in [I(p_k)]} \sum_{j=1}^{l} \mathbb{1}\left(\mathbf{p}(t) = \mathbf{p}_l, c_t(p_k) > c_{t-1}(p_k), c_{t-1}(p_k) \in (C_T(\Delta_{j-1}^{p_k}), C_T(\Delta_j^{p_k})]\right) \cdot \Delta_j^{p_k} \\
&\leq \sum_{t=\lceil K/\epsilon \rceil + 1}^{T} \sum_{l,j \in [I(p_k)]} \mathbb{1}\left(\mathbf{p}(t) = \mathbf{p}_l, c_t(p_k) > c_{t-1}(p_k), c_{t-1}(p_k) \in (C_T(\Delta_{j-1}^{p_k}), C_T(\Delta_j^{p_k})]\right) \cdot \Delta_j^{p_k} \\
&= \sum_{t=\lceil K/\epsilon \rceil + 1}^{T} \sum_{j \in [I(p_k)]} \mathbb{1}\left(\mathbf{p}(t) \in \mathcal{S}(p_k), c_t(p_k) > c_{t-1}(p_k), c_{t-1}(p_k) \in (C_T(\Delta_{j-1}^{p_k}), C_T(\Delta_j^{p_k})]\right) \cdot \Delta_j^{p_k} \\
&\overset{(b)}{\leq} \sum_{j \in [I(p_k)]} (C_T(\Delta_j^{p_k}) - C_T(\Delta_{j-1}^{p_k})) \cdot \Delta_j^{p_k}.
\end{aligned}
$$

Now, we can compute the regret incurred by selecting the meta arm which includes under-selected discretized arms:

$$
\begin{aligned}
\sum_{p_k} \sum_{l \in [I(p_k)]} c_T^{l,\mathtt{und}}(p_k) \Delta_l^{p_k} &\leq \sum_{p_k} \sum_{j \in [I(p_k)]} (C_T(\Delta_j^{p_k}) - C_T(\Delta_{j-1}^{p_k})) \cdot \Delta_j^{p_k} \\
&= \sum_{p_k} \left(C_T(\Delta_{\min}^{p_k}) \Delta_{\min}^{p_k} + \sum_{j \in [I(p_k)-1]} C_T(\Delta_j^{p_k}) \cdot (\Delta_j^{p_k} - \Delta_{j+1}^{p_k}) \right) \\
&\leq \sum_{p_k} \left(C_T(\Delta_{\min}^{p_k}) \Delta_{\min}^{p_k} + \int_{\Delta_{\min}^{p_k}}^{\Delta_{\max}^{p_k}} C_t(x) dx \right) \\
&= \sum_{p_k} \left(\frac{4K \ln T}{\Delta_{\min}^{p_k}} + 4K \ln T \left(\frac{1}{\Delta_{\min}^{p_k}} - \frac{1}{\Delta_{\max}^{p_k}} \right) \right). \qquad \text{(B.8)}
\end{aligned}
$$

Equipped with the above set of results, the bound of regret (B.5) follows by combing the bounds in (B.6) and (B.7). □

To achieve instance-independent regret bound, we need to deal with the case when the meta-arm gap $\Delta_{\min}^{p_k}$ is too small, leading the regret to approach infinite. Nevertheless, one can still show that when $\Delta_{\min}^{p_k} \leq 1/\sqrt{T}$, the regret contributed by this scenario scales at most $\mathcal{O}(\sqrt{T})$ at time horizon T.

Lemma B.3.5. *Following the* `UCB` *designed in (B.3), we have:* $\mathrm{Reg}_\epsilon(T) \leq \mathcal{O}\big(K\sqrt{T\ln T/\epsilon}\big)$.

Proof. Following the proof of Lemma B.3.4, we only need to consider the meta arms that are played when they are under-sampled. We particularly need to deal with the situation when $\Delta_{\min}^{p_k}$ is too small. We measure the threshold for $\Delta_{\min}^{p_k}$ based on $c_T(p_k)$, i.e., the counter of disretized arm p_k at time horizon T. Let $\{T(p_k), \forall p_k\}$ be a set of possible counter values at time horizon T. Our analysis will then be conditioned on the event that $\mathcal{E}(p_k) = \{c_T(p_k) = T(p_k)\}$. By definition,

$$\begin{aligned}
&\mathbb{E}\big[\sum_{l\in[I(p_k)]} c_T^{l,\texttt{und}}(p_k)\cdot\Delta_l^{p_k} \mid \mathcal{E}(p_k)\big] \\
=&\sum_{t=\lceil K/\epsilon\rceil+1}^{T}\sum_{l\in[I(p_k)]} \mathbb{1}\left(\mathbf{p}(t)=\mathbf{p}_l, c_t(p_k)>c_{t-1}(p_k), c_{t-1}(p_k)\leq C_T(\Delta_l^{p_k}) \mid \mathcal{E}(p_k)\right)\cdot\Delta_l^{p_k}. \quad \text{(B.9)}
\end{aligned}$$

We define $\Delta^*(T(p_k)) := \left(\frac{4K\ln T}{T(p_k)}\right)^{1/2}$, i.e., $C_T(\Delta^*(T(p_k))) = T(p_k)$. To achieve *instance-independent* regret bound, we consider following two cases:

Case 1: $\Delta_{\min}^{p_k} > \Delta^*(T(p_k))$, we thus have

$$\mathbb{E}\big[\sum_{l\in[I(p_k)]} c_T^{l,\texttt{und}}(p_k)\cdot\Delta_l^{p_k} \mid \mathcal{E}(p_k)\big] \leq \mathcal{O}\left(\sqrt{4K\ln T\cdot T(p_k)}\right). \quad \text{(B.10)}$$

Case 2: $\Delta_{\min}^{p_k} < \Delta^*(T(p_k))$. Let $l^* := \min\{l \in I(p_k) : \Delta_l^{p_k} > \Delta^*(T(p_k))\}$. Observe that we have $\Delta_{l^*}^{p_k} \leq \Delta^*(T(p_k))$ and the counter $c(p_k)$ never go beyond $T(p_k)$, we thus have

$$\begin{aligned}(\text{B.9}) &\leq (C_T(\Delta^*(T(p_k))) - C_T(\Delta_{l^*-1}^{p_k})) \cdot \Delta^*(T(p_k)) + \sum_{j\in[l^*-1]} (C_T(\Delta_j^{p_k}) - C_T(\Delta_{j-1}^{p_k})) \cdot \Delta_j^{p_k} \\ &\leq C_T(\Delta^*(T(p_k))) \cdot \Delta^*(T(p_k)) + \int_{\Delta^*(T(p_k))}^{\Delta_{\max}^{p_k}} C_T(x)dx \leq \mathcal{O}\left(\sqrt{K \ln T \cdot T(p_k)}\right). \quad (\text{B.11})\end{aligned}$$

