Robust Sequential Decision-Making on Networks

by

Abhimanyu Dubey

B.Tech., Indian Institute of Technology Delhi (2016)
M.Tech., Indian Institute of Technology Delhi (2016)

Submitted to the Program in Media Arts and Sciences,
School of Architecture and Planning,
in partial fulfillment of the requirements for the degree of

Master of Science in Media Arts and Sciences

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2019

Signature redacted

Author .....
Program in Media Arts and Sciences
May 13, 2019

Signature redacted

Certified by .....
Alex P. Pentland
Toshiba Professor of Media Arts and Sciences
Thesis Supervisor

Signature redacted

Accepted by .....
Tod Machover
Academic Head, Program in Media Arts and Sciences

Robust Sequential Decision-Making on Networks

by

Abhimanyu Dubey

Submitted to the Program in Media Arts and Sciences,
School of Architecture and Planning,
on May 13, 2019, in partial fulfillment of the
requirements for the degree of
Master of Science in Media Arts and Sciences

Abstract

In this thesis, I consider the research problem of designing optimal algorithms for two specific settings of the stochastic multi-armed bandit problem. The first setting considers the problem where rewards are drawn from a family of extremely heavy-tailed distributions known as α-stable distributions. For this setting, I extended an existing upper confidence bound algorithm, to create an optimal frequentist algorithm, titled α-UCB. Next, I developed a variant of the Bayesian Thompson Sampling algorithm in this setting, titled Robust α-TS, which involved developing an efficient pipeline for posterior inference. I also proved finite-time regret bounds for this algorithm, that are optimal up to logarithmic factors.

The second problem setting I considered was the networked multi-agent problem where agents have local communication, and have unique preferences. This problem setting is a generalization of the co-operative multi-agent stochastic bandit problem, and is a closely related variant of the single-agent bandit setting with side observations. For this setting, I developed an optimal upper confidence bound algorithm, titled Net-UCB. I also proved finite-time regret bounds for this algorithm that are logarithmic in the number of rounds, and are sub-linear in the number of agents.

For both settings, I conducted extensive experiments to verify the tightness of the regret bounds established, and compare performance with existing state-of-the-art algorithms. The algorithms proposed in this thesis obtain competitive regret and state-of-the-art performance across a variety of problem settings.

Thesis Supervisor: Alex P. Pentland
Title: Toshiba Professor of Media Arts and Sciences

Robust Sequential Decision-Making on Networks

by

Abhimanyu Dubey

This thesis has been reviewed and approved by the following committee members:

Signature redacted

Professor Alex P. Pentland..... .....
Thesis Supervisor
Toshiba Professor of Media Arts and Sciences,
Massachusetts Institute of Technology

Signature redacted

Professor Phillip Isola.....
Member, Thesis Committee
Assistant Professor, Electrical Engineering and Computer Science
Massachusetts Institute of Technology

Signature redacted

Dr. Moshe Hoffman..... .....
Member, Thesis Committee
Research Scientist, Human Dynamics Laboratory
Massachusetts Institute of Technology

Acknowledgments

Finding an exciting and challenging research problem can be a non-trivial problem in itself. Having always wanted to work on creating intelligent social systems, I would often find myself caught in a tussle between rigor and intuition. When I began my research at MIT, I was guided extensively by my advisor, Sandy, through a number of tough spots. Right from scoping a massive research problem into a manageable research plan to understanding the aspects of a problem that are worthy of pursuit, he has been instrumental in the success of my research, for which I am extremely grateful. I would also like to acknowledge my gratitude for my thesis committee members Phillip and Moshe, for providing me with several conversations about research, and for providing their time and effort to review my work.

Next, I would like to thank Dhaval Adjodah for the countless hours of whiteboard conversations, idle research chats, and (much-needed) companionship for early morning lectures. He has been a constant source of support in navigating MIT and the Media Lab, and I wish him the best for his endeavors beyond MIT.

For the many impromptu advice sessions and his useful critical feedback, I would like to thank Esteban Moro. Esteban has always been approachable, kind and insightful, and I have thoroughly enjoyed collaborating and learning from him.

I would like to thank Nikhil Naik, for his constant advice beyond our research collaborations. He has been a long-time collaborator and friend, and has always provided a fresh perspective and given me things to think about.

I am grateful to my group members at the Human Dynamics Lab, and the fellow members of other labs at MIT, who have been immensely helpful through discussions, whiteboard sessions and their companionship. Michiel, Bjarke, Ziv, Matt, Otkrist, Amna and Peter: thank you for being supportive of my work and for your constant feedback.

Next, I would like to thank my friends in Cambridge – Ishaan, Spandan, Mayank and Chetan, thank you so much for making me feel that I had never left home. Our Friday nights of *Settlers of Catan*, *Age of Empires* and obsessive pseudo-philosophy

have been endlessly rejuvenating.

I am eternally grateful to my family – my parents, Pramod and Geeta, my brother, Abhishek, and my grandfather, Jagdish, for their boundless love and support. I am grateful to my mother and father for instilling in me the value of selfless pursuit and hard work, and their numerous sacrifices that have today given me the chance to learn from the very best of the world. My brother is responsible for keeping my feet firmly on the ground, and reminding me constantly to respect the numerous shoulders I stand on to be able to do what I am doing today. My grandfather first taught me the joy of learning and instilled in me the desire for knowledge, which has largely shaped my career choices to date.

Finally, I am immensely grateful to my partner Surabhi. I have learnt so much from her that it is hard to imagine graduate school without her guiding presence. Her love, support and keen sense of judgement have enabled me to constantly take on bigger challenges, grow as a person, and strive for excellence.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

It is immensely intriguing to me how human accomplishments are so much more than the sum of their parts. Automobiles and personal computers are considered commonplace modern tools, and are yet so incredibly sophisticated in their design that it is unimaginable to conceive every aspect of their operation alone in one human lifetime. People collaborate and cooperate in intricate ways, combine diverse sets of skills and knowledge to produce superhuman creations, and we do that almost on a daily basis.

When we consider the dynamics of such large teams of individuals, we often easily overlook the minutiae of management. Design decisions need to be made, a consensus needs to be established, and innovation has to emerge - all of which are typically constrained by an impossible deadline. Yet, through years of cooperation, our collective society has evolved to (often) achieve synergy at scale.

For those of us who wish to understand the science behind this massive collective decision-making, it has been a rough journey. Conducting experiments in the controlled environments of laboratories can only provide preliminary insight into complex human dynamics, and by virtue of being controlled, they are often not representative of the extreme uncertainty of the real world.

In spite of these challenges, understanding collective human behavior remains one of the most important scientific problems of today. Modeling, inferring and learning collective patterns of behavior could lend insights into a number of humanity's press-

ing problems - international conflict, the spread of information, and most importantly, the emergence of cooperation at scale from locally selfish behavior.

Scientists have considered these problems from a variety of different lenses. For instance, physicists create models that are inspired by the dynamics of the universe, statisticians attempt to control uncertainty, economists model rational choice with the mathematics of games, and computer scientists create efficient algorithms that combine some aspects from each of the above.

Outside the realm of academia, applications of research on collective behavior are numerous, from financial systems to satellite navigation, from recommender systems to online advertising and from public policy to social commerce, to name a few. When systems borne out of academic research are deployed, we see that while they are largely successful in predicting average behavior, they are completely non-robust to outliers.

Outlying events come in all shapes and sizes. Consider the 2008 stock market crash, for example. While there had been countless scientific models and industrial systems in use to predict the behavior of the stock market, none could correctly assess the likelihood of the complex sequence of interactions that ended in the worst financial crisis since the Great Depression.

At a smaller scale, consider the phenomenon of viral content on the Internet. Scientists have unearthed that cascades of information sharing from influential sources are one of the leading causes of content becoming viral, and yet there remains significant stochasticity in the process that has not been fully accounted for. It is not true that every bit of information shared by an influential source becomes viral, or that an image which, in its essence conveys almost the same intent as a viral image, necessarily becomes viral either.

Such outcomes of intricate networks of social interactions are ubiquitous, and their complexity gives rise to frequent outlying events that befuddle sophisticated predictive mechanisms. In a future where we imagine humanity collaborating with artificially intelligent agents in synchrony, developing decision-making systems that are cognizant of these outlying events seems like an essential first step.

However, artificial intelligence is as vast as it is deep. There is constant new research day in and day out, whether it is producing the next AlphaGo or maneuvering the next generation of self-driving cars. These are highly sophisticated marvels of engineering and research, and hence modifying them becomes proportionately more challenging. Therefore, as is typical in research, we must start small.

Consider one of the simplest decision-making problems: the allocation game. Imagine you are provided N different magic bags, each of which will give you a gold coin with a different random probability. If you are given a choice to pick one of N different bags, then, without knowing which bag is the best (i.e. the one that is most likely to give you a gold coin), which one do you pick? If you were allowed to play this game over and over again, how would you try out each bag to quickly figure out the best bag to choose from?

This sequential decision-making game, commonly known as the multi-armed bandit problem, is a cornerstone of artificial intelligence and lies at the heart of the problem of general artificial intelligence. It encompasses many everyday tasks such as allocating drugs to patients in clinical trials, providing music recommendations and selecting which stocks to invest in.

Now consider two additional modifications of the problem, the first of which being: All your friends are playing the allocation game, and every time each person makes a choice, they convey to their friends the bag they chose, and whether or not the bag gave them a gold coin.

Now, how would you use these additional observations you obtain from your friends? Should you disregard your friends' communication or should you adjust which bag you choose from based on it? Assume you are aware that your friends have been given different magic bags compared to you. How does that change your policy?

This is an example of a collective decision-making problem, and has applications from wireless communication to data center optimization, to name a few. However, in most existing research, all actors present are assumed to have identical preferences. In a more generalized setting, we have a collection of agents that are each faced with the same fixed set of possible actions to take, however, they have unique preferences for

each of these actions. Each agent sequentially takes actions, and observes a payoff. Even though each agent operates on the same set of actions, it is not necessary, however, that all agents will have identical preferences for each action. An agent might observe that it is receiving a higher payoff by taking a different action from its neighbor, whereas another agent might observe that imitating the actions of its neighbor is more beneficial for them. In this generalized setting, the agents must learn how and when to pay attention to the observations from their neighbors based on their own unique preferences and decision-making.

The online music streaming service Spotify also provides listeners with information about the music their friends from a social networks are listening to, and how they have rated each song. Consider now, that you were to explore a new genre of music. You will have access to numerous recommendations from all your friends, and without knowing which artist you prefer the most, you initially might listen to suggestions from all your friends. As you explore more and more music, you will gradually begin to ignore the recommendations from friends who share different tastes, and consider recommendations from friends whose music you like. This is a classic example of the collective decision-making problem, and numerous other applications exist beyond this one that employ a same underlying structure.

Consider now the second modification to our magic bag game. In addition of bags giving you rewards with some probability, there is a very small chance that choosing any bag will provide you a million coins at once, but also a very small chance that you will lose all your coins by choosing any bag. How does this alter your strategy?

It is fairly obvious that the modification introduced here is the addition of outlying events, which are typically extreme events that usually happen with low probability. This is an example of a heavy-tailed decision-making problem, which is the most common framework for dealing with outlier events. They are named as such because the probability distribution that the events follow have big "tails", which refer to the probability with which an outlying event (e.g. getting a million coins) occurs.

This is the nature of questions that I explore in this thesis, in the context of multi-armed bandit problems. We construct algorithms that efficiently provide optimal

decision-making when an underlying network structure (friends playing the game) or heavy-tailed rewards (million coins at once) are present.

While the narrative henceforth is technical and can often get nuanced, the big picture still remains the same: to eventually create artificial intelligence that can adapt easily to the complex networks of interactions and extreme uncertainty of our world, and I believe this work serves as a preliminary step in this direction.

1.1 Summary of Contributions

With this thesis, we extend methods of sequential decision-making to two important settings that are increasingly becoming relevant as the availability of big data increases: heavy tails and network structure, by making fundamental algorithmic and analytical contributions. Since many complex systems and social systems exhibit heavy-tailed behavior on network structures, we believe that this thesis will serve as an important contribution to the literature of machine learning on these structures, which are summarized below.

Bandits on Heavy-Tailed Distributions

In the problem of stochastic bandits under heavy-tailed reward distributions, we make the following contributions.

1. We create an algorithm titled α-UCB by extending the Robust-UCB of [17] using a truncated mean estimator on bandits with symmetric α-stable rewards, and prove optimal $O(\ln T)$ bounds on its finite-time frequentist regret.

2. Using auxiliary variables, we construct a framework for posterior inference in symmetric α-stable stochastic bandits that leads to the first efficient algorithm for Thompson Sampling in this setting, which we call α-TS.

3. To the best of our knowledge, we provide the first finite-time polynomial bound on the Bayesian Regret of Thompson Sampling achieved by α-TS in this setting.

4. We improve on the regret by proposing a modified Thompson Sampling algorithm, called Robust α-TS, that utilizes a truncated mean estimator, and obtain the first $\tilde{O}(N^{\frac{1}{1+\epsilon}})$ Bayesian Regret in the symmetric α-stable stochastic setting. Our bound matches the optimal bound for $\alpha \in (1, 2)$ (within logarithmic factors).

5. Through a series of experiments, we demonstrate the proficiency of both the frequentist and Bayesian algorithms for symmetric α-stable rewards, which consistently outperform all existing benchmarks.

Multi-Agent Bandits on Networks

The multi-agent bandit problem has largely been explored in the case when agents are all playing the same arms, and also have the same utility (average reward) for each arm. In this case, the agents always co-operate to consistently learn from their neighbors. In our setting, we consider a generalized version of the problem, where, although agents play the same set of arms, they have different utilities (average reward) from each arm. This introduces additional complexity in the decision-making from the agent's perspective, in the form of attention. As it observes the rewards and actions its neighbors take, it must accept or discard their observations based on how similar (or different) its own preferences are from its neighbors.

In this setting of *generalized* multi-agent stochastic bandits with networked communication, we make the following contributions.

1. We propose a generalization of the classic cooperative multi-agent stochastic bandit problem, known as *generalized network bandits*, which bridges problems in multi-agent cooperative bandits with bandit problems with side observations.

2. We propose Net-UCB, a multi-agent upper confidence bound algorithm that solves the generalized bandit problem on sub-Gaussian rewards by constructing probabilistic upper confidence bounds over network observations.

3. We prove an $O(\ln(T))$ bound on the finite-time regret of the Net-UCB algo-

rithm, which is optimal for the stochastic bandit problem. Moreover, we prove that the bound follows an $O(\chi(\bar{G}))$ dependence on the connectivity of the agents (where G is the connectivity matrix, and $\chi(\cdot)$ refers to the chromatic number). This bound is significantly tighter than previous bounds that grow linearly with the number of agents.

4. Through a series of experiemnts, we justify the superior empirical performance of our algorithm over existing state-of-the-art methods.

The thesis is organized as follows: Next, we describe the related work in this problem domain. Subsequently, we describe the multi-armed bandit problem, and several key analytical insights within it. We then describe the statistical treatment of heavy tails, and the concept of concentration of measure, which is critical in our algorithmic framework. In the next chapter, we describe existing algorithms for light-tailed bandit problems and discuss their convergence properties. Chapter 5 covers the first set of contributions: single-agent bandit algorithms on heavy-tailed reward functions, and Chapter 6 provides the development of the generalized network bandit problem, and an efficient algorithm for the same. In chapter 7, we describe a series of experiments and ablations we run to verify the practical performance of our algorithms. We conclude the thesis with final remarks and a brief discussion of possible future research directions.

1.2 Related Work

This thesis lies at the intersection of two largely disparate fields of research in sequential decision-making: robustness and network structure. While robustness in machine learning now is frequently attributed to adversarial robustness [33], we consider the classical interpretation of robustness to uncertainty. In this regard, our work relates to a vast literature spanning decades of work in mathematical statistics, decision-making, inference and machine learning. We discuss these connections in the subsequent sections.

The other aspect of our work relates to network structure in decision-making. As the availability of data and applications of machine learning increase, multi-agent systems become more important in decision-making. This can occur due to a variety of factors – distributed handling and storage of training data might require distributed inference algorithms, or a personalized machine learning system might require a unique agent tuned to each individual. In several modern decision problems, such as online advertising, recommendations and personalized medical diagnosis, such systems are already deployed to large extents, and are created with the underlying network structure in mind. These systems are largely prevalent in the wireless communication, machine learning and web analysis literatures, and in the next sections, we discuss how our work places in connection to this extensive literature.

1.2.1 Machine Learning on Social Networks

Inference on structured data such as social networks has typically been considered a problem in *structured prediction.* Classical machine learning problems are typically variants of a regression (where outcomes are in $\mathbb{R}$) or classification (where outcomes are in a subset of $\mathbb{Z}_+$). *Structured prediction* is a generalized decision problem where the outcome variable are objects, i.e. they are sets whose elements are in $\mathbb{R}$ or $\mathbb{Z}$, and elements are typically correlated.

Structured prediction encompasses several applications of machine learning. In the domain of natural language processing, tasks such as semantic parsing [46, 65, 72] and machine translation [15, 69, 70] are examples of structured prediction problems. In computer vision, the most notable structured prediction problems include semantic segmentation [37, 57, 68], image generation [40] and inters-domain style transfer [92].

There has been a significant amount of work in inference on social networks, such as inferring network structure from network observations [16, 32, 66, 67, 78], creating compressed feature learning for vertices in social networks [41, 71] and link prediction [35, 82, 91].

This thesis, however, looks at network structure from a different lens altogether. We consider decision problems that involve multiple agents that are communicating

on an existing network, which is in contrast to problems discussed earlier, that involve inferring network structure itself. Problems in a similar vein have been discussed in the context of classification when different features are held by different agents [39], when agents are collectively learning concepts [13, 85] and collective reinforcement learning [2, 73].

Multi-Agent Bandit Problems with Network Structure

The multi-agent multi-armed bandit problem has been analysed under a variety of different lenses. Most of the analysis comes from the literature of distributed control, where algorithms focus on distributed settings with communication between agents over a network. In this setting, there are a few typical variants of the problem that have been explored:

- **Distributed Cooperative Bandits.** In this setting, $M > 1$ agents are connected over a network, and are faced with identical decision problems. The agents communicate either via a consensus protocol [49, 50], or a local communication protocol [51]. The communication allows for improved collective performance.