Thus, combining (B.10) and (B.11), we have

$$\begin{aligned}\mathbb{E}\big[\sum_{p_k:\Delta_{\min}^{p_k}>0} \sum_{l\in[I(p_k)]} c_T^{l,\texttt{und}}(p_k) \cdot \Delta_l^{p_k} \mid \mathcal{E}(p_k)\big] &\leq \sum_{p_k:\Delta_{\min}^{p_k}>0} \mathcal{O}(\sqrt{K \ln T \cdot T(p_k)}) \\ &\overset{\text{(a)}}{\leq} \mathcal{O}(K\sqrt{T \ln T/\epsilon}),\end{aligned}$$

where (a) is by Jesen's inequality and $\sum_{p_k} T(p_k) \leq KT/\epsilon$. Put all pieces together, we have the instance-independent regret bound as stated in the lemma. Observe that the final inequality does not depend on the event $\mathcal{E}(p_k)$, we thus can drop this conditional expectation. □

With the above lemma in hand, picking $\epsilon = \Theta((\ln T/T)^{1/3})$ will give us desired result in Theorem 3.4.1. [26]

Remark B.3.1. *When only one arm is activated according to* $\mathbf{p}(t)$*, the Hoeffding's inequality is adapted as follows:*

$$\begin{aligned}\mathbb{P}\big(|\overline{U}_t(\mathbf{p}) - U(\mathbf{p})| \geq \delta\big) &\leq \sum_k \mathbb{P}\big(|p_k\bar{r}(p_k) - p_k r(p_k)| \geq \delta/K\big) \\ &\leq \sum_k 2\exp\left(-2\delta^2 n_t(p_k)/K^2\right) \leq 2K\exp\left(-2\delta^2 n_t(p_{\min}(\mathbf{p}))/K^2\right).\end{aligned}$$

[26] Here the choice of ϵ absorbs Lipschitz constant of $r_k(\cdot)$.

The below analysis carries over with accordingly changing $\delta = \sqrt{\frac{K \ln t}{n_t(p_{\min}(\mathbf{p}))}}$ *to* $\delta = \sqrt{\frac{K^2 \ln(\sqrt{K}t)}{n_t(p_{\min}(\mathbf{p}))}}$, *and the condition of* $n_t(p_{\min}(\mathbf{p}))$ *in Lemma B.3.2 is changed to* $4K^2 \ln(\sqrt{K}t)/\Delta_{\mathbf{p}}^2$ *to account for larger* δ. *As a result, the instance-independent regret bound in Lemma B.3.5 is changed to* $\mathcal{O}\left(K\sqrt{KT\ln(\sqrt{K}T)/\epsilon}\right)$. *Together with the discretization error, one can then optimize the choice of* ϵ *to get* $\tilde{\mathcal{O}}(K^{4/3}T^{2/3})$ *regret bound.*

Regret Bound Comparison with [29]

In the work [29], the authors study the setting when pulling the meta arm, each base arm in (or possibly other base arm) this meta arm will be triggered and played as a result. Back to our setting, this is saying that when pulling a meta arm $\mathbf{p} = (p_1, \ldots, p_K)$, each base arm k will be triggered with its corresponding probability (discretized arm) p_k. The authors in [29] discuss a general setting which allows complex reward structure where only requires two mild conditions. In particular, one of the condition they need for expected reward of playing a meta arm is the bounded smoothness (cf., Definition 1 in [29].). In the Theorem 2 of [29], the authors give results when the function used to characterize bounded smoothness is $f(x) = \gamma \cdot x^{\omega}$ for some $\gamma > 0$ and $\omega \in (0, 1]$. In more detail, they achieve a regret bound $\mathcal{O}\left(\frac{2\gamma}{2-\omega}\left(\frac{12|\mathcal{M}|\ln T}{p^*}\right)^{\omega/2} \cdot T^{1-\omega/2} + |\mathcal{M}| \cdot \Delta_{\max}\right)$ where $p^* \in (0, 1)$ is the minimum triggering probability across all base arms and $\Delta_{\max}$ is the largest badness of the suboptimal meta arm in discretized space. [27] Adapt to our setting, by inspection, we have $\gamma = L^*, \omega = 1$, $p^* = \epsilon$, $|\mathcal{M}| = \Theta(K/\epsilon)$, and $\Delta_{\max} = \Theta(KL^*)$. Substituting these values to the above bound, ignoring constant factors and combining with the discretization error, we have

$$\mathcal{O}\left(\left(\frac{K\ln T}{\epsilon^2}\right)^{1/2} \cdot T^{1/2} + K^2/\epsilon\right) + \mathcal{O}(TK\epsilon).$$

Picking $\epsilon = \Theta(\ln T/(KT))^{1/4}$ will give us result.

[27] For simplicity, the bound we present here omits a non-significant term.

B.4 Proof of Theorem 3.5.1 for History-dependent Bandits

In this section, we provide the analysis of Theorem 3.5.1. The analysis follows a similar structure to the one used in the proof of the regret bound in Theorem 3.4.1. However, due to the existence of historical bias, we need to perform a careful computation when handling the high-probability bounds. Specifically, we need to prove that, after deploying $\mathbf{p}$ consecutively for moderate long rounds (tuning s_a), the approximation error $\left|U(\mathbf{p}) - \overline{U}_m^{\texttt{est}}(\mathbf{p})\right|$ is small enough. The analysis is provided below.

Step 1: Bounding the small error of $\left|U(\mathbf{p}) - \overline{U}_m^{\texttt{est}}(\mathbf{p})\right|$ with high-probability. Our first step is to ensure the empirical mean reward estimation we obtain from the information we collected in all the estimation stages will approximate well the true mean of meta arm we want to deploy.

To return a high-probability error bound, we first bound the approximation error incurred due to the dependency of history of arm selection ("historical bias"). This is summarized below.

Lemma B.4.1. *Keeping deploying $\mathbf{p} = \{p_1, \ldots, p_K\}$ in the approaching stage with s_a rounds, and collect all reward feedback in the following estimation stage for the empirical estimation of rewards generated by $\mathbf{p}$, one can bound the approximation error as follows:*

$$\mathbb{E}\left[\left|\overline{U}_m^{\texttt{est}}(\mathbf{p}) - \overline{U}(\mathbf{p})\right|\right] \leq K\gamma^{s_a}(L^* + 1),$$

where $\overline{U}(\mathbf{p})$ denote the empirical mean of rewards if the instantaneous reward is truly sampled from mean reward function according to $\mathbf{p}$.

Proof. The proof of this lemma is mainly built on analyzing the convergence of $\mathbf{p}^{(\gamma)}$ via pulling the base arms with the same probability consistently. For the ease of presentation, let us suppose $t = mL$ and let $t_m^{\texttt{est}} := \frac{t}{L}(L - s_a) = m(L - s_a)$ be the total number of estimation rounds in the first m phases. Thus, at the end of the approaching stage, we have

$$\hat{p}_k^{(\gamma)}(t + s_a) = \frac{p_k(t + s_a)\gamma^0 + \ldots + p_k(t + 1)\gamma^{s_a - 1} + (1 + \gamma + \ldots + \gamma^{t-1})\gamma^{s_a}\hat{p}_k^{(\gamma)}(t)}{1 + \gamma + \ldots + \gamma^{t+s_a-1}},$$

where $\hat{p}_k^{(\gamma)}(t) = \frac{p_k(t)\gamma^0 + \ldots + p_k(1)\gamma^{t-1}}{1+\gamma+\ldots+\gamma^{t-1}}$. Recall that during the approaching stage, we consistently pull arm k with the same probability p_k. Thus, the approximation error of $\hat{p}_k^{(\gamma)}(t + s_a)$ w.r.t. p_k can be computed as:

$$\left|\hat{p}_k^{(\gamma)}(t + s_a) - p_k\right| = \left|\frac{p_k(1 - \gamma^{s_a}) + \hat{p}_k^{(\gamma)}(t)\gamma^{s_a}(1 - \gamma^t)}{1 - \gamma^{t+s_a}} - p_k\right| \leq \frac{\gamma^{s_a}(1 - \gamma^t)}{1 - \gamma^{t+s_a}} < \gamma^{s_a}.$$

Recall that $U(\mathbf{p}) = \sum_{p_k \in \mathbf{p}} p_k r_k(p_k)$. In the estimation stage, we approximate all the realized utility as the utility generated by the meta arm $\mathbf{p}$. However, note that we actually cannot compute the empirical value of $\overline{U}(\mathbf{p})$, instead, we use $\overline{U}_m^{\texttt{est}}(\mathbf{p}(t + s_a))$ of each phase as an approximation of $\overline{U}(\mathbf{p})$, i.e., we approximate all $\mathbf{p}^{(\gamma)}(t + s), \forall s \in (s_a, L]$ as $\mathbf{p}(t + s_a)$ and use $\mathbf{p}(t + s_a)$ as the approximation of $\mathbf{p}$. Recall that for any $s \in (s_a, L]$, we have:

$$\left|\hat{p}_k^{(\gamma)}(t + s) - p_k\right| = \left|\frac{\gamma^s(1 - \gamma^t)(\hat{p}_k^{(\gamma)}(t) - p_k)}{1 - \gamma^{t+s}}\right| \leq \frac{\gamma^s(1 - \gamma^t)}{1 - \gamma^{t+s}} < \frac{\gamma^{s_a}(1 - \gamma^t)}{1 - \gamma^{t+s_a}} < \gamma^{s_a}.$$

Thus, the approximation error on the empirical estimation can be computed as follows:

$$\begin{aligned}
\mathbb{E}\left[\left|\overline{U}_m^{\texttt{est}}(\mathbf{p}(t+s_a)) - \overline{U}(\mathbf{p})\right|\right] &= \mathbb{E}\left[\left|\sum_{p_k^{(\gamma)} \in \mathbf{p}(t+s_a)} p_k^{(\gamma)} \bar{r}_{t+s_a}^{\texttt{est}}(p_k^{(\gamma)}) - \sum_{p_k \in \mathbf{p}} p_k \bar{r}_{t+s_a}^{\texttt{est}}(p_k)\right|\right] \\
&= \left|\sum p_k^{(\gamma)} \mathbb{E}\left[\bar{r}_{t+s_a}^{\texttt{est}}(p_k^{(\gamma)})\right] - \sum p_k \mathbb{E}\left[\bar{r}_{t+s_a}^{\texttt{est}}(p_k)\right]\right| \\
&= \left|\sum p_k^{(\gamma)} r_k(p_k^{(\gamma)}) - \sum p_k r_k(p_k)\right| \\
&= \left|\sum \left(p_k^{(\gamma)}\left(r_k(p_k^{(\gamma)}) - r_k(p_k)\right) + r_k(p_k)(p_k^{(\gamma)} - p_k)\right)\right| \\
&\leq \sum \left|\gamma^{s_a} L_k p_k^{(\gamma)} + r_k(p_k)\gamma^{s_a}\right| \leq K\gamma^{s_a}(L^* + 1).
\end{aligned}$$

□

With the approximation error at hand, we can then bound the error of $\left|U(\mathbf{p}) - \overline{U}_m^{\texttt{est}}(\mathbf{p})\right|$ with high probability:

Lemma B.4.2. *With probability at least* $1 - \frac{6}{\left(L\rho m\right)^2}$*, we have*

$$\left|U(\mathbf{p}) - \overline{U}_m^{\texttt{est}}(\mathbf{p})\right| \leq \texttt{err} + 3\sqrt{\frac{K \ln\left(L\rho m\right)}{n_m^{\texttt{est}}(p_{\min}(\mathbf{p}))}},$$

where $p_{\min}(\mathbf{p}) = \arg\min_{p_k \in \mathbf{p}} n_m^{\texttt{est}}(p_k)$.