- **Multi-Agent Cooperative Bandits without Collisions.** In this setting, $M > 1$ agents are connected over a network, and have unique decision problems. The agents communicate via their communication network, but actions are taken collectively to minimize the group regret [75].

- **Multi-Agent Competitive Bandits with Collisions.** In this problem, $M > 1$ agents have access to the same K arms, however, agents can pull only unique arms during each iteration to obtain rewards. This setting has numerous applications in channel selection in distributed communication [53–56, 58–60].

- **Bandits with Side Observations.** This problem focuses on the extension of the classical stochastic bandit setting in which, in addition to obtaining a reward from the arm pulled, the agent obtains rewards from neighboring arms, and the

set of arms assumes an underlying network structure [6,7,20,21,25,36,87]. While this problem does not focus on exactly the setting we are considering, there are several insights that we draw from it, since our generalized communication protocol imposes a network structure on arms (See Chapter 6 for details).

1.2.2 Machine Learning with Heavy-Tailed Losses

Heavy-tailed rewards and loss functions are increasingly getting more prominence in machine learning [30, 31, 44]. Extensions of common machine learning algorithms such as least-squares regression [44] and classification via margin-based techniques [30] have been analysed using a variety of robust estimation techniques, most common of which being the median-of-means estimators.

The difficulty in optimization for machine learning problems is compounded in the presence of heavy-tailed loss functions, and is also an area of active study [31, 42, 88]. One of the typical assumptions made in classical machine learning problems (that provide provable learning guarantees) is that of bounded tail functions, specifically considering random variables that have finite moment generating functions. This enables a tight analysis by using concentration results derived using Chernoff's technique [43], which is not possible when losses are heavy-tailed. This results in weaker convergence and generalization guarantees that grow polynomially instead of logarithmically, and the typical approach is to truncate losses to ensure logarithmic growth.

In more empirical settings, these assumptions of light-tailed behavior are somewhat implicit, yet massive amounts of training data available today provide remarkable practical performance. While some methods have been proposed to handle data with heavy tails [83, 84, 89], these are largely guided by intuition or the generation of synthetic training data and rarely by techinques involving truncation or robust estimation (and its accompanying analysis), which leads us to believe that there is a lot of room for principled improvements in these domains.

Stochastic Bandits under Heavy Tailed Rewards

Heavy-tailed rewards have been studied in the context of bandit problems as well. A version of the UCB algorithm [10] has been proposed in [17] coupled with several robust mean estimators to obtain Robust-UCB algorithms with optimal *problem-dependent* (i.e. dependent on individual μ_ks) regret when rewards are heavy-tailed. However, the optimal version of their algorithm has a few shortcomings : first, their algorithm requires the median-of-means estimator, which has a space complexity of $O(\log T)$ and time complexity of $O(\log \log T)$ per update. This makes the algorithm expensive as the confidence is increased (for accurate prediction). Second, their regret bound has a dependence of $(\mu^* - \mu_k)^{1/(1-\alpha)}$, which can become arbitrarily large for a stable distribution with α close to 1. Finally, there is no mechanism to incorporate prior information, which can be advantageous, as seen even with weak priors. Alternatively [81] introduce a deterministic exploration-exploitation algorithm, which achieves same order regret as Robust-UCB for heavy-tailed rewards. While these algorithms do enjoy strong regret bounds, as is with the frequentist approach, there is little room to incorporate prior knowledge, which is extremely beneficial in practical settings.

The Bayesian route for bandits incorporates prior knowledge quite succintly via the general framework of Thompson Sampling [80]. There has been work in analysing Thompson Sampling for specific Pareto and Weibull heavy-tailed distributions in [47], however, they do not provide an efficient posterior update rule and rely on approximate inference under the Jeffrey's prior, which, by definition, is un-informative. More importantly, the Weibull and Pareto distributions typically have "lighter" tails owing to the existence of more higher order moments, and hence cannot typically be used to model very heavy tailed signals.

In related problems, [90] provide a purely exploratory algorithm for best-arm identification under heavy-tailed rewards, using a finite $(1+\epsilon)^{th}$ moment assumption. Similarly, [63, 76] explore heavy-tailed rewards in the linear bandit setting.

In the next section, we begin an overview of the key mathematical concepts in-

volved in this thesis, starting with the multi-armed bandit framework.

Chapter 2

The Multi-Armed Bandit Framework

2.1 Introduction

The multi-armed bandit problem has a long history in the science of decision-making. It was first introduced by William R. Thompson [80] in 1933, in the statistics journal *Biometrika*. Thompson was interested in understanding the worst-case outcome of running medical trials blindly, without any attempt to adjust the allocation of treatments as the drug in question appeared more or less effective.

Later, in the 1950s, statisticians Frederick Mosteller and Robert Bush [23] coined the term "bandits" in their quest to study animal learning using trials on mice. The mice were tasked with choosing to go right or left in a T-shaped structure, one direction of which contained food. They extended this study in humans by constructing a two-armed machine (the bandit) that would give a random payoff every time one of its arms was pulled. The name paid homage to the original slot machine, which was called the one-armed bandit, since they typically were designed to steal money from unwitting participants.

Since then, the applications and variations of bandit problems have grown manifold. At its core, it represents the fundamental problem of simultaneous learning and decision-making under uncertainty, and many important practical decision problems can be stripped down to essentially an instance of a multi-armed bandit problem. The design of clinical trials, which we have already mentioned, involves the medical

practitioner making the choice of drug they administer to a patient. The dilemma here is between *exploration* and *exploitation*: a drug that has been effective in most patients can be administered (exploitation), or a drug that has not been tested much but could be potentially more effective could also be administered (exploration). Deciding when and how much to explore versus exploit is at the heart of this problem, and is what makes it interesting as a research problem.

While the bandit framework and associated algorithms are yet to be adopted in actual clinical trials, there are several problems where bandit algorithms are readily deployed in practice. For instance, a bandit algorithm is used in placing ads online [79] and content recommendation [53]. A bandit algorithm is present in Monte-Carlo Tree Search (MCTS), which played an important role in DeepMind's AlphaGo [45].

Apart from its applications, the mathematics of bandit problems has strong connections with other fields of mathematics, including optimization, probability theory, statistics, complexity theory and information theory, which make it a very exciting problem to work on. This has led to a significant recent increase in research interest on the bandit problem, and we hope to see it only increase as time progresses.

2.2 Stochastic Multi-Armed Bandits

While there are numerous versions of the multi-armed bandit problem, including linear bandits [1], contextual bandits [3] and Lipschitz bandits [19], we restrict ourselves in this thesis to the fundamental multi-armed bandit problem, known as the stochastic multi-armed bandit.

In any instance of the K-armed stochastic bandit problem, there exists an agent with access to a set of K actions (or "arms"). The problem proceeds in rounds, indexed by $t \in [1, T]$. The total number of rounds, known as the *time horizon* T, is known in advance. For each round $t \in [T]$:

1. The agent picks arm $a_t \in [K]$.

2. The agent observes reward $r(t)$ from that arm.

The agent's objective is to pull arms $a_1, ..., a_T$ such that it obtains the largest cumulative reward $\sum_{t=1}^{T} r(t)$. For arm $k \in [K]$, rewards come from a distribution $\mathcal{D}_k$ with mean $\mu_k = \mathbb{E}_{\mathcal{D}_k}[r]$. The largest expected reward is denoted by $\mu^* = \max_{k \in [K]} \mu_k$, and the corresponding arm(s) is denoted as the *optimal* arm(s) k^*. In our analysis, we will focus exclusively on the i.i.d. setting, that is, for each arm, rewards are independently and identically drawn from $\mathcal{D}_k$, every time arm k is pulled.

2.2.1 Regret

To create algorithms for the bandit problem, we must first create a metric to measure performance. A natural metric would be its cumulative obtained reward $\sum_{t=1}^{T} r(t)$. However, since the rewards themselves are randomly drawn from distributions, it is difficult to obtain strong guarantees for any individual trial. This is because even if an algorithm knew the optimal arm (the arm with the highest mean reward) *a priori*, it is possible, with very low probability, to obtain a very small cumulative reward. Hence, it is more feasible to analyse the average cumulative reward, given by $\mathbb{E}[\sum_{t=1}^{T} r(t)]$, where the average is taken over each of the distributions $\mathcal{D}_k$.

As it turns out, a more convenient measure of performance is *Regret* $R(T)$, which, at any round T, is the difference of the cumulative mean reward of the algorithm against the expected reward of always playing an optimal arm.

$$R(T) = \mu^* T - \mathbb{E}[\sum_{t=1}^{T} r(t)] = \mu^* T - \sum_{t=1}^{T} \mu_{a_t} \tag{2.1}$$

This metric is intuitive: it measures how bad the average performance of the algorithm is compared to the best possible algorithm. Instead of maximizing the cumulative reward, we can now aim to minimize regret. An alternative way to represent regret is as follows.

Lemma 1. *(Regret Decomposition) For any algorithm choosing actions $a_1, ..., a_T$ over T rounds in a K-armed stochastic bandit problem (for finite K), the regret at T rounds*

satisfies:

$$R(T) = \sum_{k=1}^{K} \Delta_k \cdot \mathbb{E}[n_k(T)] \tag{2.2}$$

Here, $n_k(t)$ is the number of times arm k has been pulled until time t, and $\Delta_k = \mu^* - \mu_k$. Typically, when analysing an algorithm, we will attempt to control the expected number of times an algorithm pulls a suboptimal arm, since each pull of a suboptimal arm will contribute $\mu^* - \mu_k$ to the cumulative regret. $\mathbb{E}[n_k(T)]$ is also known as the *suboptimality gap* or *action gap*, and controlling this quantity has been the basis of almost all analysis of the bandit problem.

Proof. Let the agent pull arms $a_1, ..., a_T$ over T rounds, and obtain rewards $r(1), ..., r(T)$. We know that:

$$\begin{aligned}
R(T) &= \mu^* T - \sum_{t=1}^{T} \mu_{a_t} \\
&= \sum_{k=1}^{K} \sum_{t=1}^{T} \mathbb{E}\left[(\mu^* - r(t))\mathbf{1}\left\{a_t = k\right\}\right] \\
&= \sum_{k=1}^{K} \sum_{t=1}^{T} (\mu^* - \mu_k)\mathbb{E}\left[\mathbf{1}\left\{a_t = k\right\}\right] \\
&= \sum_{k=1}^{K} (\mu^* - \mu_k)\mathbb{E}\left[\sum_{t=1}^{T} \mathbf{1}\left\{a_t = k\right\}\right] \\
&= \sum_{k=1}^{K} (\mu^* - \mu_k)\mathbb{E}\left[n_k(T)\right].
\end{aligned}$$

□

2.2.2 Lower Bounds on Regret

Before constructing algorithms for this problem it is important to understand our computational limits. Intuitively, if an agent has no knowledge of the problem, it must pull each arm once, which provides us a naive lower bound of $\sum_{k=1}^{K} \Delta_k$. A much stronger lower bound on the regret has been provided by Lai and Robbins in

their seminal paper from 1985 [48], and generalized subsequently by Burnetas and Katehakis [22]. Before we discuss this bound, it is important to distinguish between "unreasonable" and "reasonable" policies. For instance, if an algorithm always plays arm 1, then it will incur zero regret if arm 1 is indeed optimal, but an infinite regret as $T \to \infty$ if it is not. It is not possible to obtain a meaningful lower bound on the best-case performance of such an algorithm. Hence, we define *consistent* algorithms (policies) that do not behave "unreasonably" as follows.

Definition 1. *An algorithm is* ***consistent*** *over a class of unstructured bandits $\mathcal{B}$ if for all members $b \in \mathcal{B}$, and all $p > 0$ we have:*

$$\lim_{T \to \infty} \frac{R(T, b)}{n^p} = 0. \tag{2.3}$$

This definition excludes algorithms such as the one discussed above, that have growing regret on certain instances within a class of unstructured bandits $\mathcal{B}$. For all arms $k \in K$, let the joint set of all possible reward distributions across all arms come from a set $\mathcal{B}$. If $\mathcal{B}$ admits a factorization of the form $\prod_{k \in [K]} \mathcal{M}_k$, where $\mathcal{M}_k$ is the set of all possible reward distributions for arm k, then we denote such a class of bandit problems $\mathcal{B}$ as unstructured, and structured if it is not[1]. Armed with this terminology, we can now state the lower bound provided by Burnetas and Katehakis.

Theorem 1. *(Regret Lower Bound for Consistent Policies [22]) Let $\mathcal{B}$ be a class of unstructured bandits and π be a consistent policy (algorithm) over $\mathcal{B}$. Then, for all $\nu = (P_i)_{i=1}^k \in \mathcal{B}$ such that $\mu_i = \mathbb{E}_{X \sim P_i}[X] < \mu^*$, it holds that*

$$\liminf_{T \to \infty} \frac{R(T)}{\ln(T)} = \sum_{k : \Delta_k > 0} \frac{\Delta_k}{d_{\inf}(P_i, \mu^*, \mathcal{M}_i)}, \quad \textit{where,} \tag{2.4}$$

$$d_{\inf}(P, \mu^*, \mathcal{M}) = \inf_{P' \in \mathcal{M}} \{\mathbb{D}_{\mathsf{KL}}(P, P') : \mu(P') > \mu^*\} \tag{2.5}$$

[1]It is not true that a factorization of this form will always exist – in the case when arms are correlated, for instance, we cannot factor out the sets of distributions for arms with their individual sets, and we would have to represent them via their joint set.

We refer the reader to [22] for a complete proof of this theorem. For most parameterized families of reward distributions (exponential families, etc.) d_{inf} is equivalent to the parameterized KL-divergence between the reward distributions. Hence, we see that asymptotically, for such reward distributions, any algorithm must incur a regret of $\Omega(\ln T)$. In this thesis, we design algorithms that obtain regret within a constant factor of this lower bound.

2.3 Multi-Agent Stochastic Bandits

Thus far, we have considered the bandit problem in context of only one agent. We can extend this problem to the setting where we have several agents solving bandit problems simultaneously. In this setting, it is possible for agents to share information between them, and obtain better performance than the case when they operate individually.

Consider $M > 1$ agents $\{1, ..., M\}$, where, for each agent $m \in [M]$, we have an individual stochastic bandit problem. We extend the notation from the previous section: let μ_k^m denote the mean reward for arm $k \in [K]$, and μ_*^m denote the optimal mean reward, and $\Delta_k^m = \mu_*^m - \mu_k^m$ for agent m and arm k. Each agent thus solves an individual multi-armed bandit problem over K arms.

It is important to note that for all problems on unstructured bandits with finite arms, this formulation is fairly general. We can allow each agent m to have its own set of $\mathcal{K}_m$ arms from a superset of $\mathcal{K}$ arms (such that $|\mathcal{K}| = K$) and extend all current algorithms trivially, by setting $\mu_k^m = -\infty$ for each arm $k : k \notin \mathcal{K}_m$. The algorithms can then be modified to ensure that they do not pull any of these arms at all, and we will obtain identical regret bounds.

Additionally, at this time, we do not impose any constraint on the reward distributions across agents (i.e. make them identical etc.), although to obtain improvements, we will have to assume structure across agents.

2.3.1 Group Regret

A natural extension of the single-agent regret is the *Group Regret*, $R_G(T)$, which can be defined analogously.

$$R_G(T) = \sum_{m=1}^{M} \sum_{k=1}^{K} \Delta_K^m \mathbb{E}[n_k^m(T)]. \tag{2.6}$$

$n_k^m(T)$ denotes the number of times arm k has been pulled by agent m until time T. A natural extension of the result in [22] can be applied to obtain a lower bound over the group regret over exponential family unstructured multi-agent bandits.

Corollary 1. *Lower Bound on Group Regret] For a series of consistent policies $\pi_1, ..., \pi_M$ over an unstructured M-agent K-armed bandit problem, the group regret satisfies:*

$$\liminf_{T\to\infty} \frac{R_G(T)}{\ln T} \geq \sum_{m=1}^{M} \sum_{k=1}^{K} \left(\frac{\min_{m\in[M]} \Delta_k^m}{\max \mathbb{D}_{\mathsf{KL}}(P_m^k || P_{k_m^*}^m)} \right) + o(M). \tag{2.7}$$

P_k^m refers to the probability distribution of rewards for arm k in agent m, and k_m^* denotes the optimal arm for agent m. This corollary can be proved trivially by simply summing up the lower bound for the single-agent case over all M agents. We would like to design algorithms that obtain regret of similar order as the lower bound for optimal performance. In the next sections, we describe some fundamental algorithmic concepts that we will require to build new algorithms in these settings.

Chapter 3

Heavy-Tailed Distributions

3.1 Tail Probabilities

In the stochastic bandit problem, the optimal arm is the one with the largest mean reward. Since we do not know the mean for any arm in advance, it is typically estimated from the rewards obtained, and is then used to take actions. It is then essential to understand how good of an estimate we have of the true mean from the sample mean. For N i.i.d. samples $X_1, ..., X_N$ of a random variable X with mean μ, the empirical mean is given by

$$\hat{\mu} = \frac{1}{N} \sum_{i=1}^{N} X_i.$$

We wish to know how this quantity relates with the true mean μ, i.e. as N increases, how the difference between the two means ε decays as a function of the number of observations N:

$$|\hat{\mu} - \mu| \leq \varepsilon(N)$$

. To understand this decay, we are interested in the tails of the distribution. For some $\varepsilon > 0$, we are concerned with the following probabilities:

$$\Pr\left(\hat{\mu} \geq \mu + \varepsilon\right) \text{ and } \Pr\left(\hat{\mu} \leq \mu - \varepsilon\right).$$

These are known as the concentrations of the mean. Hence, finding bounds for these concentration probabilities is an essential step in the analysis and design of optimal algorithms for the bandit problem.

3.1.1 Chernoff Bounds

The most powerful technique to obtain bounds for deviations is the Chernoff technique. For a random variable X with mean 0, consider the **cumulant generating function** $\psi(t) = \log M_X(t)$, where $M_X(t)$ is the moment generating function of X, given by

$$M_X(t) = \mathbb{E}[\exp(tX)]. \tag{3.1}$$

We can then obtain a bound on the concentration by applying Markov's Inequality to $\psi(t)$. For all $\varepsilon > 0$,

$$\Pr(|X| \geq \varepsilon) \leq \inf_t \left\{\exp\left(\psi(t) - t\varepsilon\right)\right\}. \tag{3.2}$$

If the cumulant generating function is finite, by bounding the RHS of the above equation, we can obtain tight bounds on the concentration. In the next section, we cover a few families of distributions that allow for such concentration results.

3.2 sub-Gaussian Distributions

Consider the case of random variables X that, for some $\sigma > 0$ have the property

$$M_X(t) \leq \exp\left(\frac{\sigma^2 t^2}{2}\right). \tag{3.3}$$

These distributions have tails that decay faster than Gaussian distributions, and are known as **sub-Gaussian** distributions. For these distributions, we have the following property of concentration.

Lemma 2. *(sub-Gaussian Concentration) Let X be a σ-sub-Gaussian random variable with mean 0. Then, with probability at least $1-\delta$ $\forall \delta > 0, \varepsilon > 0$,*

$$|X| \leq \sigma\sqrt{2\ln(\frac{1}{\delta})}.$$

This is a central property that we will utilize in the analysis of problems involving light-tailed (specifically sub-Gaussian) distributions.

Proof.

$$\begin{aligned}\Pr\left(|X| \geq \varepsilon\right) &\leq \inf_t \left\{\exp\left(\psi(t) - t\varepsilon\right)\right\} \\ &\leq \inf_t \left\{\exp\left(\frac{\sigma^2 t^2}{2} - t\varepsilon\right)\right\}\end{aligned}$$

Setting $t = \frac{\varepsilon}{\sigma^2}$,

$$\leq \exp\left(-\frac{\varepsilon^2}{2\sigma^2}\right).$$

By setting the LHS as δ, we obtain the proof. □

3.3 Heavy-Tailed Distributions

In the previous settings we looked at providing a rate of concentration by upper bounding the moment generating function. However, when random variables exhibit tails that decay slowly enough that they do not admit a finite moment generating function at all. In this case, Chernoff's method for obtaining concentration is inapplicable.

Definition 2. *(Heavy-Tailed Random Variables) Let X be a random variable with a probability density function $p(\cdot) : \mathbb{R} \to [0,1]$. X is heavy-tailed if for all $t > 0$,*

$$\int_{-\infty}^{\infty} e^{tx} p(x) = \infty.$$

A natural consqeuence of this definition of heavy-tailed random variables is that they do not admit exponential concentration around the mean, and we must resort to alternate strategies to estimate the mean with few samples. This behavior is expected, since for heavy-tailed distributions, the probability of observing an extremely large sample is much higher than it would be for a sub-Gaussian distribution. Therefore, even a single outlier would throw off the estimate for the mean significantly.

While the bandit problem has been analysed for several heavy-tailed random variables from exponential families [24,47] and having finite p^{th} moments ($p > 2$) [17], the extremely heavy-tailed class of densities known as α-stable densities have yet to be understood in the context of stochastic bandits. In this thesis, we focus exclusively on α-stable bandits, and describe the family of distributions in the next section.

3.4 α-Stable Distributions

α-Stable distributions, introduced by Lévy [52] are a class of heavy-tailed probability distributions defined over $\mathbb{R}$ whose members are closed under linear transformations.

Definition 3 (α-Stable Random Variables [12]). *Let X_1 and X_2 be two independent instances of the random variable X. X is **stable** if, for $a_1 > 0$ and $a_2 > 0$, $a_1X_1 + a_2X_2$ follows the same distribution as $cX + d$ for some $c > 0$ and $d \in \mathbb{R}$.*

A random variable $X \sim S_\alpha(\beta, \mu, \sigma)$ follows an α-stable distribution described by the following parameters:

- $\alpha \in (0, 2]$ (characteristic exponent). This parameter controls the decay of the tails of the distribution.
- $\beta \in [-1, 1]$ (skewness). This controls the skew of the distribution around its median. $\beta = -1$ implies that all mass is in the negative half-plane, and $\beta = 1$ implies that all mass is at the positive half-plane.
- $\mu \in \mathbb{R}$ (shift). This controls the mean of the distribution, if it exists.

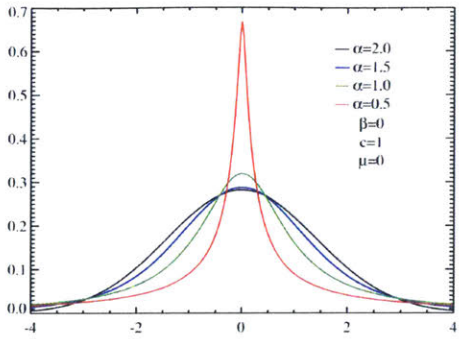

Figure 3-1: Sample probability density for standard ($\mu = 0, \sigma = 1$) symmetric α-stable distributions with various values of α [86].

- $\sigma \in \mathbb{R}^+$ (scale). This corresponds to a scaled variance parameter. While the variance is not defined for any α-stable density, this analog controls the scale at the α-exponent, instead of the 2-exponent (as in the case of variance).

While it is not possible to analytically express the density function for generic α-stable distributions, they are known to admit the characteristic function $\phi(x; \alpha, \beta, \sigma, \mu)$:

$$\phi(x; \alpha, \beta, \sigma, \mu) = \exp\left\{ix\mu - |\sigma x|^{\alpha}\left(1 - i\beta \operatorname{sign}(x)\Phi_{\alpha}(x)\right)\right\},$$

where $\Phi_{\alpha}(x)$ is given by

$$\Phi_{\alpha}(x) = \begin{cases} \tan(\frac{\pi\alpha}{2}) \text{ when } \alpha \neq 1, \\ -\frac{2}{\pi}\log|x|, \text{ when } \alpha = 1 \end{cases}$$

For fixed values of α, β, σ and μ we can recover the density function from $\phi(\cdot)$ via the inverse Fourier transform:

$$p(z) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \phi(x; \alpha, \beta, \sigma, \mu)e^{-izx}dx$$

Most of the attention in the analysis of α-stable distributions has been focused on the stability parameter α, which is responsible for the tail "fatness". It can be shown

that asymptotically, the tail behavior ($x \to \pm\infty$) of $X \sim S_\alpha(\beta, \mu, \sigma)$ follows [12]:

$$f(x) \sim \frac{1}{|x|^{1+\alpha}} \left(\sigma^\alpha (1 + \text{sgn}(x)\beta) \sin\left(\frac{\pi\alpha}{2}\right) \frac{\Gamma(\alpha+1)}{\pi} \right)$$

where $\alpha < 2$ and $\Gamma(\cdot)$ denotes the Gamma function. The power-law relationship admitted by the density is responsible for the heaviness of said tails.

Lemma 3 (Moments of α-Stable Density [12]). *$X \sim S_\alpha(\beta, \mu, \sigma), \alpha < 2$ admits a moment of order λ only when $\lambda \in (-\infty, \alpha)$. For $\alpha = 2$, it admits all moments of orders in $(0, 2]$.*

From Lemma 3 it follows that α-stable random variables only admit a finite mean for $\alpha > 1$, and also admit a finite variance only when $\alpha = 2$, which corresponds to the family of Gaussian distributions. In the case of the bandit problem, we are concerned with the regret, which is an expectation. Hence, to continue with existing measures of regret, we restrict our analysis hence to α-stable distributions only with $\alpha > 1$. Note that for all our discussions, $1 < \alpha < 2$, hence, all distributions examined are heavy-tailed, with infinite variance. Additionally, we restrict ourselves to only symmetric ($\beta = 0$) distributions: asymmetric distributions do not allow a scaled mixture representation (which is the basis of our framework, see Section 5.2.1).

3.4.1 Sampling from α-Stable Densities

For general values of $\alpha, \beta, \sigma, \mu$, it is not possible to analytically express the density of α-stable distributions, and hence we resort to using auxiliary variables for sampling. The Chambers-Mallows-Stuck [28] algorithm is a straightforward method to generate samples from the density $S_\alpha(\beta, 1, 0)$ (for $\alpha \neq 1$) via a non-linear transformation of a uniform random variable V and an exponential random variable W, which can then be re-scaled to obtain samples from $S_\alpha(\beta, \sigma, \mu)$ (Algorithm 1).

Algorithm 1 Chambers-Mallows-Stuck Generation

Input: $V \sim U(-\pi/2, \pi/2), W \sim E(1)$

Output: $X \sim S_\alpha(\beta, \sigma, \mu)$

Set $B_{\alpha,\beta} = \arctan(\beta \tan(\pi\alpha/2))\alpha^{-1}$

Set $S_{\alpha,\beta} = (1 + \beta^2 \tan^2(\pi\alpha/2))^{1/(2\alpha)}$

Set $Y = S_{\alpha,\beta} \times \frac{\sin(\alpha(V+B_{\alpha,\beta}))}{\cos(V)^{1/\alpha}} \times \left(\frac{\cos(V-\alpha(V+B_{\alpha,\beta}))}{W}\right)^{\frac{1-\alpha}{\alpha}}$

return $X = \sigma Y + \mu$.

3.4.2 Properties of α-Stable Densities

In our algorithm and analysis, we will be requiring several elementary mathematical properties of α-stable densities. We outline them as follows.

Lemma 4 (Sums of Symmetric α-Stable Densities [12]). *If $X \sim S_\alpha(0, \sigma_1, \mu_1)$ and $Y \sim S_\alpha(0, \sigma_2, \mu_2)$, then $X + Y \sim S_\alpha(0, \sigma, \mu)$, where,*

$$\sigma = (\sigma_1^\alpha + \sigma_2^\alpha)^{1/\alpha}, \mu = \mu_1 + \mu_2.$$

We see that the sums behave similar to Gaussian random variables, with the resulting scale parameter σ being an L_α-norm of the vector of scales of the components. The next property we discuss is behavior under linear transformations.

Lemma 5 (Linear Transformations of α-Stable Densities [12]). *If $X \sim S_\alpha(0, \sigma, \mu)$, then for $a \neq 0, b \in \mathbb{R}$,*

$$aX + b \sim S_\alpha(0, |a|\sigma, a\mu + b).$$

It is crucial to note that the above property is valid only for symmetric ($\beta = 0$) densities. If $\beta \neq 0$, then the behavior of a linear transformation differs [12]. Finally, we discuss the product property.

Lemma 6 (Products of α-Stable Densities [12]). *Let Z and $Y > 0$ be independent random variables such that $Z \sim S_\gamma(0, \sigma_1, \mu_1)$ and $Y \sim S_\delta(1, \sigma_2, \mu_2)$. Then $ZY^{1/\gamma}$ is stable with exponent $\gamma\delta$.*

We refer the reader to [12] for detailed proofs for all these elementary properties. In the next section, we will discuss the concentration measures derived from these properties of α-stable distributions.

3.4.3 Concentration of Measure

As we remarked in the beginning of this chapter, the quantity of interest is how the empirical mean concentrates around the true mean as the number of samples grows. In this section, we derive a few results in this direction. We begin with the empirical mean:

Lemma 7 (Scale of Empirical Mean in Symmetric α-Stable Densities). *Let $X_1, X_2, ..., X_n$ be an i.i.d. sequence of n random variables following $S_\alpha(0, \sigma, \mu)$, and $\bar{X}$ denote the empirical mean:*

$$\bar{X} = \frac{1}{n}\sum_{i=1}^{n} X_i$$

Then $\bar{X} \sim S_\alpha(0, \sigma^*, \mu^*)$, where,

$$\sigma^* = \sigma n^{(\frac{1}{\alpha}-1)}, \mu^* = \mu$$

This result characterizes the dependence of the number of samples on the empirical mean. In contrast to a Gaussian distribution, where the variance of the empirical mean decays as $O(n^{-1/2})$ in the number of samples, here we observe an $O(n^{1/\alpha-1})$ decline. This matches the Gaussian rate for $\alpha = 2$, but is slower otherwise. The above lemma can be proved by invoking elementary properties of the α-stable density.

Proof. Let $Y = \sum_{i=1}^{n} X_i$. Then $Y \sim S_\alpha(\sigma n^{1/\alpha}, \mu)$, by Lemma 4. Then, $\bar{X} = Y/n$, and the result follows from Lemma 5. □

Next, we state an important result that allows us to obtain the moments of symmetric, zero-mean α-stable distributions.

Lemma 8 (Theorem 4 of [77], p-Moment of α-Stable Densities). *For* $X \sim S_\alpha(0, \sigma, 0), p \in (0, \alpha)$,

$$\mathbb{E}[|X|^p] = C(p, \alpha)\sigma^{\frac{p}{\alpha}} \text{ where, } C(p, \alpha) = \frac{2^{p+1}\Gamma(\frac{p+1}{2})\Gamma(-p/\alpha)}{\alpha\sqrt{\pi}\Gamma(-p/2)}, \text{ and } \Gamma(x) = \int_0^\infty t^{x-1}e^{-t}dt.$$

This result provides a closed-form representation of the p^{th}-moment of α-stable densities ($p \in (0, 2)$). Note the $O(\sigma^{p/\alpha})$ dependence on the scale, which is crucial to the result of analysis. For the full proof, we refer readers to Theorem 4 of [77]. The next two lemmas allow us to obtain a concentration bound of the empirical of a finite number of samples of a symmetric α-stable distribution.

Lemma 9 (Lemma 3 of [17], Polynomial Concentration). *Let* $X_1, ..., X_n$ *be an i.i.d. sequence of random variables with finite mean* μ, *finite* $(1 + \epsilon)$ *centered moments* $\mathbb{E}[|X - \mu|^{1+\epsilon}] \leq v_\epsilon$ *and finite raw moments* $\mathbb{E}[|X|^{1+\epsilon}] \leq u_\epsilon$, *for* $\epsilon \in (0, 1]$. *Let* $\hat{\mu}$ *denote the empirical mean:*

$$\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} X_i$$

Then, for any $\delta \in (0, 1)$, *we have, with probability at least* $1 - \delta$,

$$\hat{\mu} \leq \mu + \left(\frac{3v_\epsilon}{\delta n^\epsilon}\right)^{\frac{1}{1+\epsilon}}$$

For a full proof of this lemma, we refer readers to Lemma 3 of [17]. We can see that, as expected, we obtain a polynomial rate of concentration. This is expected from the nature of the distribution itself, and cannot typically be improved without some form of truncation.

Lemma 10. *Let* $X_1, ..., X_n$ *be an i.i.d. sequence of random variables following* $S_\alpha(0, \sigma, \mu)$ *for* $\alpha \in (1, 2)$. *Let* $\hat{\mu}$ *denote the empirical mean:*

$$\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} X_i$$

Then, for any $\delta \in (0,1)$ *and* $\epsilon \in (0, \alpha - 1)$*, we have, with probability at least* $1 - \delta$*,*

$$|\hat{\mu} - \mu| \leq \sigma^{\frac{1}{\alpha}} \left(\frac{2C(1+\epsilon, \alpha)}{\delta n_k(t)^{\epsilon}} \right)^{\frac{1}{1+\epsilon}}$$

Proof. From Lemma 5, we know that if $X \sim S_\alpha(0, \sigma, \mu)$ then $X - \mu \sim S_\alpha(0, \sigma, 0)$. Applying this in Lemma 8, we get $v_\epsilon = C(1+\epsilon, \alpha)\sigma^{\frac{1+\epsilon}{\alpha}}$. Note that the raw moments of order $1+\epsilon < \alpha$ are also finite for α-stable densities, therefore $u_\epsilon < \infty \; \forall \epsilon \in (0, \alpha - 1)$ and Lemma 9 can be applied. Thus, an application of Lemma 9 for both tails with probability $\delta/2$ gets the desired result. □

A few remarks are in order. First, we note that this bound is tight, and while this introduces polynomial deviations for the empirical mean, it is not an artefact of the proof method, but the heaviness of the tails themselves [17].

Now that we have covered some essential results in the concentration of measure, we proceed to provide some detailed development of common types of bandit algorithms.

Chapter 4

Bandit Algorithms

In this chapter, we discuss typical single-agent bandit algorithms for unstructured bandits, which will be the basic heuristic we will utilize in this thesis. As discussed previously, the crucial dilemma in the bandit problem is the tradeoff between exploration and exploitation. Hence, when designing algorithms, it is crucial to balance exploration and exploitation as time progresses for optimal performance.

Additionally, another concern is the time complexity of the algorithm - we would desire algorithms that do not grow significantly in complexity as the number of trials increases, or as the number of arms increases. While there are several heuristics that have been proposed for the stochastic bandit problem, in this thesis, we consider the major two heuristics that form the basis of our technique in heavy-tailed and networked settings respectively.

The two primary heuristics we will consider are somewhat different in their formulation. The first heuristic (Upper Confidence Bound) is based on optimism, which enables an agent to explore an arm more if it does not have enough knowledge about the mean reward for that arm. The second heuristic (Thompson Sampling) is in fact the oldest heuristic for the bandit problem, first proposed in 1933 [80]. It proceeds via a Bayesian formulation, with prior distributions set over the parameters of each arm.

4.1 Optimism in the Face of Uncertainty

The Optimism in the Face of Uncertainty (OFU) heuristic was proposed in the seminal work of Auer *et. al.* [10] in the context of stochastic bandits with bounded rewards, however, the proposed algorithm with its regret guarantees holds for generally sub-Gaussian reward distributions. The algorithm, known as Upper Confidence Bound (UCB), operates as follows.

We first allow an 'initialization phase', in which, the agent selects each arm once. This allows the agent to obtain a basic confidence interval (or concentration) of the mean reward around its mean. For each trial of the problem then onwards ($t \in (K, T]$), the agent then computes, for each arm k

$$Q_k(t) = \frac{\sum_{i=1}^{n_k(t-1)} r_k^i}{n_k(t-1)} + \sqrt{\frac{2\ln(t-1)}{n_k(t-1)}}. \tag{4.1}$$

r_k^i represents the i^{th} reward obtained by the agent for the k^{th} arm. It can be seen that $Q_k(t)$ is the sum of the empirical mean reward for arm k up to time $t-1$, and another term that is inversely proportional to $\sqrt{n_k(t-1)}$. The agent then selects the arm with the highest value of $Q_k(t)$.

$$a_t = \arg\max_{k \in [K]} Q_k(t). \tag{4.2}$$

If we inspect $Q_k(t)$ we see an interesting tradeoff encoded within the formulation. The second term $\sqrt{\frac{2\ln(t-1)}{n_k(t-1)}}$ controls the amount of exploration the agent will do for arm k. For instance, when for an arm k, $n_k(t-1)$ is small, then this second term will be large, leading to a larger $Q_k(t)$. When $n_k(t-1)$ is large, then the "exploration" bonus introduced is smaller, and the agent will typically choose the arm with the largest empirical mean reward, since it can now trust the empirical mean estimate with more confidence. Since the agent is 'optimistic' of the estimate of the mean initially, this heuristic is known as optimism under uncertainty.