Proof. We first decompose $\left|U(\mathbf{p}) - \overline{U}_m^{\texttt{est}}(\mathbf{p}_e^{(\gamma)})\right|$ as $\left|U(\mathbf{p}) - \overline{U}(\mathbf{p})\right| + \left|\overline{U}(\mathbf{p}) - \overline{U}_m^{\texttt{est}}(\mathbf{p})\right|$ and then apply union bound.

$$
\begin{aligned}
&\mathbb{P}\left(\left|U(\mathbf{p}) - \overline{U}_m^{\texttt{est}}(\mathbf{p}(t+s_a))\right| \geq \delta\right) \\
&\leq \mathbb{P}\left(\left|U(\mathbf{p}) - \overline{U}(\mathbf{p})\right| + \left|\overline{U}(\mathbf{p}) - \overline{U}_m^{\texttt{est}}(\mathbf{p}(t+s_a))\right| \geq \delta\right) \qquad \text{By triangle inequality} \\
&= \mathbb{P}\Bigg(\left|U(\mathbf{p}) - \overline{U}(\mathbf{p})\right| + \left|\overline{U}_m^{\texttt{est}}(\mathbf{p}(t+s_a)) - \mathbb{E}[\overline{U}_m^{\texttt{est}}(\mathbf{p}(t+s_a))] - \\
&\qquad (\overline{U}(\mathbf{p}) - \mathbb{E}[\overline{U}(\mathbf{p})]) + \mathbb{E}[\overline{U}(\mathbf{p})] - \mathbb{E}[\overline{U}_m^{\texttt{est}}(\mathbf{p}(t+s_a))]\right| \geq \delta\Bigg) \\
&\leq \mathbb{P}\left(2\left|U(\mathbf{p}) - \overline{U}(\mathbf{p})\right| + \left|\overline{U}_m^{\texttt{est}}(\mathbf{p}(t+s_a)) - \mathbb{E}[\overline{U}_m^{\texttt{est}}(\mathbf{p}(t+s_a))\right| \geq \delta - \texttt{err}\right) \\
&\overset{\text{(a)}}{\leq} 3\mathbb{P}\left(\left|U(\mathbf{p}) - \overline{U}(\mathbf{p})\right| \geq \frac{\delta - \texttt{err}}{3}\right) \leq 6\exp\left(-\frac{2n_m^{\texttt{est}}(p_{\min}(\mathbf{p}))(\delta - \texttt{err})^2}{9K}\right),
\end{aligned}
$$

where in step (a), we use the Hoeffding's Inequality on Weighted Sums and Lemma B.4.1. □

Step 2: Bounding the probability on deploying suboptimal meta arm. Till now, with the help of the above high probability bound on the empirical reward estimation, the history-dependent reward bandit setting is largely reduced to an action-dependent one with a certain approximation error. Then, similar to our argument on upper bound of action-dependent bandits, we have the following specific Lemma for history-dependent bandits:

Lemma B.4.3. *At the end of each phase, for a suboptimal meta arm* $\mathbf{p}$*, if it satisfies* $n_m^{\texttt{est}}(p_{\min}(\mathbf{p})) \geq \frac{9K\ln\left(L\rho m\right)}{\left(\Delta_{\mathbf{p}}/2 - \texttt{err}\right)^2}$*, then with the probability at least* $1 - \frac{12}{\left(L\rho m\right)^2}$*, we have* $\texttt{UCB}_m(\mathbf{p}) < \texttt{UCB}_m(\mathbf{p}^*)$*, i.e.,*

$$
\mathbb{P}\left(\mathbf{p}(m+1) = \mathbf{p} | n_m^{\texttt{est}}(p_{\min}(\mathbf{p})) \geq \frac{9K\ln\left(L\rho m\right)}{\left(\frac{\Delta_{\mathbf{p}}}{2} - \texttt{err}\right)^2}\right) \leq \frac{12}{\left(L\rho m\right)^2}.
$$

Proof. To prove the above lemma, we construct two high-probability events. `Event 1` corresponds to that the `UCB` index of each meta arm concentrates on the true mean utility of

$\mathbf{p}$; `Event 2` corresponds to that the empirical mean utility of each approximated meta arm $\mathbf{p}^{(\gamma)}$ concentrates on the true mean utility of $\mathbf{p}$. The probability of `Event 1` or `Event 2` not holding is at most $4/t^2$. By the definition of the constructed `UCB`, we'll have

$$\begin{aligned}
\texttt{UCB}_m(\mathbf{p}) &= \overline{U}_m^{\texttt{est}}(\mathbf{p}(t+s_a)) + \texttt{err} + 3\sqrt{\frac{K\ln(L\rho m)}{n_m^{\texttt{est}}(p_{\min}(\mathbf{p}))}} \overset{\text{(a)}}{\leq} \overline{U}_m^{\texttt{est}}(\mathbf{p}(t+s_a)) + \Delta_{\mathbf{p}}/2 \\
&\overset{\text{(b)}}{<} (U(\mathbf{p}) + \Delta_{\mathbf{p}}/2) + \Delta_{\mathbf{p}}/2 && \text{By } \texttt{Event 1} \\
&= U(\mathbf{p}_\epsilon^*) \overset{\text{(c)}}{<} \texttt{UCB}_m(\mathbf{p}_\epsilon^*), && \text{By } \texttt{Event 2}
\end{aligned}$$

where the first inequality (a) is due to $n_m^{\texttt{est}}(p_{\min}(\mathbf{p})) \geq \frac{9K\ln(L\rho m)}{(\Delta_{\mathbf{p}}/2-\texttt{err})^2}$, and the probability of step (b) or (c) not holding is at most $12/(L\rho m)^2$. □

The above lemma implies that we will stop deploying suboptimal meta arm $\mathbf{p}$ and further prevent it from incurring regret as we gather more information about it such that $\texttt{UCB}_m(\mathbf{p}) < \texttt{UCB}_m(\mathbf{p}_\epsilon^*)$.