One may question the specific form of the uncertainty term. However, it can be seen from the analysis of concentration, that the term is chosen such that the

algorithm obtains optimal regret. Recall Lemma 2, which introduces a bound of a similar form, which is essential for the design of the algorithm and its associated analysis. We state the regret result more formally as follows.

Theorem 2 (Regret of sub-Gaussian UCB [10]). *For a K-armed unstructured stochastic bandit with rewards for arm $k \in [K]$ drawn i.i.d. from σ_k-sub-Gaussian distributions with mean μ_k, the UCB-1 algorithm satisfies*

$$R(T) \leq \sum_{k=1}^{K} \Delta_k \cdot \left(\frac{8\sigma_k^2 \ln T}{\Delta_k^2} + \left(1 + \frac{\pi^2}{3}\right) \right)$$

We refer the reader to [10] for a complete proof. Note that the regret of the UCB algorithm is optimal (up to a constant factor) in T since it matches the lower bound of $\Omega(\ln T)$. We will now move to our next heuristic.

4.2 Thompson Sampling

The most prominently studied class of algorithms are the Upper Confidence Bound (UCB) algorithms discussed previously, that employ the "optimism in the face of uncertainty" heuristic [10]. Over the past few years, however, there has been a resurgence in interest in the Thompson Sampling (TS) algorithm [80], that approaches the problem from a Bayesian perspective.

Rigorous empirical evidence in favor of TS demonstrated by [29] has sparked new interest in the theoretical analysis of the algorithm, and the seminal work of [4, 5, 74] demonstrated the optimality of TS when rewards are bounded in $[0, 1]$ or are Gaussian. These results were extended in the work of [47] to more general, exponential family reward distributions. The empirical studies, along with theoretical guarantees, have established TS as a powerful algorithm for the stochastic bandit problem.

Thompson Sampling [80] proceeds by maintaining a posterior distribution over the parameters of the bandit arms. If we assume that for each arm k, the reward distribution $\mathcal{D}_k$ is parameterized by a (possibly vector) parameter θ_k that come from

a set $\boldsymbol{\Theta}$ with a prior probability distribution $p(\theta_k)$ over the parameters, the Thompson Sampling algorithm proceeds by selecting arms based on the posterior probability of the reward under the arms. For each round $t \in [T]$, the agent:

1. Draws parameters $\hat{\theta}_k(t)$ for each arm $k \in [K]$ from the posterior distribution of parameters, given the previous rewards $\mathbf{r}_k(t-1) = \{r_k^{(1)}, r_k^{(2)}, ...\}$ till round $t-1$ (note that the posterior distribution for each arm only depends on the rewards obtained using that arm). When $t = 1$, this is just the prior distribution over the parameters.

$$\hat{\theta}_k(t) \sim p(\theta_k|\mathbf{r}_k(t-1)) \propto p(\mathbf{r}_k(t-1)|\theta_k)p(\theta_k) \tag{4.3}$$

2. Given the drawn parameters $\hat{\theta}_k(t)$ for each arm, chooses arm a_t with the largest mean reward over the posterior distribution.

$$a_t = \arg\max_{k \in [K]} \mu_k(\hat{\theta}_k(t)) \tag{4.4}$$

3. Obtains reward r_t after taking action a_t and updates the posterior distribution for arm a_t.

4.2.1 Bayes Regret

It is evident that the performance of Thompson Sampling depends heavily on the priors chosen for the algorithm. If the priors for each arm are very close to the rewards, then it is likely that the agent will obtain a much lower regret since the posterior will concentrate quickly. While there has been recent work analysing the performance of TS under worst-case priors that have shown optimal performance in sub-Gaussian [4] and exponential-family [47] bandits under regret, we utilize a different metric to measure the performance of this Bayesian algorithm.

Bayes Regret (BR) [74] is the expected value of regret over the priors. Denoting the parameters over all arms as $\bar{\theta} = \{\theta_1, ..., \theta_k\}$ and their corresponding product

distribution as $\bar{\mathcal{D}} = \prod_i \mathcal{D}_i$, for any policy π, the Bayes Regret is given by:

$$\text{BayesRegret}(T, \pi) = \mathbb{E}_{\bar{\theta} \sim \bar{\mathcal{D}}}[R(T)]. \tag{4.5}$$

While the regret provides a stronger analysis, any bound on the Bayes Regret is essentially a bound on the expected regret, since if an algorithm admits a Bayes Regret of $O(g(T))$, then its Expected Regret is also stochastically bounded by $g(\cdot)$ [74]. Formally, we have, for constants M, ϵ:

$$\mathbb{P}\left(\frac{\mathbb{E}[R(T)|\bar{\theta}]}{g(T)} \geq M\right) \leq \epsilon \; \forall \; T \in \mathbb{N}. \tag{4.6}$$

Analogous bounds for Thompson Sampling can be obtained for sub-Gaussian rewards as well.

Theorem 3 (Bayes Regret of sub-Gaussian TS [18,74]). *For a K-armed unstructured stochastic bandit with rewards for arm $k \in [K]$ drawn i.i.d. from σ_k-sub-Gaussian distributions with mean μ_k, the UCB-1 algorithm satisfies*

$$BayesRegret(T) = O(\sqrt{KT \ln T})$$

and,

$$BayesRegret(T) = \omega(\sqrt{KT})$$

These results come independently from [18] and [74] respectively, and effectively bound the regret, implying that an optimal algorithm must be at least $\tilde{O}(KT)$ in the sub-Gaussian case. Armed with this background information, we now discuss algorithms for heavy-tailed bandits in the next chapter.

Chapter 5

α-Stable Stochastic Bandits

We develop both Bayesian and frequentist algorithms for the stochastic bandits problem under α-stable reward distributions. These algorithms are based on the concentration properties derived earlier. We begin with the frequentist α-UCB algorithm, which is largely based on the work of Bubeck *et. al.* [17] on heavy-tailed bandits.

5.1 The α-UCB Algorithm

The central idea of the UCB Algorithm is to obtain a confidence bound around the empirical mean for each arm k. As discussed earlier, heavy-tailed distributions, and especially α-stable distributions, which do not even admit a finite variance, provide a polynomial concentration bound around the mean. While this leads to an efficient algorithm (in terms of time complexity), it does not provide us with logarithmic regret [17], specifically in case of the UCB algorithm.

To alleviate this issue, we must incorporate a robust estimator of the mean, which allows for a concentration that is much stronger than the basic polynomial concentration. The work of Bubeck *et. al.* [17] first explored the use of robust mean estimators in context of heavy-tailed stochastic bandits, and we will follow a similar approach for α-stable bandits.

5.1.1 Robust Mean Estimators for α-Stable Densities

There have been several mean estimators proposed in the domain of robust statistics. The most basic one is the truncated mean estimator, which calculates a "truncated" version of the empirical mean by rejecting all samples larger than a particular margin. Similar to it are other trimming-based estimators, such as the trimmed mean of Tukey [11]. More elaborate estimators also exist that provide better convergence properties, such as the median-of-means estimator [8] and Catoni's M-estimator [27]. However, there are two restrictions on our problem that prevent us from using very sophisticated estimators:

- **Limit on computational complexity**. The median-of-means estimator [8] has an update rule that requires $O(\ln T)$ space and $O(\ln \ln T)$ time per update, which becomes infeasible as the problem horizon grows.
- **No higher order moments available.** Catoni's M-estimator requires the underlying distribution to admit a finite variance, which, in the case of α-stable rewards, is not possible.

Since the truncated mean estimator does not require any guarantee on moments (only requires finite mean) and has constant time and space complexity per update, we proceed with a truncated mean estimator as the basis for our algorithm. For the concentration of the truncated mean estimator, we first require a bound on the raw moments of the probability distribution, which we derive as follows.

Lemma 11 (Proposition 2.2 of [62]). *For any random variable* $X \sim S_\alpha(0, \sigma, 0), \epsilon \in (-\infty, \alpha - 1)$ *and* $\nu \in \mathbb{R}$,

$$\mathbb{E}[|X - \nu|^{1+\epsilon}] = \frac{\epsilon \cdot \sigma^{1+\epsilon}}{\sin\left(\frac{\pi \cdot \epsilon}{2}\right)\Gamma(1-\epsilon)} \left[\frac{\nu}{\sigma} \int_0^\infty u^{-(1+\epsilon)} e^{-u^\alpha} \sin\left(\frac{\nu u}{\sigma}\right) du + \alpha \int_0^\infty u^{\alpha-\epsilon-2} e^{-u^\alpha} \cos\left(\frac{\nu u}{\sigma}\right) du\right].$$

This result provides an exact expression for the $(1 + \epsilon)$-moment of an α-stable distribution around an arbitrary constant ν. However, as can be seen, this can diverge

as $\nu \to \infty$. To prevent this behavior, we make an additional assumption on our bandit problem.

Assumption 1. *For the K-armed unstructured bandit problem in consideration, assume that rewards are distributed following a symmetric α-stable distribution with fixed scale parameter σ, known beforehand, and that $\mu_k \leq M$, $\forall k \in [K]$ for a known constant M.*

This assumption on the problem is not entirely unfeasible, since for most bandit problems, we have a reasonable upper bound on the *possible* mean of each arm. Note that this does not imply that the support of the reward distribution is bounded. We now proceed with bounding the raw moments of the distribution under this assumption.

Lemma 12 (Raw Moment Bound for Symmetric α-Stable Densities). *For any random variable $X \sim S_\alpha(0, \sigma, \mu), \epsilon \in (-\infty, \alpha - 1), \mu \leq M$,*

$$\mathbb{E}[|X|^{1+\epsilon}] \leq \frac{\epsilon \cdot \sigma^{1+\epsilon} \left(M \cdot \Gamma(-\epsilon/\alpha) + \sigma\alpha\Gamma(1 - \frac{\epsilon+1}{\alpha})\right)}{\sigma\alpha \sin\left(\frac{\pi\cdot\epsilon}{2}\right) \Gamma(1-\epsilon)}$$

Proof. Let $X \sim S_\alpha(0, \sigma, \mu)$. Then $X - \mu \sim S_\alpha(0, \sigma, 0)$. Applying Lemma 11 to $X - \mu$ with $\nu = -\mu, \epsilon \in (-\infty, \alpha - 1)$, we have,

$$\begin{aligned}
&\mathbb{E}[|X|^{1+\epsilon}] \\
&= \mathbb{E}[|X - \mu - (-\mu)|^{1+\epsilon}] \\
&= \frac{\epsilon \cdot \sigma^{1+\epsilon}}{\sin\left(\frac{\pi\cdot\epsilon}{2}\right) \Gamma(1-\epsilon)} \left[\frac{-\mu}{\sigma} \int_0^\infty u^{-(1+\epsilon)} e^{-u^\alpha} \sin\left(\frac{-\mu u}{\sigma}\right) du + \alpha \int_0^\infty u^{\alpha-\epsilon-2} e^{-u^\alpha} \cos\left(\frac{\mu u}{\sigma}\right) du\right] \\
&\overset{(a)}{=} \frac{\epsilon \cdot \sigma^{1+\epsilon}}{\sin\left(\frac{\pi\cdot\epsilon}{2}\right) \Gamma(1-\epsilon)} \left[\frac{\mu}{\sigma} \int_0^\infty u^{-(1+\epsilon)} e^{-u^\alpha} \sin\left(\frac{\mu u}{\sigma}\right) du + \alpha \int_0^\infty u^{\alpha-\epsilon-2} e^{-u^\alpha} \cos\left(\frac{\mu u}{\sigma}\right) du\right] \\
&\overset{(b)}{\leq} \frac{\epsilon \cdot \sigma^{1+\epsilon}}{\sin\left(\frac{\pi\cdot\epsilon}{2}\right) \Gamma(1-\epsilon)} \left[\frac{\mu}{\sigma} \int_0^\infty u^{-(1+\epsilon)} e^{-u^\alpha} du + \alpha \int_0^\infty u^{\alpha-\epsilon-2} e^{-u^\alpha} du\right] \\
&\overset{(c)}{=} \frac{\epsilon \cdot \sigma^{1+\epsilon}}{\sin\left(\frac{\pi\cdot\epsilon}{2}\right) \Gamma(1-\epsilon)} \left[\frac{\mu}{\sigma\alpha} \int_0^\infty t^{-1-\epsilon/\alpha} e^{-t} dt + \int_0^\infty t^{-\frac{1+\epsilon}{\alpha}} e^{-t} dt\right] \\
&\overset{(d)}{\leq} \frac{\epsilon \cdot \sigma^{1+\epsilon} \left(M \cdot \Gamma(-\epsilon/\alpha) + \sigma\alpha\Gamma(1 - \frac{\epsilon+1}{\alpha})\right)}{\sigma\alpha \sin\left(\frac{\pi\cdot\epsilon}{2}\right) \Gamma(1-\epsilon)}
\end{aligned}$$

Here, (a) follows from $\sin(-x) = -\sin(x)$, (b) follows from $\sin(x) \leq 1, \cos(x) \leq 1\ \forall x$, (c) follows by the substitution $t = u^{\alpha}$, and (d) follows from $\mu \leq M$. □

Now that we have a bound on the raw moments of the distribution, we can derive a concentration result. We adopt the concentration from [17], restated next for clarity, however, a result of identical order can be derived by applying Markov's Inequality to the $(1+\epsilon)$ raw moment.

Lemma 13 (Lemma 1 of [17]). *Let $\delta \in (0,1), \epsilon \in (0,1], u > 0$. Consider the truncated empirical mean $\hat{\mu}_T$ defined as,*

$$\hat{\mu}_T = \frac{1}{n}\sum_{t=1}^{n} X_t \mathbb{1}\left\{|X_t| \leq \left(\frac{ut}{\log(\delta^{-1})}\right)^{\frac{1}{1+\epsilon}}\right\}$$

If $\mathbb{E}[|X|^{1+\epsilon}] < u, \mathbb{E}[X] = \mu$, then with probability $1-\delta$,

$$\hat{\mu}_t \leq \mu + 4u^{\frac{1}{1+\epsilon}}\left(\frac{\log(\delta^{-1})}{n}\right)^{\frac{\epsilon}{1+\epsilon}}$$

The full proof for this Lemma can be found in [17]. By replacing the previous result in this one, we obtain our final concentration bound.

Lemma 14. *For n copies of any random variable $X \sim S_{\alpha}(0,\sigma,\mu), \epsilon \in (-\infty, \alpha - 1), \mu \leq M$, and $\epsilon \in (0, \alpha-1)$, we have, with probability at least $1-\delta$,*

$$|\hat{\mu}_T - \mu| \leq 4\sigma\left(\frac{\epsilon\left(M\Gamma(-\epsilon/\alpha) + \sigma\alpha\Gamma(1-\frac{\epsilon+1}{\alpha})\right)}{\sigma\alpha\sin\left(\frac{\pi\cdot\epsilon}{2}\right)\Gamma(1-\epsilon)}\right)^{\frac{1}{1+\epsilon}}\left(\frac{\log(\delta^{-1})}{n}\right)^{\frac{\epsilon}{1+\epsilon}}$$

Proof. $X \sim S_{\alpha}(0,\sigma,\mu)$. Applying Lemma 12 to X, we obtain $u = \frac{\epsilon\cdot\sigma^{1+\epsilon}\left(M\cdot\Gamma(-\epsilon/\alpha)+\sigma\alpha\Gamma(1-\frac{\epsilon+1}{\alpha})\right)}{\sigma\alpha\sin\left(\frac{\pi\cdot\epsilon}{2}\right)\Gamma(1-\epsilon)}$. Using this value of u for both tails of $\hat{\mu}$ in Lemma 13 with probability $\delta/2$, we obtain the result. □

This result provides us with the following robust estimator for any arm k of the K-armed bandit. We substitute $H(\epsilon,\alpha,\sigma) = \left(\frac{\epsilon\left(M\cdot\Gamma(-\epsilon/\alpha)+\sigma\alpha\Gamma(1-\frac{\epsilon+1}{\alpha})\right)}{\sigma\alpha\sin\left(\frac{\pi\cdot\epsilon}{2}\right)\Gamma(1-\epsilon)}\right)$ for brevity.

Definition 4. *Let $\delta \in (0,1), \epsilon \in (0, \alpha - 1), k \in [K]$ and time horizon be T. For any time $t \in [T]$, the truncated mean estimator $\hat{r}_k^*(t)$ is given by:*

$$\hat{\mu}_k^T(t) = \frac{1}{n_k(t)} \sum_{i=1}^{n_k(t)} r_k^{(i)} \mathbb{1} \left\{ |r_k^{(i)}| \leq \left(\frac{H(\epsilon, \alpha, \sigma) \cdot i}{2 \log(t)} \right)^{\frac{1}{1+\epsilon}} \right\} \tag{5.1}$$

We now proceed with the algorithm.

5.1.2 UCB on α-Stable Rewards

The UCB algorithm we develop, titled α-UCB, is similar in spirit to the original UCB algorithm for sub-Gaussian rewards. We consider the K-armed unstructured stochastic bandit problem, where for each arm $k \in [K]$, rewards are drawn from $S_\alpha(0, \mu_k, \sigma)$, and σ, α are known *a priori*. Additionally, we make the standard assumption that $\forall k \in [K], \mu_k \leq M$ for a known $M > 0$. The algorithm proceeds as follows.

For each round t, the agent calculates the following for each arm k:

$$Q_k(t) = \mu_k^T(t-1) + 4\sigma \left(H(\epsilon, \alpha, \sigma)\right)^{\frac{1}{1+\epsilon}} \left(\frac{\log(t)}{n_k(t-1)} \right)^{\frac{\epsilon}{1+\epsilon}} \tag{5.2}$$

If $n_k(t-1) = 0$, then we set $Q_k(t) = \infty$. The agent then chooses the arm with maximum $Q_k(t)$.

$$a_t = \arg\max_{k \in [K]} Q_k(t) \tag{5.3}$$

It is important to note that instead of the empirical mean, we employ a truncated mean estimator in this setting. The algorithm is straightforward, with constant time and space per iteration. We now bound the regret obtained by this algorithm.

5.1.3 Regret Analysis

The regret analysis follows the typical procedure of the UCB algorithm, by bounding the number of pulls of a suboptimal arm.

Theorem 4. *Consider the K-armed unstructured stochastic bandit problem, where for each arm $k \in [K]$, rewards are drawn from $S_\alpha(0, \mu_k, \sigma)$, and σ, α are known a*

priori and $\forall k \in [K], \mu_k \leq M$ *for a known* $M > 0$. *Let* $\Delta_k = \mu^* - \mu_k$. *The* α*-UCB strategy then satisfies, for any* $T \geq 0$,

$$R(T) \leq \inf_{\epsilon \in (0,\alpha-1)} \sum_{k=1}^{K} \left(H(\epsilon, \alpha, \sigma)^{1/\epsilon} \left(\frac{(8\sigma)^{1+\frac{1}{\epsilon}}}{\Delta_k^{\frac{1}{\epsilon}}} \right) + 2\Delta_k \right) \ln(T)$$

Proof. We start by bounding the number of pulls of a suboptimal arm. Let $k^* = \arg\max_{k \in [K]} \mu_k$ denote the optimal arm, and μ^* denote the corresponding optimal mean reward of that arm. Then, for any arm k,

$$\begin{aligned}
\mathbb{E}[n_k(T)] &= \mathbb{E}\left[\sum_{t=1}^{T} \mathbb{1}\{a_t = k\} \right] \\
&= \sum_{t=1}^{T} \Pr\{a_t = k\} \\
&\leq \eta + \sum_{t=1}^{T} \Pr\{a_t = k \cap n_k(t) \geq \eta\}.
\end{aligned}$$

η is a constant that will be decided later. Now, we know that arm k can be pulled only in one of three events:

$$\begin{aligned}
E_1 &= \left\{ \mu_k^T(t-1) \leq \mu^* - 4\sigma \left(H(\epsilon, \alpha, \sigma)\right)^{\frac{1}{1+\epsilon}} \left(\frac{\log(t)}{n_k(t-1)} \right)^{\frac{\epsilon}{1+\epsilon}} \right\} \\
E_2 &= \left\{ \mu_k^T(t-1) \geq \mu^* + 4\sigma \left(H(\epsilon, \alpha, \sigma)\right)^{\frac{1}{1+\epsilon}} \left(\frac{\log(t)}{n_k(t-1)} \right)^{\frac{\epsilon}{1+\epsilon}} \right\} \\
E_3 &= \left\{ \mu^* < \mu_k + 8\sigma \left(H(\epsilon, \alpha, \sigma)\right)^{\frac{1}{1+\epsilon}} \left(\frac{\log(t)}{n_k(t-1)} \right)^{\frac{\epsilon}{1+\epsilon}} \right\}
\end{aligned}$$

From Lemma 14, we know that events E_1 and E_2 hold with a probability at most $\frac{1}{2t}$, by setting $\delta = 1/t$. We will now bound the probability of event E_3. When E_3 occurs,

$$\begin{aligned}
&\frac{(\mu^* - \mu_k)^{1+\epsilon}}{(8\sigma)^{1+\epsilon} H(\epsilon, \alpha, \sigma)} < \left(\frac{\log(t)}{n_k(t-1)} \right)^{\epsilon} \\
&\implies n_k(t-1) < H(\epsilon, \alpha, \sigma)^{1/\epsilon} \left(\frac{8\sigma}{\Delta_k} \right)^{1+\frac{1}{\epsilon}} \log(t)
\end{aligned}$$

This implies that if $n_k(t-1) \geq H(\epsilon,\alpha,\sigma)^{1/\epsilon}\left(\frac{8\sigma}{\Delta_k}\right)^{1+\frac{1}{\epsilon}}\log(T)$, then, by the monotonicity of the logarithm function, E_3 does not occur. Hence, we can set $\eta = \lceil H(\epsilon,\alpha,\sigma)^{1/\epsilon}\left(\frac{8\sigma}{\Delta_k}\right)^{1+\frac{1}{\epsilon}}\log(T)\rceil$. Combining these results, we have,

$$\begin{aligned}
\mathbb{E}[n_k(T)] &\leq \eta + \sum_{t=1}^{T}\Pr\{a_t = k \cap n_k(t) \geq \eta\} \\
&\leq \eta + \sum_{t=1}^{T}\sum_{j=1}^{3}\Pr\{a_t = k \cap n_k(t) \geq \eta \cap E_j\} \\
&\leq \lceil H(\epsilon,\alpha,\sigma)^{1/\epsilon}\left(\frac{8\sigma}{\Delta_k}\right)^{1+\frac{1}{\epsilon}}\log(T)\rceil + \sum_{t=1}^{T}\frac{1}{t} \\
&\leq \ln(T)\left(H(\epsilon,\alpha,\sigma)^{1/\epsilon}\left(\frac{8\sigma}{\Delta_k}\right)^{1+\frac{1}{\epsilon}} + 2\right)
\end{aligned}$$

Now, we can bound the regret using the decomposition of Lemma 1.

$$R(T) \leq \sum_{k=1}^{K}\left(H(\epsilon,\alpha,\sigma)^{1/\epsilon}\left(\frac{(8\sigma)^{1+\frac{1}{\epsilon}}}{\Delta_k^{\frac{1}{\epsilon}}}\right) + 2\Delta_k\right)\ln(T)$$

If we are given the horizon in advance, we can optimize over $\epsilon \in (0, \alpha-1)$ to obtain a tighter bound.

$$R(T) \leq \inf_{\epsilon\in(0,\alpha-1)}\sum_{k=1}^{K}\left(H(\epsilon,\alpha,\sigma)^{1/\epsilon}\left(\frac{(8\sigma)^{1+\frac{1}{\epsilon}}}{\Delta_k^{\frac{1}{\epsilon}}}\right) + 2\Delta_k\right)\ln(T)$$

□

We can see that this bound is optimal in T, however there is a stronger dependence on Δ_k compared to the sub-Gaussian case. This dependence is unimprovable, as pointed out in a result from [17].

Theorem 5 (Theorem 2 of [17], Lower Bound on Heavy-Tailed Regret). *For any $\Delta \in (0, 0.25)$ there exist two distributions ν_1 and ν_2 $(1+\epsilon)$ heavy-tailed distributions such that the following holds. Consider an algorithm such that for any two-armed bandit problem with arm 2 being suboptimal, one as $\mathbb{E}[n_2(T)] = o(n^a)$ for any $a > 0$.*

Then on the two-armed bandit problem with distributions ν_1 and ν_2, the algorithm satisfies

$$\lim_{n\to\infty} \inf \frac{R(T)}{\ln T} \geq \frac{0.4}{\Delta^{1/\epsilon}}$$

This implies that this strategy is optimal in both T and its dependence on the problem instance. However, if supplied with prior knowledge, it is possible to perform better. In the next section, we discuss a Bayesian algorithm that incorporates prior knowledge into the bandit algorithm.

5.2 The α-Thompson Sampling Algorithm

In this section, we discuss a version of Thompson Sampling, a Bayesian algorithm, for α-stable stochastic bandits. We consider the setting where, for an arm k, the corresponding reward distribution is given by $\mathcal{D}_k = S_\alpha(0, \sigma, \mu_k)$ where $\alpha \in (1, 2), \sigma \in \mathbb{R}^+$ are known in advance, and μ_k is unknown[1]. Note that $\mathbb{E}[r_k] = \mu_k$, and hence we set a prior distribution over the variable μ_k which is the expected reward for arm k. We can see that since the only unknown parameter for the reward distributions is μ_k, $\mathcal{D}_k$ is parameterized by $\theta_k = \mu_k$. Therefore, the steps for Thompson Sampling can be outlined as follows. For each round $t \in [T]$, the agent:

1. Draws parameters $\hat{\theta}_k(t) = \bar{\mu}_k(t)$ for each arm $k \in [K]$ from the posterior distribution of parameters, given the previous rewards $\mathbf{r}_k(t-1) = \{r_k^{(1)}, r_k^{(2)}, ...\}$ till round $t-1$.

$$\bar{\mu}_k(t) \sim p(\mu_k | \mathbf{r}_k(t-1)) \propto p(\mathbf{r}_k(t-1) | \mu_k) p(\mu_k) \tag{5.4}$$

2. Given the drawn parameters $\bar{\mu}_k(t)$ for each arm, chooses arm a_t with the largest mean reward over the posterior distribution.

$$a_t = \arg\max_{k \in [K]} \bar{\mu}_k(t) \tag{5.5}$$

[1]Typically, more general settings have been explored for TS in the Gaussian case, where the variance is also unknown. Our algorithm can also be extended to an unknown scale (σ) using an inverse Gamma prior, similar to [38].

3. Obtains reward r_t after taking action a_t and updates the posterior distribution for arm a_t.

Now, we will derive the form of the prior distribution, and outline an algorithm for Bayesian inference.

5.2.1 Scale Mixtures of Normals

On setting $\gamma = 2$ (Gaussian) and $\beta = \alpha/2 < 1$ in Lemma 6, the product distribution $X = ZY^{1/2}$ is stable with exponent α. This property is an instance of the general framework of *scale mixtures of normals* (SMiN) [9], which are described by the following:

$$p_X(x) = \int_0^\infty \mathcal{N}(x|0, \lambda\sigma^2) p_\Lambda(\lambda) d\lambda \tag{5.6}$$

This framework contains a large class of heavy-tailed distributions which include the exponential power law, Student's-t and symmetric α-stable distributions [38]. The precise form of the variable X depends on the *mixing distribution* $p_\Lambda(\lambda)$. For instance, when p_Λ is the inverted Gamma distribution (the conjugate prior for a unknown variance Gaussian), the resulting p_X follows a Student's-t distribution.

Bayesian inference directly from $S_\alpha(0, \sigma, \mu)$ is difficult: the non-analytic density prevents a direct evaluation of the likelihood function, and the non-Gaussianity introduces difficulty in its implementation. However, the SMiN representation enables us to draw samples directly from $S_\alpha(0, \sigma, \mu)$ using the auxiliary variable λ:

$$x \sim \mathcal{N}(\mu, \lambda\sigma^2), \lambda \sim S_{\alpha/2}(1, 1, 0) \tag{5.7}$$

This sampling assists in inference since x is Gaussian conditioned on λ: given samples of λ, we can generate x from the induced conditional distribution (which is Gaussian).

5.2.2 Posteriors for α-Stable Rewards

Let us examine approximate inference for a particular arm $k \in [K]$. At any time $t \in [T]$, assume this arm has been pulled $n_k(t)$ times previously, and hence we have a

vector of reward samples $\mathbf{r}_k(t) = \{r_k^{(1)}, ..., r_k^{(n_k(t))}\}$ observed until time t. Additionally, assume we have $n_k(t)$ samples of an auxiliary variable $\boldsymbol{\lambda}_k(t) = \{\lambda_k^{(1)}, ..., \lambda_i^{(n_k(t))}\}$ where $\lambda_k \sim S_{\alpha/2}(1, 1, 0)$.

Recall that $r_k \sim S_\alpha(0, \sigma, \mu_k)$ for an unknown (but fixed) μ_k. From the SMiN representation, we know that r_k is conditionally Gaussian given the auxiliary variable λ_k, that is $p(r_k|\lambda_k, \mu_k) \sim \mathcal{N}(\mu_k, \lambda_k\sigma^2)$. We can then obtain the conditional likelihood as the following:

$$p(\mathbf{r}_k(t)|\boldsymbol{\lambda}_k(t), \mu_k) \propto \exp\left(-\frac{1}{2\sigma^2}\left(\sum_{i=1}^{n_k(t)} \frac{(r_k^{(i)} - \mu_k)^2}{\lambda_k^{(i)}}\right)\right). \tag{5.8}$$

We can now assume a conjugate prior over μ_k, which is a normal distribution with mean μ_k^0 and variance σ^2. We then obtain the posterior density for μ_k.

Lemma 15. *The posterior density for the mean reward μ_k for arm $k \in [K]$ at $t \in [T]$ trials follows, for obtained rewards $\mathbf{r}_k(t)$ and sampled auxiliary variables $\boldsymbol{\lambda}_k(t)$,*

$$p(\mu_k|\mathbf{r}_k(t), \boldsymbol{\lambda}_k(t)) \propto \mathcal{N}\left(\hat{\mu}_k(t), \hat{\sigma}_k^2(t)\right) \text{ where, } \hat{\sigma}_k^2(t) = \frac{\sigma^2}{\sum_{i=1}^{n_k(t)} \frac{1}{\lambda_k^{(i)}} + 1}, \ \hat{\mu}_k(t) = \frac{\sum_{i=1}^{n_k(t)} \frac{r_k^{(i)}}{\lambda_k^{(i)}} + \mu_k^0}{\sum_{i=1}^{n_k(t)} \frac{1}{\lambda_k^{(i)}} + 1}.$$

Proof. At any time $t \in [T]$, arm $k \in [K]$, assume this arm has been pulled $n_k(t)$ times previously, and hence we have a vector of reward samples $\mathbf{r}_k(t) = \{r_k^{(1)}, ..., r_k^{(n_k(t))}\}$ observed until time t. Additionally, assume we have $n_k(t)$ samples of an auxiliary variable $\boldsymbol{\lambda}_k(t) = \{\lambda_k^{(1)}, ..., \lambda_i^{(n_k(t))}\}$ where $\lambda_k \sim S_{\alpha/2}(1, 1, 0)$. Recall that $r_k \sim S_\alpha(0, \sigma, \mu_k)$

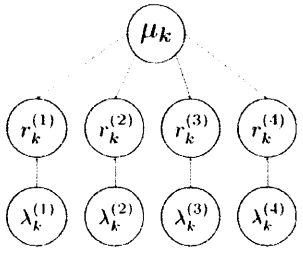

Figure 5-1: Directed graphical model for arm k where $n_k(t) = 4$.

for an unknown (but fixed) μ_k. From the SMiN representation, we know that r_k is conditionally Gaussian given the auxiliary variable λ_k, that is $p(r_k|\lambda_k, \mu_k) \sim \mathcal{N}(\mu_k, \lambda_k\sigma^2)$.

We can then obtain the conditional likelihood as

$$p(\mathbf{r}_k(t)|\boldsymbol{\lambda}_k(t),\mu_k) \propto \exp\left(\frac{1}{2\sigma^2}\left(\sum_{i=1}^{n_k(t)}\frac{(r_k^{(i)}-\mu_k)^2}{\lambda_k^{(i)}}\right)\right).$$

We can now assume a conjugate prior over μ_k, which is a normal distribution with mean μ_k^0 and variance σ^2. By Bayes' rule, we know that

$$\begin{aligned}
p(\mu_k|\boldsymbol{\lambda}_k(t),\mathbf{r}_k(t)) &\propto p\left(\mathbf{r}_k(t)|\mu_k,\boldsymbol{\lambda}_k(t)\right)p(\mu_k|\boldsymbol{\lambda}_k(t)) \\
&\overset{(a)}{\propto} p(\mathbf{r}_k(t)|\mu_k,\boldsymbol{\lambda}_k(t))p(\mu_k) \\
&\propto \exp\left(-\frac{1}{2\sigma^2}\left(\sum_{i=1}^{n_k(t)}\frac{(r_k^{(i)}-\mu_k)^2}{\lambda_k^{(i)}}\right)\right)\cdot\exp\left(-\frac{(\mu_k-\mu_k^0)^2}{2\sigma^2}\right) \\
&\propto \exp\left(-\frac{1}{2\sigma^2}\left(\sum_{i=1}^{n_k(t)}\frac{(r_k^{(i)}-\mu_k)^2}{\lambda_k^{(i)}}+(\mu_k-\mu_k^0)^2\right)\right) \\
&\propto \exp\left(-\frac{1}{2\sigma^2}\left(\mu_k^2\left(\sum_{i=1}^{n_k(t)}\frac{1}{\lambda_k^{(i)}}+1\right)-2\mu_k\left(\sum_{i=1}^{n_k(t)}\frac{r_k^{(i)}}{\lambda_k^{(i)}}+\mu_k^0\right)\right)\right) \\
&\propto \exp\left(-\frac{1}{2\sigma^2\left(\sum_{i=1}^{n_k(t)}\frac{1}{\lambda_k^{(i)}}+1\right)^{-1}}\left(\mu_k^2-2\mu_k\left(\frac{\left(\sum_{i=1}^{n_k(t)}\frac{r_k^{(i)}}{\lambda_k^{(i)}}+\mu_k^0\right)}{\left(\sum_{i=1}^{n_k(t)}\frac{1}{\lambda_k^{(i)}}+1\right)}\right)\right)\right) \\
&\propto \exp\left(-\frac{\left(\mu_k-\frac{\left(\sum_{i=1}^{n_k(t)}\frac{r_k^{(i)}}{\lambda_k^{(i)}}+\mu_k^0\right)}{\left(\sum_{i=1}^{n_k(t)}\frac{1}{\lambda_k^{(i)}}+1\right)}\right)^2}{2\sigma^2\left(\sum_{i=1}^{n_k(t)}\frac{1}{\lambda_k^{(i)}}+1\right)^{-1}}\right).
\end{aligned}$$

Where (a) follows from the independence of $\lambda_k^{(t)}$ and μ_k. Comparing the above to a standard normal distribution, we then obtain the posterior density for μ_k as

$$p(\mu_k|\mathbf{r}_k(t),\boldsymbol{\lambda}_k(t)) \propto \mathcal{N}\left(\hat{\mu}_k(t),\hat{\sigma}_k^2(t)\right) \text{ where, } \hat{\sigma}_k^2(t)=\frac{\sigma^2}{\sum_{i=1}^{n_k(t)}\frac{1}{\lambda_k^{(i)}}+1},\ \hat{\mu}_k(t)=\frac{\sum_{i=1}^{n_k(t)}\frac{r_k^{(i)}}{\lambda_k^{(i)}}+\mu_k^0}{\sum_{i=1}^{n_k(t)}\frac{1}{\lambda_k^{(i)}}+1}. \tag{5.9}$$

□

We know that $\hat{\sigma}_k^2(t) > 0$ since $\lambda_k^{(i)} > 0 \, \forall i$ as they are samples from a positive stable distribution ($\beta = 1$). Given $\mathbf{r}_k(t)$ and μ_k, we also know that the individual elements of $\boldsymbol{\lambda}_k(t)$ are independent, which provides us with the following decomposition for the conditional density of $\boldsymbol{\lambda}_k(t)$,

$$p(\boldsymbol{\lambda}_k(t)|\mathbf{r}_k(t), \mu_k) = \prod_{i=1}^{n_k(t)} p(\lambda_k^{(i)}|r_k^{(i)}, \mu_k), \text{ where, } p(\lambda_k^{(i)}|r_k^{(i)}, \mu_k) \propto \mathcal{N}(r_k^{(i)}|\mu_k, \lambda_k^{(i)}, \sigma^2) f_{\alpha/2,1}(\lambda_k^{(i)}). \tag{5.10}$$

Here, $f_{\alpha,\beta}(\cdot)$ is the density of a random variable following $S_\alpha(\beta, 1, 0)$. Our stepwise posterior sampling routine is hence as follows. At any time t, after arm k is pulled and we receive reward $r_k^{n_k(t)}$, we set $\mathbf{r}_k(t) = [\mathbf{r}_k(t-1), r_k^{n_k(t)}]$. Then for a fixed Q iterations, we repeat:

1. For $i \in [1, n_k(t)]$, draw $\lambda_k^{(i)} \sim p(\lambda_k^{(i)}|r_k^{(i)}, \mu_k(t))$.

2. Draw $\mu_k(t) \sim p(\mu_k|\mathbf{r}_k(t), \boldsymbol{\lambda}_k(t))$.

Sampling from the conditional posterior of μ_k is straightforward since it is Gaussian. To sample from the complicated posterior of $\lambda_k^{(i)}$, we utilize rejection sampling.