Step 3: Bounding the $\mathbb{E}[n_m^{\texttt{est}}(p_{\min}(\mathbf{p}))]$. The results we obtain in Step 2 implies following guarantee:

Lemma B.4.4. *For each suboptimal meta arm $\mathbf{p} \neq \mathbf{p}^*$, we have following:*

$$\mathbb{E}[n_m^{\texttt{est}}(p_{\min}(\mathbf{p}))] \leq \frac{9K\ln\left(L\rho m\right)}{\left(\Delta_{\mathbf{p}}/2 - \texttt{err}\right)^2} + \frac{2\pi^2}{L - s_a}.$$

Proof. For notation simplicity, suppose $t = mL$. For each suboptimal arm $\mathbf{p} \neq \mathbf{p}^*_\epsilon$, and suppose there exists $p_{\min}(\mathbf{p}) \notin \mathbf{p}^*_\epsilon$ such that $p_{\min}(\mathbf{p}) = \arg\min_{p_k \in \mathbf{p}} n_t^{\texttt{est}}(p_k)$, then

$$
\begin{aligned}
&\mathbb{E}[n_t^{\texttt{est}}(p_{\min}(\mathbf{p}))] \\
&= (L - s_a)\mathbb{E}\left[\sum_{i=1}^{m} \mathbb{1}\left(\mathbf{p}(i) = \mathbf{p}, \mathbf{p} \in \mathcal{S}(p_{\min}(\mathbf{p}))\right)\right] \\
&= (L - s_a)\mathbb{E}\left[\sum_{i=1}^{m} \mathbb{1}\left(\mathbf{p}(i) = \mathbf{p}, \mathbf{p} \in \mathcal{S}(p_{\min}(\mathbf{p})); n_i^{\texttt{est}}(p_{\min}(\mathbf{p})) < \frac{9K\ln\left(i(L-s_a)\right)}{\left(\Delta_{\mathbf{p}}/2 - \texttt{err}\right)^2}\right)\right] + \\
&\quad (L - s_a)\mathbb{E}\left[\sum_{i=1}^{m} \mathbb{1}\left(\mathbf{p}(i) = \mathbf{p}, \mathbf{p} \in \mathcal{S}(p_{\min}(\mathbf{p})); n_i^{\texttt{est}}(p_{\min}(\mathbf{p})) \geq \frac{9K\ln\left(i(L-s_a)\right)}{\left(\Delta_{\mathbf{p}}/2 - \texttt{err}\right)^2}\right)\right] \\
&\overset{\text{(a)}}{\leq} \frac{9K\ln\left(t_m^{\texttt{est}}\right)}{\left(\Delta_{\mathbf{p}}/2 - \texttt{err}\right)^2} + (L - s_a)\mathbb{E}\left[\sum_{i=1}^{m} \mathbb{1}\left(\mathbf{p}(i) = \mathbf{p}, \mathbf{p} \in \mathcal{S}(p_{\min}(\mathbf{p})); n_i^{\texttt{est}}(p_{\min}(\mathbf{p})) \geq \frac{9K\ln\left(i(L-s_a)\right)}{\left(\Delta_{\mathbf{p}}/2 - \texttt{err}\right)^2}\right)\right] \\
&= \frac{9K\ln\left(t_m^{\texttt{est}}\right)}{\left(\Delta_{\mathbf{p}}/2 - \texttt{err}\right)^2} + (L - s_a)\sum_{i=1}^{m} \mathbb{P}\left(\mathbf{p}(i) = \mathbf{p}, \mathbf{p} \in \mathcal{S}(p_{\min}(\mathbf{p}))\middle| n_i^{\texttt{est}}(p_{\min}(\mathbf{p})) \geq \frac{9K\ln\left(i(L-s_a)\right)}{\left(\Delta_{\mathbf{p}}/2 - \texttt{err}\right)^2}\right) \cdot \\
&\qquad\qquad \mathbb{P}\left(n_i^{\texttt{est}}(p_{\min}(\mathbf{p})) \geq \frac{9K\ln\left(i(L-s_a)\right)}{\left(\Delta_{\mathbf{p}}/2 - \texttt{err}\right)^2}\right) \\
&\leq \frac{9K\ln\left(t_m^{\texttt{est}}\right)}{\left(\Delta_{\mathbf{p}}/2 - \texttt{err}\right)^2} + (L - s_a)\sum_{i=1}^{m} \frac{12}{\left(i(L-s_a)\right)^2} \leq \frac{9K\ln\left(t_m^{\texttt{est}}\right)}{\left(\Delta_{\mathbf{p}}/2 - \texttt{err}\right)^2} + \frac{2\pi^2}{L - s_a}.
\end{aligned}
$$

In step (a), suppose for contradiction that the indicator $\mathbb{1}\left(\mathbf{p}(i) = \mathbf{p}, \mathbf{p} \in \mathcal{S}(p_{\min}(\mathbf{p})); n_i^{\texttt{est}}(p_{\min}(\mathbf{p})) < S\right)$ takes value of 1 at more than $S-1$ time steps, where $S = \frac{9K\ln(i(S-s_a))}{\left(\Delta_{\mathbf{p}}/2-\texttt{err}\right)^2}$. Let τ be the phase at which this indicator is 1 for the $(S-1)$-th phase. Then the number of pulls of all meta arms in $\mathcal{S}(p_{\min}(\mathbf{p}))$ is at least L times until time τ (including the initial pull), and for all $i > \tau$, $n_i(p_{\min}(\mathbf{p})) \geq S$ which implies $n_i^{\texttt{est}}(p_{\min}(\mathbf{p})) \geq \frac{9K\ln(i(S-s_a))}{\left(\Delta_{\mathbf{p}}/2-\texttt{err}\right)^2}$. Thus, the indicator cannot be 1 for any $i \geq \tau$, contradicting the assumption that the indicator takes value of 1 more than S times. This bounds $1 + \mathbb{E}\left[\sum_{i=1}^{m} \mathbb{1}\left(\mathbf{p}(i) = \mathbf{p}, \mathbf{p} \in \mathcal{S}(p_{\min}(\mathbf{p})); n_i^{\texttt{est}}(p_{\min}(\mathbf{p})) < S\right)\right]$ by S. □

Wrapping up: Proof of Theorem 3.5.1. Following the similar analysis in Section 3, we can also get an instance-dependent regret bound for history-dependent bandits:

Lemma B.4.5. *Following the* `UCB` *designed in Algorithm 5, we have following instance-dependent regret on discretized arm space for history-dependent bandits:*

$$\mathrm{Reg}_\epsilon(T) \leq \mathcal{O}\left(\frac{K\Delta_{\max}}{L\epsilon\rho^2}\right) + \sum_{p_k}\left(\frac{9K\ln(T\rho)}{\rho}\left(\frac{\Delta_{\min}^{p_k}}{(\Delta_{\min}^{p_k}/2 - \texttt{err})^2} + \frac{2}{\Delta_{\min}^{p_k}/2 - \texttt{err}}\right)\right).$$