5.2.3 Rejection Sampling for $\lambda_k^{(i)}$

Sampling directly from the posterior is intractable since it is not analytical. Therefore, to sample $\lambda_k^{(i)}$ we follow the pipeline described in [38]. We note that the likelihood of the mean-normalized reward $v_k^{(i)} = r_k^{(i)} - \mu_k(t)$ forms a valid rejection function since it is bounded:

$$p\left(v_k^{(i)}|0, \lambda_k^{(i)}\sigma^2\right) \leq \frac{1}{v_k^{(i)}\sqrt{2\pi}} \exp(-1/2) \tag{5.11}$$

Since $v_k^{(i)} \sim \mathcal{N}(0; \lambda_k^{(i)}\sigma^2)$. Thus, we get the procedure:

1. Draw $\lambda_k^{(i)} \sim S_{\alpha/2}(1, 1, 0)$ (using Algorithm 1).

2. Draw $u \sim \mathcal{U}\left(0, (v_k^{(i)}\sqrt{2\pi})^{-1} \exp(-1/2)\right)$.

3. If $u > p(v_k^{(i)}|0, \lambda_k^{(i)}\sigma^2)$, reject $\lambda_k^{(i)}$ and go to Step 1.

Algorithm 2 α-Thompson Sampling

Input: Arms $k \in [K]$, priors $\mathcal{N}(\mu_k^0, \sigma^2)$ for each arm.
Set $D_k = 1, N_k = 0$ for each arm k.
for For each iteration $t \in [1, T]$ **do**
  Draw $\bar{\mu}_k(t) \sim \mathcal{N}\left(\frac{\mu_k^0 + N_k}{D_k}, \frac{\sigma^2}{D_k}\right)$ for each arm k.
  Choose arm $A_t = \arg\max_{k \in [K]} \bar{\mu}_k(t)$, and get reward r_t.
  for $q \in [0, Q)$ **do**
    Calculate $v_{A_t}^{(t)} = r_t - \bar{\mu}_{A_t}$.
    Draw $\lambda_k^{(t)}$ following Section 5.2.3.
    Set $D_q = D_k + 1/\lambda_k^{(t)}, N_q = N_k + r_t/\lambda_k^{(t)}$.
    Draw $\bar{\mu}_{A_t} \sim \mathcal{N}\left(\frac{\mu_k^0 + N_q}{D_q}, \frac{\sigma^2}{D_q}\right)$.
  end for
  Set $D_k = D_k + 1/\lambda_k^{(t)}, N_k = N_k + r_t/\lambda_k^{(t)}$.
end for

Combining all these steps, we can now outline our algorithm, α-Thompson Sampling (α-TS) as described in Algorithm 2.

It is critical to note that in Algorithm 2, we do not draw from the full vector of $\boldsymbol{\lambda}_k(t)$ at every iteration, but only from the last obtained reward. This is done to accelerate the inference process, and while it leads to a slower convergence of the posterior, we observe that it performs well in practice. Alternatively, one can re-sample $\boldsymbol{\lambda}_k(t)$ over a fixed window of the previous rewards, to prevent the runtime from increasing linearly while enjoying faster convergence.

5.2.4 Regret Analysis

In this section, we derive an upper bound on the finite-time Bayesian Regret (BR) incurred by the α-TS algorithm. We continue with the notation used in previous sections, and assume a K armed bandit with T maximum trials. Each arm k follows an α-stable reward $S_\alpha(0, \sigma, \mu_k)$, and without loss of generality, let $\mu^* = \max_{k \in [K]} \mu_i$ denote the arm with maximum mean reward.

Theorem 6 (Regret Bound). *Let $K > 1, \alpha \in (1, 2), \sigma \in \mathbb{R}^+, \mu_{k:k \in [K]} \in [-M, M]$. For a K-armed bandit with rewards for each arm k drawn from $S_\alpha(0, \sigma, \mu_k)$, we have,*

asymptotically, for ϵ chosen a priori such that $\epsilon \to (\alpha-1)^-$,

$$\textit{Bayes Regret}(T, \pi^{TS}) = O(K^{\frac{1}{1+\epsilon}} T^{\frac{2}{1+\epsilon}})$$

Proof. Consider a K-armed bandit with rewards for arm k drawn from $S_\alpha(0, \sigma, \mu_k)$. Let $n_k(t)$ denote the number of times arm k has been pulled until time t. Then $t-1 = \sum_{k=1}^{K} n_k(t)$. Let us denote the empirical average reward for arm k up to (and including) time $t-1$ as $\hat{r}_k(t)$, and denote the arm pulled at any time t as a_t, and the optimal arm as a_t^*. We then set an upper confidence bound for arm k at any time t as

$$U_k(t) = \text{clip}_{[-M,M]} \left[\hat{r}_k(t) + \sigma^{\frac{1}{\alpha}} \left(\frac{2C(1+\epsilon, \alpha)}{\delta n_k(t)^\epsilon} \right)^{\frac{1}{1+\epsilon}} \right] \tag{5.12}$$

for $0 < \epsilon < 1, M > 0$. Let E be the event when for all $k \in [K]$ arms, over all iterations $t \in [T]$, we have:

$$|\hat{r}_k(t) - \mu_k| \leq \sigma^{\frac{1}{\alpha}} \left(\frac{2C(1+\epsilon, \alpha)}{\delta n_k(t)^\epsilon} \right)^{\frac{1}{1+\epsilon}}. \tag{5.13}$$

Lemma 16. *For the setup described above, we have, for event E and $\delta \in (0,1)$,*

$$\mathbb{P}(E^c) \leq KT\delta.$$

Proof. From Lemma 10, for arm $k \in [K]$ at time $t \in [T]$:

$$\mathbb{P}\left(|\hat{r}_k(t) - \mu_k| \leq \sigma^{\frac{1}{\alpha}} \left(\frac{2C(1+\epsilon, \alpha)}{\delta n_k(t)^\epsilon} \right)^{\frac{1}{1+\epsilon}} \right) \geq 1 - \delta$$

The event E^c holds whenever the bound is violated for at least one arm k at one

instance t. Therefore,

$$\begin{aligned}
\mathbb{P}(E^c) &\leq \mathbb{P}\left(\bigcup_{\substack{k=1\\t=1}}^{K,T}\left\{|\hat{r}_k(t-1)-\mu_k| > \sigma^{\frac{1}{\alpha}}\left(\frac{2C(1+\epsilon,\alpha)}{\delta n_k(t)^\epsilon}\right)^{\frac{1}{1+\epsilon}}\right\}\right)\\
&\overset{(a)}{\leq} \sum_{\substack{k=1\\t=1}}^{K,T}\mathbb{P}\left(|\hat{r}_k(t-1)-\mu_k| > \sigma^{\frac{1}{\alpha}}\left(\frac{2C(1+\epsilon,\alpha)}{\delta n_k(t)^\epsilon}\right)^{\frac{1}{1+\epsilon}}\right)\\
&\overset{(b)}{\leq} KT\delta.
\end{aligned}$$

Where (a) is an application of the union bound, and (b) is obtained using Lemma 10. □

Using this lemma, we can now prove Theorem 6. We begin with the seminal result of [74].

Lemma 17 (Proposition 1 of [74]). *Let π^{TS} be any policy followed by Thompson Sampling. For any sequence of upper confidence bounds $\{U_t | t \in \mathbb{N}\}$,*

$$\textit{Bayes Regret}(T,\pi^{TS}) = \mathbb{E}\left[\sum_{t=1}^{T}\left(U_{a_t}(t)-\mu_{a_t}\right)\right] + \mathbb{E}\left[\sum_{t=1}^{T}\left(\mu_{a_t^*}-U_{a_t^*}(t)\right)\right]$$

We use this result directly from [74], and refer readers to the full text for the proof. By the tower rule, we can condition over event E:

$$\begin{aligned}
\text{Bayes Regret}(T,\pi^{TS}) = \mathbb{E}\left[\sum_{t=1}^{T}\left(U_{a_t}(t)-\mu_{a_t}\right)+\left(\mu_{a_t^*}-U_{a_t^*}(t)\right)\middle| E\right]\mathbb{P}(E) +\\
\mathbb{E}\left[\sum_{t=1}^{T}\left(U_{a_t}(t)-\mu_{a_t}\right)+\left(\mu_{a_t^*}-U_{a_t^*}(t)\right)\middle| E^c\right]\mathbb{P}(E^c)
\end{aligned}$$

Since $\mathbb{P}(E) \leq 1$,

$$\text{Bayes Regret}(T, \pi^{TS}) \leq \mathbb{E}\left[\sum_{t=1}^{T} (U_{a_t}(t) - \mu_{a_t}) + \left(\mu_{a_t^*} - U_{a_t^*}(t)\right) \middle| E\right] + \mathbb{E}\left[\sum_{t=1}^{T} (U_{a_t}(t) - \mu_{a_t}) + \left(\mu_{a_t^*} - U_{a_t^*}(t)\right) \middle| E^c\right] \mathbb{P}(E^c)$$

When E^c holds, each term in the summation in the conditional expectation is bounded by $4M$ (Equation 5.15). Therefore,

$$\begin{aligned}
\text{Bayes Regret}(T, \pi^{TS}) &\leq \mathbb{E}\left[\sum_{t=1}^{T} (U_{a_t}(t) - \mu_{a_t}) + \left(\mu_{a_t^*} - U_{a_t^*}(t)\right) \middle| E\right] + 4MT \cdot \mathbb{P}(E^c) \\
&\overset{(a)}{\leq} \mathbb{E}\left[\sum_{t=1}^{T} (U_{a_t}(t) - \mu_{a_t}) + \left(\mu_{a_t^*} - U_{a_t^*}(t)\right) \middle| E\right] + 4KMT^2\delta \\
&\overset{(b)}{\leq} 2\mathbb{E}\left[\sum_{k=1}^{K}\sum_{t=1}^{T} \mathbb{1}\{A_t = k\} \left(\frac{2C(1+\epsilon,\alpha)}{\delta n_k(t)^\epsilon}\right)^{\frac{1}{1+\epsilon}}\right] + 4KMT^2\delta \\
&= 2\left(\frac{2C(1+\epsilon,\alpha)}{\delta}\right)^{\frac{1}{1+\epsilon}} \mathbb{E}\left[\sum_{k=1}^{K}\sum_{t=1}^{T} \mathbb{1}\{A_t = k\} \left(\frac{1}{n_k(t)^\epsilon}\right)^{\frac{1}{1+\epsilon}}\right] + 4KMT^2\delta \\
&\overset{(c)}{\leq} 2\left(\frac{2C(1+\epsilon,\alpha)}{\delta}\right)^{\frac{1}{1+\epsilon}} \mathbb{E}\left[\sum_{k=1}^{K} \int_{s=0}^{n_k(T)} \left(\frac{1}{s^\epsilon}\right)^{\frac{1}{1+\epsilon}} ds\right] + 4KMT^2\delta \\
&= 2(1+\epsilon)\left(\frac{2C(1+\epsilon,\alpha)}{\delta}\right)^{\frac{1}{1+\epsilon}} \mathbb{E}\left[\sum_{k=1}^{K} n_k(T)^{\frac{1}{1+\epsilon}}\right] + 4KMT^2\delta \\
&\overset{(d)}{\leq} 4\left(\frac{2C(1+\epsilon,\alpha)}{\delta}\right)^{\frac{1}{1+\epsilon}} \mathbb{E}\left[K^{\frac{1}{1+\epsilon}} \left(\sum_{k=1}^{K} n_k(T)\right)^{\frac{1}{1+\epsilon}}\right] + 4KMT^2\delta \\
&\overset{(e)}{\leq} 4\left(\frac{2C(1+\epsilon,\alpha)}{\delta}\right)^{\frac{1}{1+\epsilon}} (KT)^{\frac{1}{1+\epsilon}} + 4KMT^2\delta.
\end{aligned}$$

Here, (a) follows from Lemma 16, (b) follows from event E: whenever E holds, each term inside the summation is bounded by Equation (5.13), (b) follows from the upper bound of a finite discrete sum with a definite integral, (c) follows from Hölder's Inquality of order $\frac{1}{1+\epsilon}$ and (d) follows from $T = 1 + \sum_{k=1}^{K} n_k(T)$, and that $1 + \epsilon < 2$. By setting $\delta = 1/T$, we have:

For a finite time analysis, we can now choose ϵ such that it minimizes the RHS.

Therefore,

$$\text{Bayes Regret}(T, \pi^{TS}) \leq \inf_{\epsilon<\alpha-1} \left\{ 4\left(2C(1+\epsilon,\alpha)\right)^{\frac{1}{1+\epsilon}} \left(KT^2\right)^{\frac{1}{1+\epsilon}} + 4KMT \right\}$$

Asymptotically, we see that for ϵ chosen *a priori* close to $\alpha - 1$, when $T > K^{\frac{\epsilon}{1-\epsilon}}$,

$$\text{Bayes Regret}(T, \pi^{TS}) = O(K^{\frac{1}{1+\epsilon}} T^{\frac{2}{1+\epsilon}})$$

□

We note the following: First, the only additional assumption we make on the reward distributions is the boundedness of the means, which is a standard assumption [74]. Next, given the established polynomial deviations of the empirical mean, obtanining a polylogarithmic regret is not possible, and the selection of ϵ governs the finite-time regret. As $\epsilon \to \alpha - 1$, we see that while the growth of $T^{\frac{2}{1+\epsilon}}$ decreases, $C(1+\epsilon, \alpha)$ grows, and is not finite at $\epsilon = \alpha - 1$. This behavior arises from the non-compactness of the set of finite moments for α-stable distributions.

Contrasted to the problem-independent regret bound of $O(\sqrt{KT \log T})$ for Thompson Sampling on the multi-armed Gaussian bandit problem demonstrated by [5], our bound differs in two aspects: first, we admit a $K^{\frac{1}{1+\epsilon}}$ complexity on the number of arms, which contrasted with the Gaussian bandit is identical when $\epsilon \to 1$. Second, we have a superlinear dependence of order $T^{\frac{2}{1+\epsilon}}$. Compared to the Gaussian case, we see that (i) we obtain a polynomial regret instead of a polylogarithmic regret, which can be attributed to the polynomial concentration of heavy-tailed distributions. Furthermore, the second term of $T^{\frac{1}{1+\epsilon}}$ approaches the Gaussian case ($\sqrt{T}$) when $\alpha \to 1$, leaving us with the additional sub-linear term ($T^{\frac{1}{1+\epsilon}}$) compared to the sub-logarithmic term ($\sqrt{\log T}$) from the Gaussian case.

In the next section, we address the issue of polynomial concentration by utilizing the more robust, truncated mean estimator instead of the empirical mean, and obtain a modified, robust version of α-TS.

Algorithm 3 Robust α-Thompson Sampling

Input: Arms $k \in [K]$, priors $\mathcal{N}(\mu_k^0, \sigma^2)$ for each arm.
Set $D_k = 1, N_k = 0$ for each arm k.
for For each iteration $t \in [1, T]$ **do**
  Draw $\bar{\mu}_k(t) \sim \mathcal{N}\left(\frac{\mu_k^0 + N_k}{D_k}, \frac{\sigma^2}{D_k}\right)$ for each arm k.
  Choose arm $A_t = \arg\max_k \bar{\mu}_k(t)$, and get reward r_t.
  If $|r_t| > \left(\frac{H(\epsilon,\alpha,\sigma)\cdot i}{2\log(T)}\right)^{\frac{1}{1+\epsilon}}$, set $r_t = 0$.
  for $q \in [0, Q)$ **do**
    Calculate $v_{A_t}^{(t)} = r_t - \bar{\mu}_{A_t}$.
    Sample $\lambda_k^{(t)}$ following Section 5.2.3.
    Set $D_q = D_k + 1/\lambda_k^{(t)}, N_q = N_k + r_t/\lambda_k^{(t)}$.
    Sample $\bar{\mu}_{A_t} \sim \mathcal{N}\left(\frac{\mu_k^0 + N_q}{D_q}, \frac{\sigma^2}{D_q}\right)$.
  end for
  Set $D_k = D_k + 1/\lambda_k^{(t)}, N_k = N_k + r_t/\lambda_k^{(t)}$.
end for

5.3 The Robust α-Thompson Sampling Algorithm

In this section, we will utilize a similar truncated mean estimator to derive an algorithm that provides a much tighter regret bound on the bandit problem.