Proof. For notation simplicity, we include all initialization rounds to phase 0 and suppose the time horizon $T = ML$. Note that by definitions, we can compute the regret $\mathrm{Reg}_\epsilon(T)$ as follows:

$$\mathrm{Reg}_\epsilon(T) = \sum_{\mathbf{p}\in\mathcal{P}_\epsilon} \mathbb{E}[N_T(\mathbf{p})]\Delta_{\mathbf{p}} \leq \sum_{p_k}\sum_{\mathbf{p}_l\in\mathcal{S}(p_k)} \mathbb{E}[N_T(\mathbf{p}_l)]\Delta_l^{p_k}. \tag{B.12}$$

where $N_t(\mathbf{p}) = K + L\sum_{m=1}^{M}\mathbb{1}(\mathbf{p}(m) = \mathbf{p})$, where K here accounts for the initialization. Follow the same analysis in action-dependent bandits, we can also define a counter $c^{\texttt{est}}(p_k)$ for each discretized arm p_k and the value of $c^{\texttt{est}}(p_k)$ at phase m is denoted by $c_m^{\texttt{est}}(p_k)$. But different from the action-dependent bandit setting, we update the counter $c^{\texttt{est}}(p_k)$ only when we start a new phase. In particular, for a phase $m \geq 1$, let $\mathbf{p}(m)$ be the meta arm selected in the phase m by the algorithm. Let $p_k = \arg\min_{p_k\in\mathbf{p}(m)} c_m^{\texttt{est}}(p_k)$. We increment $c_m^{\texttt{est}}(p_k)$ by one, i.e., $c_m^{\texttt{est}}(p_k) = c_{m-1}^{\texttt{est}}(p_k) + 1$. In other words, we find the discretized arm p_k with the smallest counter in $\mathbf{p}(m)$ and increment its counter. If such p_k is not unique, we pick an arbitrary discretized arm with the smallest counter. Note that the initialization gives $\sum_{p_k} c_0^{\texttt{est}}(p_k) = \lceil K/\epsilon\rceil$. It is easy to see that for any $p_k = p_{\min}(\mathbf{p})$, we have $n_m(p_k) = L\rho\cdot c_m(p_k)$.

Like in action-dependent bandits, we also define $C_M^{\texttt{est}}(\Delta) := \frac{9K\ln(ML\rho)}{L\rho(\Delta/2-\texttt{err})^2}$, the number of selection that is considered sufficient for a meta arm with reward Δ away from the optimal strategy $\mathbf{p}_\epsilon^*$ with respect to phase horizon M. With the above notations, we define following

two situations for the counter of each discretized arm:

$$c_M^{\texttt{est},l,\texttt{suf}}(p_k) := \sum_{m=1}^{M} \mathbb{1}\left(\mathbf{p}(m) = \mathbf{p}_l, c_m^{\texttt{est}}(p_k) > c_{m-1}^{\texttt{est}}(p_k) > C_M^{\texttt{est}}(\Delta_l^{p_k})\right) \tag{B.13}$$

$$c_M^{\texttt{est},l,\texttt{und}}(p_k) := \sum_{m=1}^{M} \mathbb{1}\left(\mathbf{p}(m) = \mathbf{p}_l, c_m^{\texttt{est}}(p_k) > c_{m-1}^{\texttt{est}}(p_k), c_{m-1}^{\texttt{est}}(p_k) \leq C_M^{\texttt{est}}(\Delta_l^{p_k})\right). \tag{B.14}$$

Clearly, we have $c_M^{\texttt{est}}(p_k) = 1 + \sum_{l\in I(p_k)} \left(c_M^{\texttt{est},l,\texttt{suf}}(p_k) + c_M^{\texttt{est},l,\texttt{und}}(p_k)\right)$. With these notations, we can write (B.12) as follows:

$$\mathrm{Reg}_\epsilon(T) \leq \mathbb{E}\left[\sum_{p_k}\left(\Delta_{\max}^{p_k} + L\cdot\sum_{l\in[I(p_k)]}\left(c_M^{\texttt{est},l,\texttt{suf}}(p_k) + c_M^{\texttt{est},l,\texttt{und}}(p_k)\right)\cdot\Delta_l^{p_k}\right)\right]. \tag{B.15}$$

We now first show that for any $m \geq 1$, we have following upper bound over counters of sufficiently selected discretized arms:

$$\mathbb{E}\left[L\cdot\sum_{p_k}\sum_{l\in[I(p_k)]} c_M^{l,\texttt{suf}}(p_k)\right] \leq \mathcal{O}\left(\frac{K}{L\epsilon\rho^2}\right). \tag{B.16}$$

To see this, by definition of $c_M^{\texttt{est},l,\texttt{suf}}(p_k)$, it reduces to show that for any $M \geq m > 1$,

$$\begin{aligned}
&\mathbb{E}\left[L\cdot\sum_{p_k}\sum_{l\in[I(p_k)]}\mathbb{1}\left(\mathbf{p}(m) = \mathbf{p}_l, c_m^{\texttt{est}}(p_k) > c_{m-1}^{\texttt{est}}(p_k) > C_M^{\texttt{est}}(\Delta_l^{p_k})\right)\right] \\
&= L\cdot\sum_{p_k}\sum_{l\in[I(p_k)]}\mathbb{P}\left(\mathbf{p}(m) = \mathbf{p}_l, p_k = p_{\min}(\mathbf{p}_l); \forall p\in\mathbf{p}_l, L\rho\cdot c_{m-1}^{\texttt{est}}(p) > \frac{9K\ln(ML\rho)}{\left(\Delta_l^{p_k}/2 - \texttt{err}\right)^2}\right) \\
&\overset{\text{(a)}}{\leq} \lceil 12LK/\epsilon\rceil\cdot(ML\rho)^{-2},
\end{aligned}$$

where the last step (a) is due to Lemma B.4.3, thus (B.16) follows from a simple series bound.