5.3.1 Truncated Mean Estimator

Assume that for all arms $k \in [K], \mu_k \leq M$. Note that this assumption is equivalent to the boundedness assumption in the analysis of α-TS, and is a reasonable assumption to make in any practical scenario with some domain knowledge of the problem. Let $\delta \in (0,1), \epsilon \in (0, \alpha - 1)$. Now, consider the truncated mean estimator $\hat{r}_k^*(t)$ given by:

$$\hat{r}_k^*(t) = \frac{1}{n_k(t)} \sum_{i=1}^{n_k(t)} r_k^{(i)} \mathbb{1}\left\{|r_k^{(i)}| \leq \left(\frac{H(\epsilon,\alpha,\sigma)\cdot i}{2\log(T)}\right)^{\frac{1}{1+\epsilon}}\right\}$$
$$\text{where, } H(\epsilon,\alpha,\sigma) = \left(\frac{\epsilon\left(M\cdot\Gamma(-\epsilon/\alpha) + \sigma\alpha\Gamma(1-\frac{\epsilon+1}{\alpha})\right)}{\sigma\alpha\sin\left(\frac{\pi\cdot\epsilon}{2}\right)\Gamma(1-\epsilon)}\right) \quad (5.14)$$

$\hat{r}_k^*(t)$ then describes a truncated mean estimator for an arm k (pulled $n_k(t)$ times), where a reward $r_k^{(i)}$ at any trial i of the arm is discarded if it is larger than the bound.

Intuitively, we see that this truncated mean will prevent outliers from affecting the posterior. We choose such a form of the truncation since it allows us to obtain an exponential concentration for α-stable densities, which will assist us in obtaining tighter regret, as in the case of the α-UCB algorithm.

5.3.2 Regret Analysis

The corresponding Robust α-TS algorithm is identical to the basic α-TS algorithm, except for this step of rejecting a reward (and replacing it with 0) and is outlined in Algorithm 3. We now describe the regret incurred by this modified α-TS algorithm.

Theorem 7 (Regret Bound). *Let $K > 1, \alpha \in (1,2), \sigma \in \mathbb{R}^+, \mu_{k:k\in[K]} \in [-M, M]$. For a K-armed bandit with rewards for each arm k drawn from $S_\alpha(0, \sigma, \mu_k)$, we have, for ϵ chosen a priori such that $\epsilon \to (\alpha - 1)^-$ and truncated estimator given in Equation (5.14),*

$$\textit{Bayes Regret}(T, \pi^{RTS}) = \tilde{O}\left((KT)^{\frac{1}{1+\epsilon}}\right)$$

Proof. Consider a K-armed bandit with rewards for arm k drawn from $S_\alpha(0, \sigma, \mu_k)$. Let $n_k(t)$ denote the number of times arm k has been pulled until time t. Then $t - 1 = \sum_{k=1}^{K} n_k(t)$. Let us denote the empirical average reward for arm k up to (and including) time $t-1$ as $\hat{r}_k(t)$, and denote the arm pulled at any time t as a_t, and the optimal arm as a_t^*. We then set an upper confidence bound for arm k at any time t as

$$U_k(t) = \text{clip}_{[-M,M]}\left[\hat{r}_k(t) + 4\sigma H(\epsilon, \alpha, \sigma)^{\frac{1}{1+\epsilon}}\left(\frac{\log(2/\delta)}{n_k(t)}\right)^{\frac{\epsilon}{1+\epsilon}}\right] \tag{5.15}$$

for $0 < \epsilon < 1, M > 0$. Let E be the event when for all $k \in [K]$ arms, over all iterations $t \in [T]$, we have:

$$|\hat{r}_k(t) - \mu_k| \leq 4\sigma H(\epsilon, \alpha, \sigma)^{\frac{1}{1+\epsilon}}\left(\frac{\log(2/\delta)}{n_k(t)}\right)^{\frac{\epsilon}{1+\epsilon}}. \tag{5.16}$$

Lemma 18. *For the setup described above, we have, for event E and $\delta \in (0,1)$,*

$$\mathbb{P}(E^c) \leq KT\delta.$$

Proof. The event E^c holds whenever the bound is violated for at least one arm k at one instance t. Therefore,

$$\begin{aligned}
\mathbb{P}(E^c) &\leq \mathbb{P}\left(\bigcup_{\substack{k=1\\t=1}}^{K,T}\left\{|\hat{r}_k(t-1)-\mu_k| > 4\sigma H(\epsilon,\alpha,\sigma)^{\frac{1}{1+\epsilon}}\left(\frac{\log(2/\delta)}{n_k(t)}\right)^{\frac{\epsilon}{1+\epsilon}}\right\}\right)\\
&\overset{(a)}{\leq} \sum_{\substack{k=1\\t=1}}^{K,T}\mathbb{P}\left(|\hat{r}_k(t-1)-\mu_k| > 4\sigma H(\epsilon,\alpha,\sigma)^{\frac{1}{1+\epsilon}}\left(\frac{\log(2/\delta)}{n_k(t)}\right)^{\frac{\epsilon}{1+\epsilon}}\right)\\
&\overset{(b)}{\leq} KT\delta.
\end{aligned}$$

Where (a) is an application of the union bound, and (b) is obtained using Lemma 14. □

We now prove Theorem 7 in a similar manner as Theorem 6, via Lemma 17. By the tower rule, we can condition over event E:

$$\begin{aligned}
\text{Bayes Regret}(T,\pi^{RTS}) = \mathbb{E}\left[\sum_{t=1}^{T}(U_{a_t}(t)-\mu_{a_t})+\left(\mu_{a_t^*}-U_{a_t^*}(t)\right)\middle|E\right]\mathbb{P}(E)+\\
\mathbb{E}\left[\sum_{t=1}^{T}(U_{a_t}(t)-\mu_{a_t})+\left(\mu_{a_t^*}-U_{a_t^*}(t)\right)\middle|E^c\right]\mathbb{P}(E^c) \quad (5.17)
\end{aligned}$$

Since $\mathbb{P}(E) \leq 1$,

$$\begin{aligned}
\text{Bayes Regret}(T,\pi^{RTS}) \leq \mathbb{E}\left[\sum_{t=1}^{T}(U_{a_t}(t)-\mu_{a_t})+\left(\mu_{a_t^*}-U_{a_t^*}(t)\right)\middle|E\right]+\\
\mathbb{E}\left[\sum_{t=1}^{T}(U_{a_t}(t)-\mu_{a_t})+\left(\mu_{a_t^*}-U_{a_t^*}(t)\right)\middle|E^c\right]\mathbb{P}(E^c) \quad (5.18)
\end{aligned}$$

When E^c holds, each term in the summation in the conditional expectation is bounded by $4M$ (Equation 5.15). Therefore,

$$
\begin{aligned}
\text{Bayes Regret}(T, \pi^{RTS}) &\leq 4MT \cdot \mathbb{P}(E^c) + \mathbb{E}\left[\sum_{t=1}^{T} (U_{a_t}(t) - \mu_{a_t}) + \left(\mu_{a_t^*} - U_{a_t^*}(t)\right) \middle| E\right] \\
&\overset{(a)}{\leq} 4KMT^2\delta + \mathbb{E}\left[\sum_{t=1}^{T} (U_{a_t}(t) - \mu_{a_t}) + \left(\mu_{a_t^*} - U_{a_t^*}(t)\right) \middle| E\right] \\
&\overset{(b)}{\leq} 4KMT^2\delta + 8\mathbb{E}\left[\sum_{k=1}^{K}\sum_{t=1}^{T} \mathbb{1}\{A_t = k\}\sigma H(\epsilon,\alpha,\sigma)^{\frac{1}{1+\epsilon}} \left(\frac{\log(2/\delta)}{n_k(t)}\right)^{\frac{\epsilon}{1+\epsilon}}\right] \\
&= 4KMT^2\delta + 8\sigma H(\epsilon,\alpha,\sigma)^{\frac{1}{1+\epsilon}} \log(2/\delta)^{\frac{\epsilon}{1+\epsilon}} \mathbb{E}\left[\sum_{k=1}^{K}\sum_{t=1}^{T} \mathbb{1}\{A_t = k\} \left(\frac{1}{n_k(t)^\epsilon}\right)^{\frac{1}{1+\epsilon}}\right] \\
&\overset{(c)}{\leq} 4KMT^2\delta + 8\sigma H(\epsilon,\alpha,\sigma)^{\frac{1}{1+\epsilon}} \log(2/\delta)^{\frac{\epsilon}{1+\epsilon}} \mathbb{E}\left[\sum_{k=1}^{K} \int_{s=0}^{n_k(T)} \left(\frac{1}{s^\epsilon}\right)^{\frac{1}{1+\epsilon}} ds\right] \\
&= 4KMT^2\delta + 8(1+\epsilon)\sigma H(\epsilon,\alpha,\sigma)^{\frac{1}{1+\epsilon}} \log(2/\delta)^{\frac{\epsilon}{1+\epsilon}} \mathbb{E}\left[\sum_{k=1}^{K} n_k(T)^{\frac{1}{1+\epsilon}}\right] \\
&\overset{(d)}{\leq} 4KMT^2\delta + 16\sigma H(\epsilon,\alpha,\sigma)^{\frac{1}{1+\epsilon}} \log(2/\delta)^{\frac{\epsilon}{1+\epsilon}} \mathbb{E}\left[K^{\frac{1}{1+\epsilon}} \left(\sum_{k=1}^{K} n_k(T)\right)^{\frac{1}{1+\epsilon}}\right] \\
&\overset{(e)}{\leq} 4KMT^2\delta + 16\sigma H(\epsilon,\alpha,\sigma)^{\frac{1}{1+\epsilon}} \log(2/\delta)^{\frac{\epsilon}{1+\epsilon}} (KT)^{\frac{1}{1+\epsilon}} \\
&\overset{(f)}{=} 4KM + 16\sigma H(\epsilon,\alpha,\sigma)^{\frac{1}{1+\epsilon}} \left(\log(2)^{\frac{\epsilon}{1+\epsilon}}\right) + 32\sigma H(\epsilon,\alpha,\sigma)^{\frac{1}{1+\epsilon}} \log(T)^{\frac{\epsilon}{1+\epsilon}} (KT)^{\frac{1}{1+\epsilon}}.
\end{aligned}
$$

Here, (a) follows from Lemma 18, (b) follows from event E: whenever E holds, each term inside the summation is bounded by Equation (5.16), (c) follows from the upper bound of a finite discrete sum with a definite integral, (d) follows from Hölder's Inquality of order $\frac{1}{1+\epsilon}$ and (e) follows from $T = 1 + \sum_{k=1}^{K} n_k(T)$, and that $1 + \epsilon < 2$, and (f) is obtained by setting $\delta = 1/T^2$. Asymptotically,for ϵ chosen *a priori* close to $\alpha - 1$,

$$
\text{Bayes Regret}(T, \pi^{RTS}) = O((KT)^{\frac{1}{1+\epsilon}} (\log T)^{\frac{\epsilon}{1+\epsilon}}) = \tilde{O}\left((KT)^{\frac{1}{1+\epsilon}}\right).
$$

□

We see that this bound is tight: when $\alpha = 2$ (Gaussian), ϵ can be set to 1, and we see that this matches the optimal bound of $O(\sqrt{KT})$ [5] (up to logarithmic factors).

Algorithm 3 is hence more robust, with performance improvements increasing as α decreases: the likelihood of obtaining confounding outliers increases as $\alpha \to 1$, and can perturb the posterior mean in the naive α-TS algorithm.

Now that we have established regret bounds for these algorithms, we move on to our next problem setting, after which we will discuss experimental results using these algorithms.

Chapter 6

Multi-Agent Stochastic Bandits

Our application scenario focuses on the problem where $M > 1$ agents are solving unique decision problems but there is a network structure embedded among these different decision problems. We consider a generalized version of the multi-agent stochastic bandit problem, where M agents play the same set of K arms, however, each agent $m \in [M]$ has different utilities (average reward μ_k^m) from each arm. This introduces additional complexity in the decision-making from the agent's perspective, in the form of attention. As it observes the rewards and actions its neighbors take, it must accept or discard their observations based on how similar (or different) its own preferences are from its neighbors.

To provide intuition, recall the problem of online advertising. Each "agent" in this scenario is faced with the task of determining the optimal advertisement from a set of K websites (actions) for a particular website from a total of M websites (hence, M agents in total). In isolation, with the assumption that advertising preferences are stationary, this is an example of a classic stochastic bandit problem[1].

However, there is an underlying network structure between websites – either through hyperlinks from pages, or via the amount of common visitors to a pair of websites. This leads us to believe that it may be possible to leverage this underlying network structure, if encoded in an accurate manner, to improve the performance

[1]Typically, this is a case of a linear or contextual bandit problem, however, for this thesis we consider the stochastic bandit problem as an illustrative example.

from the isolated single-player setting.

A close variant of this setting has been analysed in a related context. The literature of bandits with **side observations** focuses on a separate problem: there is a set of $\mathcal{K}$ arms that are arranged on a network, and pulling any arm $i \in \mathcal{K}$ provides *side observations* from $\mathcal{K}_i$ arms in the neighborhood of arm i. There has been extensive work in providing efficient algorithms in this context for the stochastic bandit case [20, 21, 25, 36, 87] and the linear bandit case [6, 7].

The multi-agent version of the *side observation* problem differs in a few key aspects: first, this assumes that each player will have the same set of $\mathcal{K}$ arms over a fixed network, whereas our problem involves each agent having a *distinct* decision problem. Secondly, in the former setting, for each round of the game, each player will receive side observations for the arm they pull, whereas in our problem, while the agents *do* receive observations from their neighbors, but they are *of the arms the neighbors pull* and not necessarily of their own arm. In terms of the bandit problem involved, the side observation problem assumes a graph structure over a set of common actions, whereas in our problem, each node in the graph structure is an agent with its own bandit problem.

In context of the advertisement setting, our problem corresponds to learning the advertising preferences for each website or user from a set of M users given the similarity graph between users, whereas the side observation problem corresponds to learning the users similarity graph itself. In summary, this problem can be thought of as a generalization of the purely co-operative multi-agent bandit problem, where each agent acts individually to minimize its regret, however, co-operates by providing its observations to neighboring users.

In the next section, we formalize the associated bandit problem and the communication protocol.

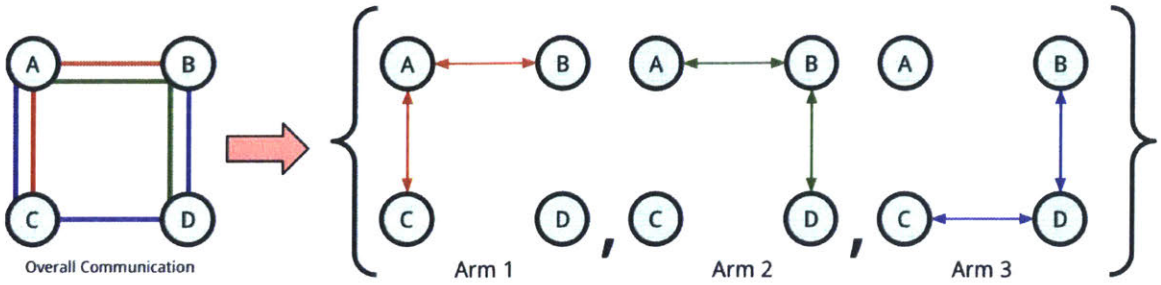

Figure 6-1: The generalized network bandit problem for 4 agents over 3 actions (denoted as red, green and blue arrows respectively). Note that the mean rewards are distinct for each agent and each action.

6.1 The Generalized Network Bandit Problem

Consider M agents each solving a different K-armed unstructured bandit problem, where rewards are drawn i.i.d. from their respective distributions. Hence, for two agents i and j, and any arm $k \in [K]$, μ_k^i is not necessarily equal to μ_k^j. We now state the key assumption about the nature of preferences (average rewards) among agents.

Assumption 2 (Closeness Assumption). *In the generalized network bandit problem, there exist K associated undirected "similarity" networks $\mathcal{G}_k, k \in [K]$ with associated sets of edges $\mathcal{E}_1, ..., \mathcal{E}_K$ such that for all $k \in [K], (i,j) \in [M] \times [M]$,*

$$(i,j) \in \mathcal{E}_k \implies |\mu_k^i - \mu_k^j| \leq \epsilon_k, \text{ and, agents } i \text{ and } j \text{ can communicate.}$$

Where ϵ_k is a constant that is known in advance from the construction of the networks.

This property of closeness of rewards leads to the terminology of the Generalized Network problem. The problem definition implies that, for each arm $k \in [K]$, there is a separate neighborhood that is determined by the corresponding edge set $\mathcal{E}_k$. To illustrate, an example problem for 4 agents over 3 actions is described in Figure 6-1. Note that we do not assume that the underlying reward distributions are necessarily identical.

This problem definition is the most straightforward encapsulation of the intuition delivered in the previous section. More general constraints, such as a constraint on the

distributional closeness (under some form of divergence) of the reward distributions can be imposed in future work, but are omitted here.

The central problem statement is to provide a mechanism for each agent to utilize supplementary observations to accelerate their own learning, and determine the cases in which learning can be improved. It is evident that during the initial phases of training, any agent will prefer to utilize the additional observations from its neighbors to decrease the variance of the estimate (at a cost to the bias), but later on, once it has gathered enough samples, it will prefer to use its own observations entirely. In the subsequent sections, we will formalize this heuristic and develop optimal algorithms for the problem.

6.2 The Net-UCB Algorithm

The generalized network bandit problem proceeds as follows. At any iteration $t \in [T]$, each agent $m \in [M]$ plays an arm $a_t^m \in [K]$ and observes a reward $r_m(t)$. In addition to the rewards, the agent receives observations of the actions and rewards from each of its neighboring arms, based on their actions. Let the neighborhood of arm k for agent m be denoted by $\mathcal{G}_k^m \subseteq \mathcal{G}_k$. Then the agent receives a set of observations $\mathcal{O}_m(t) = \cup_{k\in[K]} \left(\{r_{m'}(t)\ \forall m' : m' \in \mathcal{G}_k^m \cap a_t^{m'} = k\}\right)$. Let us also assume two databases $D(\cdot) : \mathbb{R} \to [K]$ and $E(\cdot) : \mathbb{R} \to [M]$ that record the arm and agent from which each member of $\mathcal{O}_m(t)$ was received.