We now proceed to analyze the discretized arms that are not sufficiently included in the meta arm chosen by the algorithm. For any under-selected discretized arm p_k, its counter $c^{\texttt{est}}(p_k)$

will increase from 1 to $C_M^{\mathtt{est}}(\Delta_{\min}^{p_k})$. To simplify the notation, we set $C_M^{\mathtt{est}}(\Delta_0^{p_k}) = 0$. Suppose that at phase $m \geq 1$, $c^{\mathtt{est}}(p_k)$ is incremented, and $c_{m-1}^{\mathtt{est}}(p_k) \in (C_M^{\mathtt{est}}(\Delta_{j-1}^{p_k}), C_M^{\mathtt{est}}(\Delta_j^{p_k})]$ for some $j \in [I(p_k)]$. Notice that we are only interested in the case that p_k is under-selected. In particular, if this is indeed the case, $\mathbf{p}(m) = \mathbf{p}_l$ for some $l \geq j$. (Otherwise, $\mathbf{p}(m)$ is sufficiently selected based on the counter value $c_{m-1}^{\mathtt{est}}(p_k)$.) Thus, we will suffer a regret of $\Delta_l^{p_k} \leq \Delta_j^{p_k}$ (step (a)). As a result, for counter $c_m^{\mathtt{est}}(p_k) \in (C_M^{\mathtt{est}}(\Delta_{j-1}^{p_k}), C_M^{\mathtt{est}}(\Delta_j^{p_k})/L]$, we will suffer a total regret for those playing suboptimal meta arms that include under-selected discretized arms at most $(C_M^{\mathtt{est}}(\Delta_j^{p_k}) - C_M^{\mathtt{est}}(\Delta_{j-1}^{p_k})) \cdot \Delta_j^{p_k}$ in rounds that $c_m^{\mathtt{est}}(p_k)$ is incremented (step (b)). In what follows we establish the above analysis rigorously.

$$
\begin{aligned}
&\sum_{l \in [I(p_k)]} c_M^{\mathtt{est},l,\mathtt{und}}(p_k) \Delta_l^{p_k} \\
&= \sum_{m=1}^{M} \sum_{l \in [I(p_k)]} \mathbb{1}\left(\mathbf{p}(m) = \mathbf{p}_l, c_m^{\mathtt{est}}(p_k) > c_{m-1}^{\mathtt{est}}(p_k), c_{m-1}^{\mathtt{est}}(p_k) \leq C_M^{\mathtt{est}}(\Delta_l^{p_k})\right) \cdot \Delta_l^{p_k} \\
&= \sum_{m=1}^{M} \sum_{l \in [I(p_k)]} \sum_{j=1}^{l} \mathbb{1}\left(\mathbf{p}(m) = \mathbf{p}_l, c_m^{\mathtt{est}}(p_k) > c_{m-1}^{\mathtt{est}}(p_k), c_{m-1}^{\mathtt{est}}(p_k) \in (C_M^{\mathtt{est}}(\Delta_{j-1}^{p_k}), C_M^{\mathtt{est}}(\Delta_j^{p_k})]\right) \cdot \Delta_l^{p_k} \\
&\overset{\text{(a)}}{\leq} \sum_{m=1}^{M} \sum_{l \in [I(p_k)]} \sum_{j=1}^{l} \mathbb{1}\left(\mathbf{p}(m) = \mathbf{p}_l, c_m^{\mathtt{est}}(p_k) > c_{m-1}^{\mathtt{est}}(p_k), c_{m-1}^{\mathtt{est}}(p_k) \in (C_M^{\mathtt{est}}(\Delta_{j-1}^{p_k}), C_M^{\mathtt{est}}(\Delta_j^{p_k})]\right) \cdot \Delta_j^{p_k} \\
&\leq \sum_{m=1}^{M} \sum_{l,j \in [I(p_k)]} \mathbb{1}\left(\mathbf{p}(m) = \mathbf{p}_l, c_m^{\mathtt{est}}(p_k) > c_{m-1}^{\mathtt{est}}(p_k), c_{m-1}^{\mathtt{est}}(p_k) \in (C_M^{\mathtt{est}}(\Delta_{j-1}^{p_k}), C_M^{\mathtt{est}}(\Delta_j^{p_k})]\right) \cdot \Delta_j^{p_k} \\
&= \sum_{m=1}^{M} \sum_{j \in [I(p_k)]} \mathbb{1}\left(\mathbf{p}(m) \in \mathcal{S}(p_k), c_m^{\mathtt{est}}(p_k) > c_{m-1}^{\mathtt{est}}(p_k), c_{m-1}^{\mathtt{est}}(p_k) \in (C_M^{\mathtt{est}}(\Delta_{j-1}^{p_k}), C_M^{\mathtt{est}}(\Delta_j^{p_k})]\right) \cdot \Delta_j^{p_k} \\
&\overset{\text{(b)}}{\leq} \sum_{j \in [I(p_k)]} (C_M^{\mathtt{est}}(\Delta_j^{p_k}) - C_M^{\mathtt{est}}(\Delta_{j-1}^{p_k})) \cdot \Delta_j^{p_k}.
\end{aligned}
$$

Now, we can compute the regret incurred by selecting the meta arm which includes under-selected discretized arms:

$$
\begin{aligned}
& L \cdot \sum_{p_k} \sum_{l \in [I(p_k)]} c_M^{\texttt{est},l,\texttt{und}}(p_k) \cdot \Delta_l^{p_k} \\
\leq\ & L \cdot \sum_{p_k} \sum_{j \in [I(p_k)]} (C_M^{\texttt{est}}(\Delta_j^{p_k}) - C_M^{\texttt{est}}(\Delta_{j-1}^{p_k})) \cdot \Delta_j^{p_k} \\
=\ & L \cdot \sum_{p_k} \left(C_M^{\texttt{est}}(\Delta_{\min}^{p_k}) \Delta_{\min}^{p_k} + \sum_{j \in [I(p_k)-1]} C_M^{\texttt{est}}(\Delta_j^{p_k}) \cdot (\Delta_j^{p_k} - \Delta_{j+1}^{p_k}) \right) \\
\leq\ & L \cdot \sum_{p_k} \left(C_M^{\texttt{est}}(\Delta_{\min}^{p_k}) \Delta_{\min}^{p_k} + \int_{\Delta_{\min}^{p_k}}^{\Delta_{\max}^{p_k}} C_M^{\texttt{est}}(x) dx \right) \\
=\ & \sum_{p_k} \left(\frac{9K \ln(ML\rho)}{\rho \left(\Delta_{\min}^{p_k}/2 - \texttt{err}\right)^2} \cdot \Delta_{\min}^{p_k} + 9K \ln(ML\rho) / \rho \cdot \int_{\Delta_{\min}^{p_k}}^{\Delta_{\max}^{p_k}} \frac{1}{(x/2 - \texttt{err})^2} dx \right) \\
=\ & \sum_{p_k} \left(\frac{9\Delta_{\min}^{p_k} K \ln(ML\rho)}{\rho \left(\Delta_{\min}^{p_k}/2 - \texttt{err}\right)^2} + \frac{9K \ln(ML\rho)}{\rho} \left(\frac{2}{\frac{\Delta_{\min}^{p_k}}{2} - \texttt{err}} - \frac{2}{\Delta_{\max}^{p_k}/2 - \texttt{err}} \right) \right) \\
\leq\ & \sum_{p_k} \left(\frac{9K \ln(ML\rho)}{\rho} \left(\frac{\Delta_{\min}^{p_k}}{\left(\Delta_{\min}^{p_k}/2 - \texttt{err}\right)^2} + \frac{2}{\Delta_{\min}^{p_k}/2 - \texttt{err}} \right) \right).
\end{aligned}
$$

Combing the bound established in (B.16) will complete the proof. □

The instance-independent regret on discretized arm space is summarized in following lemma:

Lemma B.4.6. *Following the* `UCB` *designed in Algorithm 5, the instance-independent regret is given as* $\mathrm{Reg}_\epsilon(T) \leq \mathcal{O}\left(K \cdot \sqrt{T \ln(T\rho)/(\rho\epsilon)} + K/(L\epsilon\rho^2)\right)$.

Proof. Following the proof action-dependent bandits, we only need to consider the meta arms that are played when they are under-sampled. We particularly need to deal with the situation when $\Delta_{\min}^{p_k}$ is too small. We measure the threshold for $\Delta_{\min}^{p_k}$ based on $c_M^{\texttt{est}}(p_k)$, i.e., the counter of disretized arm p_k at phase horizon M. Let $\{M(p_k), \forall p_k\}$ be a set of possible counter values at time horizon M. Our analysis will then be conditioned on the event that

$\mathcal{E}(p_k) := \{c_M^{\texttt{est}}(p_k) = M(p_k)\}$. By definition,

$$\begin{aligned}
&\mathbb{E}\big[\sum_{l\in[I(p_k)]} c_M^{\texttt{est},l,\texttt{und}}(p_k)\cdot\Delta_l^{p_k} \mid \mathcal{E}(p_k)\big] \\
&= \sum_{m=1}^{M}\sum_{l\in[I(p_k)]} \mathbb{1}\left(\mathbf{p}(m)=\mathbf{p}_l, c_m^{\texttt{est}}(p_k) > c_{m-1}^{\texttt{est}}(p_k), c_{m-1}^{\texttt{est}}(p_k) \le C_M^{\texttt{est}}(\Delta_l^{p_k}) \mid \mathcal{E}(p_k)\right)\cdot\Delta_l^{p_k}.
\end{aligned} \tag{B.17}$$

We define $\Delta^*(M(p_k)) := 2\left(\frac{9K\ln(ML\rho)}{L\rho\cdot M(p_k)}\right)^{1/2} + 2\texttt{err}$. thus we have $C_M^{\texttt{est}}(\Delta^*(M(p_k))) = M(p_k)$. To achieve *instance-independent* regret bound, we consider following two cases:

Case 1: $\Delta_{\min}^{p_k} > \Delta^*(M(p_k))$, clearly we have $\Delta_{\min}^{p_k}/2 > \texttt{err}$. Thus,

$$L\cdot\mathbb{E}\big[\sum_{l\in[I(p_k)]} c_M^{\texttt{est},l,\texttt{und}}(p_k)\cdot\Delta_l^{p_k} \mid \mathcal{E}(p_k)\big] \le \mathcal{O}\left(\sqrt{\frac{K\ln(T\rho)\cdot LM(p_k)}{\rho}}\right). \tag{B.18}$$

Case 2: $\Delta_{\min}^{p_k} < \Delta^*(M(p_k))$. Let $l^* := \min\{l\in I(p_k) : \Delta_l^{p_k} > \Delta^*(M(p_k))\}$. Observe that we have $\Delta_{l^*}^{p_k} \le \Delta^*(M(p_k))$ and the counter $c^{\texttt{est}}(p_k)$ never go beyond $M(p_k)$, we thus have

$$\begin{aligned}
L\cdot\text{(B.17)} &\le L(C_M^{\texttt{est}}(\Delta^*(M(p_k))) - C_M^{\texttt{est}}(\Delta_{l^*-1}^{p_k}))\cdot\Delta^*(M(p_k)) + \sum_{j\in[l^*-1]} L(C_M^{\texttt{est}}(\Delta_j^{p_k}) - C_M^{\texttt{est}}(\Delta_{j-1}^{p_k}))\cdot\Delta_j^{p_k} \\
&\le LC_M^{\texttt{est}}(\Delta^*(M(p_k)))\cdot\Delta^*(M(p_k)) + L\int_{\Delta^*(M(p_k))}^{\Delta_{\max}^{p_k}} C_M^{\texttt{est}}(x)dx \\
&\le \mathcal{O}\left(\sqrt{\frac{K\ln(T\rho)\cdot LM(p_k)}{\rho}}\right).
\end{aligned} \tag{B.19}$$

Combining (B.18) and (B.19), and with Jesen's inequality and $\sum_{p_k} M(p_k) \le KM/\epsilon$ will give us desired result. Put all pieces together, we have the instance-independent regret bound as stated in the lemma. The final inequality does not depend on the event $\mathcal{E}(p_k)$, we thus can drop this conditional expectation. □

Combining with the discretization error, we have

$$\mathrm{Reg}(T) \leq \mathcal{O}\left(K \cdot \sqrt{T \ln(T\rho)/(\rho\epsilon)} + K/(L\epsilon\rho^2)\right) + \mathcal{O}(K\epsilon T).$$

Picking

$$\epsilon = \mathcal{O}\left(\frac{\ln(T\rho)}{T\rho}\right)^{1/3}; \quad s_a = \mathcal{O}\left(\frac{1/3 \ln\left(\frac{\ln(T\rho)}{T\rho}\right) - \ln(L^*K)}{\ln\gamma}\right).$$

We will obtain the results as stated in the theorem.