We denote the number of times agent m chose action k until time t as $n_m^k(t)$, and the corresponding set of rewards obtained until and including time t for arm k as $\mathcal{A}_k^m(t)$. Now, let $\mathcal{S}_k^m(t)$ denote the set of all observations for arm k obtained by agent m until and including time t, i.e. $\mathcal{S}_k^m(t) = \cup_{u=1}^t \{z \in \mathcal{O}_m(t) | D(z) = k\}$. We can see that $N_k^m(t) = |\mathcal{A}_k^m(t) \cup \mathcal{S}_k^m(t)| = \sum_{m' \in \mathcal{G}_k^m \cup \{m\}} n_k^{m'}(t)$, i.e. the total number of rewards obtained by agent m for arm k is the sum of the number of times each agent in the neighborhood of arm k for agent m (including itself) has pulled the arm.

Consider the mean reward $X_k^m(t)$ from all agents in the neighborhood of action k

for agent m.

$$\hat{X}_k^m(t) = \frac{\left(\sum_{x \in \mathcal{S}_k^m(t) \cup \mathcal{A}_k^m(t)} x\right)}{N_k^m(t)}. \tag{6.1}$$

We can now use the closeness property on this random variable,

$$\left|\mathbb{E}[\hat{X}_k^m(t)] - \mu_k^m\right| = \left|\frac{1}{N_k^m(t)} \sum_{x \in \mathcal{S}_k^m(t) \cup \mathcal{A}_k^m(t)} \left(\mu_k^{E(x)} - \mu_k^m\right)\right| \tag{6.2}$$

$$\leq \epsilon_k \tag{6.3}$$

The central idea of the algorithm is to use the tighter estimate of the mean reward (in terms of variance) using the additional observations obtained from the neighborhood estimate of rewards. However, as seen above, there is a maximum non-zero bias of ϵ_k that is introduced by utilizing this estimate. Therefore, there is a tradeoff between the bias and variance of the reward estimate. As the arm is explored more, it will rely less on the neighborhood mean reward, and operate on its own estimate during exploitation.

We will now construct upper confidence bounds for sub-Gaussian reward distributions, and outline the associated algorithm for each. We first outline a few common definitions.

Definition 5 (Clique Cover). *Consider graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. A clique cover $\mathcal{C}$ of $\mathcal{G}$ is a cover of $\mathcal{G}$ such that each partition $C \in \mathcal{C}$ forms a complete subgraph. A minimum clique cover is a clique cover that uses as few cliques as possible.*

6.2.1 Algorithm Description

In this section we will describe the algorithm for the case of sub-Gaussian rewards. For each arm $k \in [K]$, consider a clique cover $\mathcal{C}_k$ of its associated connectivity graph $\mathcal{G}_k$. Then, for any agent m, arm k must be in at least one clique $Z \in \mathcal{C}$. If it is in multiple cliques, consider the largest clique C it is a part of in $\mathcal{C}$. The **clique mean reward** $\hat{X}_k^C$ for arm k of agent m is the average of all observations it has from this

clique, including its own.

$$X_k^C(t) = \frac{\sum_{m \in C} \sum_{x \in \mathcal{A}_k^m(t)} x}{\sum_{m \in C} n_k^m(t)} \tag{6.4}$$

Note that this reward is common for all agents residing in C, and hence there is no dependence of the notation on any specific agent $m \in C$. Let us denote the total number of times arm k has been pulled by any agent within this clique until and including time t as $N_k^C(t) = \sum_{m \in C} n_k^m(t)$. Now, from the previous section, we know that for all agents $m \in C$,

$$\left|\mathbb{E}[X_k^C(t)] - \mu_k^m\right| \leq \epsilon_k. \tag{6.5}$$

Let us now derive a concentration result under the assumption that the rewards are σ-sub-Gaussian.

Lemma 19 (Concentration of clique mean). *For $X_k^C(t)$ as defined above, we have, with probability at least $1 - \delta$, for all $m \in C$,*

$$\left|X_k^C(t) - \mu_k^m\right| \leq \epsilon_k + \sigma\sqrt{\frac{2\ln\left(\frac{1}{\delta}\right)}{N_k^C(t)}} \tag{6.6}$$

Proof. We know that $X_k^C(t)$ is a sum of N_k^C σ-sub-Gaussian random variables. Therefore, the proxy variance of $X_k^C(t) = \frac{\sigma^2}{N_k^C}$. Now, applying the result of Lemma 2, we have, with probability at least $1 - \delta$

$$|X_k^C(t) - \mathbb{E}[X_k^C(t)|] \leq \sigma\sqrt{\frac{2\ln\left(\frac{1}{\delta}\right)}{N_k^C(t)}} \tag{6.7}$$

Now, we have, by the triangle inequality,

$$\left|X_k^C(t) - \mu_k^m\right| \leq \left|\mathbb{E}[X_k^C(t)] - \mu_k^m\right| + \left|\mathbb{E}[X_k^C(t)] - X_k^C(t)\right| \tag{6.8}$$

Hence, with probability at least $1 - \delta$,

$$\leq \epsilon_k + \sigma\sqrt{\frac{2\ln\left(\frac{1}{\delta}\right)}{N_k^C(t)}} \tag{6.9}$$

□

Now, for arm k of agent m, we have two upper confidence bounds for the mean reward μ_k^m. The first one is obtained directly from Lemma 2, since the reward distribution is σ-sub-Gaussian. Let us denote this as the local confidence bound for arm k of agent m, UCBL_k^m. This bound has no bias, but since it uses only the samples from the agent, it has a high variance.

$$\text{UCBL}_k^m(t) = \hat{\mu}_k^m(t-1) + \sigma\sqrt{\frac{2\ln(t)}{n_k^m(t-1)}} \tag{6.10}$$

The second confidence bound is obtained from the clique mean reward, and is stated in the earlier Lemma. This bound introduces bias, however it has a much lower variance than the first bound, since it is obtained using all the samples present in the neighborhood. Let us denote this bound as the clique confidence bound UCBC_k^m.

$$\text{UCBC}_k^m(t) = X_k^C(t-1) + \epsilon + \sigma\sqrt{\frac{2\ln(t)}{N_k^C(t-1)}} \tag{6.11}$$

It is evident that the nature of this problem poses a bias-variance tradeoff in exploration. Typically, we would desire the agent to explore with lower variance, with some bias, and once the variance has decreased, the agent can exploit with the local confidence bound. We therefore develop the algorithm Net-UCB as follows. We will assume some agent $m \in [M]$ and consider $C \in \mathcal{C}_k$ to be the largest clique of which it is a member of.

Let $\hat{\mu}_k^m(t-1) = \frac{\sum_{x \in \mathcal{A}_k^m(t)} x}{n_k^m(t)}$ denote the empirical mean of rewards up to and including time t for arm $k \in [K]$. At any iteration $t \in [T]$, for each arm $k \in [K]$, if $X_k^C(t-1) > \hat{\mu}_k^m(t-1)$, the agent first samples a Bernoulli random variable

$\delta_m^k(t) \sim \mathcal{B}(\frac{1}{|C|})$. If $\delta_m^k = 0$,

$$\text{UCB}_k^m = \text{UCBC}_k^m \tag{6.12}$$

Otherwise, the agent sets

$$\text{UCB}_k^m = \text{UCBL}_k^m \tag{6.13}$$

Finally, the agent choses the arm with the largest confidence bound.

$$a_t^m = \arg\max_{k \in [K]} \text{UCB}_k^m \tag{6.14}$$

The algorithm can be interpreted as first conducting complete exploration, followed by an optimistic exploitation. We now proceed to the theoretical analysis.

6.2.2 Regret Analysis

Theorem 8. *Consider the generalized bandit problem over M agents and K arms. If the reward distribution for arm k is σ-sub-Gaussian, and for all arms $k \in [K]$, the corresponding connectivity graph $\mathcal{G}_k$ imposes a mean closeness constraint of $\epsilon_k < \frac{\min_{m \in [M]} \Delta_k^m}{2}$, then the following holds for the Net-UCB algorithm:*

$$R_G(T) \leq \inf_{\mathcal{C}_1, \ldots, \mathcal{C}_k} \left\{ \sum_{k \in [K]} \sum_{c \in \mathcal{C}_k} 8\sigma^2 \ln(T) \left(\frac{1}{\Delta_k^{C_{\min}}} + \left(1 - \frac{1}{|C|}\right) \frac{\Delta_k^{C_{\max}}}{(\Delta_k^{C_{\min}} - 2\epsilon_k)^2} \right) + O(K) \right\}.$$

The infimum for $\mathcal{C}_i$ is taken over all possible clique covers of the connectivity graph $\mathcal{G}_i$.

Proof. Consider arm $k \in [K]$. The associated connectivity graph between arms is given by $\mathcal{G}_k$. Now, let $\mathcal{C}_k$ be any arbitrary clique cover of $\mathcal{G}_k$. Let us consider the group regret $R_k^C(T)$ over a clique $C \in \mathcal{C}_k$ for arm k. Since the total regret for arm k, $R_k^G(T) \leq \sum_{C \in \mathcal{C}} R_k^C(T)$, bounding this quantity will provide us a bound over the

group regret.

$$R_k^C(T) = \sum_{m \in C} \Delta_k^m \mathbb{E}[n_k^m(T)] \tag{6.15}$$

$$= \sum_{m \in C} \Delta_k^m \sum_{t=1}^{T} \Pr\left\{a_t^m = k\right\}. \tag{6.16}$$

Now, arm can be chosen either with the local or clique confidence bound. Therefore,

$$= \sum_{m \in C} \Delta_k^m \sum_{t=1}^{T} \Pr\left\{a_t^m = k, \text{local bound}\right\} + \Pr\left\{a_t^m = k, \text{clique bound}\right\}. \tag{6.17}$$

$$\leq \frac{1}{|C|} \left(\sum_{m \in C} \Delta_k^m \sum_{t=1}^{T} \Pr\left\{a_t^m = k_{loc}\right\} + \Pr\left\{a_t^m = k_{cli}\right\} (|C| - 1) \right). \tag{6.18}$$

We will now bound the first summation.

$$\sum_{m \in C} \Delta_k^m \sum_{t=1}^{T} \Pr\left\{a_t^m = k_{loc}\right\}. \tag{6.19}$$

$$\leq \sum_{m \in C} \Delta_k^m \sum_{t=1}^{T} \Pr\left\{a_t^m = k_{loc}\right\} \tag{6.20}$$

Each agent can be considered to be playing a single-agent bandit problem. Hence, by the result of Theorem 2,

$$\leq \sum_{m \in C} \Delta_k^m \left(\frac{8\sigma^2 \ln(T)}{(\Delta_k^m)^2} + \left(1 + \frac{\pi^2}{3}\right) \right) \tag{6.21}$$

$$\tag{6.22}$$

We will now bound the second summation.

$$(|C| - 1) \left(\sum_{m \in C} \Delta_k^m \sum_{t=1}^{T} \Pr\left\{a_t^m = k_{cli}\right\} \right) \tag{6.23}$$

For some positive integer η_C, we have,

$$\leq (|C|-1)\left(\eta_C \cdot \max_{m\in C}\Delta_k^m + \sum_{m\in C}\Delta_k^m \sum_{t=1}^{T}\Pr\left\{a_t^m = k_{cli}, N_k^C(t-1)\geq \eta_C\right\}\right) \tag{6.24}$$

$$= (|C|-1)\left(\eta_C \cdot \max_{m\in C}\Delta_k^m + \sum_{m\in C}\Delta_k^m \sum_{t=1}^{T}\Pr\left\{\mathrm{UCBC}_k^m(t) > \mathrm{UCB}_{k_m^*}^m(t), N_k^C(t-1)\geq \eta_C\right\}\right) \tag{6.25}$$

$$\leq (|C|-1)\left(\eta_C \cdot \max_{m\in C}\Delta_k^m + \sum_{m\in C}\Delta_k^m \sum_{t=1}^{T}\Pr\left\{\max_{s\leq t}\mathrm{UCBC}_k^m(t) > \min_{s\leq t}\mathrm{UCB}_{k_m^*}^m(t), N_k^C(t-1)\geq \eta_C\right\}\right) \tag{6.26}$$

$$\leq (|C|-1)\left(\eta_C \cdot \max_{m\in C}\Delta_k^m + \sum_{m\in C}\Delta_k^m \sum_{t=1}^{T}\sum_{s=0}^{t}\Pr\left\{\mathrm{UCBC}_k^m(s) > \mathrm{UCB}_{k_m^*}^m(s), N_k^C(t-1)\geq \eta_C\right\}\right) \tag{6.27}$$

Now, for $\mathrm{UCBC}_k^m(t) > \mathrm{UCB}_{k_m^*}^m(t)$ to be true at any time t, one of the following must be true:

$$\text{Event 1: } \mathrm{UCB}_{k_m^*}^m \leq \mu_{k_m^*}^m \tag{6.28}$$

$$\text{Event 2: } \hat{X}_m^k(t-1) - \epsilon - \sigma\sqrt{\frac{2\ln(t)}{N_k^C(t-1)}} \leq \mu_k^m \tag{6.29}$$

$$\text{Event 3: } \mu_{k_m^*}^m - \mu_k^m < 2\left(\epsilon_k + \sigma\sqrt{\frac{2\ln(t)}{N_k^C(t-1)}}\right) \tag{6.30}$$

For Event 1, during time t, $\mathrm{UCB}_{k_m^*}^m$ may be set to only one of $\mathrm{UCBC}_{k_m^*}^m$ or $\mathrm{UCBL}_{k_m^*}^m$. In the former case, by setting $\delta = \frac{1}{t^2}$ in Lemma 19 we know that this occurs with probability at most $1/2t^2$. In the latter case, we can set $\delta = \frac{1}{t^2}$ in Lemma 2 to achieve the same probability bound. Since only one of those two events occur, the probability of Event 1 is at most $1/2t^2$.

For Event 2, by setting $\delta = \frac{1}{t^2}$ in Lemma 19 we know that this occurs with

probability at most $1/2t^2$. Let us now examine Event 3.

$$\mu_{k_m^*}^m - \mu_k^m < 2\left(\epsilon_k + \sigma\sqrt{\frac{2\ln(t)}{N_k^C(t-1)}}\right) \tag{6.31}$$

$$\implies \frac{\Delta_k^m}{2} - \epsilon_k \leq \sigma\sqrt{\frac{2\ln(t)}{N_k^C(t-1)}} \tag{6.32}$$

Since $\Delta_k^m > 2\epsilon_k \ \forall k \in [K]$,

$$\implies \left(\frac{\Delta_k^m}{2} - \epsilon_k\right)^2 \leq \frac{2\sigma^2\ln(t)}{N_k^C(t-1)} \tag{6.33}$$

$$\implies N_k^C(t-1) \leq \frac{8\sigma^2\ln(t)}{(\Delta_k^m - 2\epsilon_k)^2}. \tag{6.34}$$

Hence, Event 3 only occurs with non-zero probability as long as $N_k^C(t-1)$ is at most $\frac{8\sigma^2\ln(t)}{(\Delta_k^m-2\epsilon_k)^2}$. By setting $\eta_C = \lceil \frac{8\sigma^2\ln(T)}{(\min_{m\in C}\Delta_k^m-2\epsilon_k)^2} \rceil$ we can ensure that Event 3 does not occur. Now, we can bound $\Pr\left\{\mathrm{UCBC}_k^m(t) > \mathrm{UCB}_{k_m^*}^m(t), N_k^C(t-1) \geq \eta_C\right\}$.

$$\Pr\left\{\mathrm{UCBC}_k^m(t) > \mathrm{UCB}_{k_m^*}^m(t), N_k^C(t-1) \geq \eta_C\right\} \tag{6.35}$$

$$= \sum_{i=1}^{3} \Pr\left\{\mathrm{UCBC}_k^m(t) > \mathrm{UCB}_{k_m^*}^m(t), N_k^C(t-1) \geq \eta_C, (\text{Event } i \text{ occurs})\right\} \tag{6.36}$$

$$\leq \frac{1}{t^2}. \tag{6.37}$$

Putting it all together, we have,

$$(|C|-1)\left(\eta_C \cdot \max_{m\in C}\Delta_k^m + \sum_{m\in C}\Delta_k^m \sum_{t=1}^{T}\sum_{s=0}^{t} \Pr\left\{\mathrm{UCBC}_k^m(s) > \mathrm{UCB}_{k_m^*}^m(s), N_k^C(t-1) \geq \eta_C\right\}\right) \tag{6.38}$$

$$\leq (|C|-1)\left(\lceil\frac{8\sigma^2\ln(T)}{(\min_{m\in C}\Delta_k^m - 2\epsilon_k)^2}\rceil \cdot \max_{m\in C}\Delta_k^m + \sum_{m\in C}\Delta_k^m\frac{\pi^2}{3}\right) \tag{6.39}$$

$$\leq (|C|-1)\left(\frac{8\sigma^2\ln(T)\max_{m\in C}\Delta_k^m}{(\min_{m\in C}\Delta_k^m - 2\epsilon_k)^2} + \max_{m\in C}\Delta_k^m + \frac{\pi^2\sum_{m\in C}\Delta_k^m}{3}\right) \tag{6.40}$$

Finally, combining both confidence bounds, we obtain the following bound for the

clique regret for arm k. Let $\Delta_k^{C_{\max}} = \max_{m \in C} \Delta_k^m$ and $\Delta_k^{C_{\min}} = \min_{m \in C} \Delta_k^m$.

$$R_k^C(T) \leq 8\sigma^2 \ln(T) \left(\frac{1}{\Delta_k^{C_{\min}}} + \left(1 - \frac{1}{|C|}\right) \frac{\Delta_k^{C_{\max}}}{(\Delta_k^{C_{\min}} - 2\epsilon_k)^2} \right) + 2\Delta_k^{C_{\max}} + 2 \sum_{m \in C} \Delta_k^m \left(1 + \frac{\pi^2}{3}\right) \tag{6.41}$$

Summing up over all arms and cliques and taking an infimum over all possible clique covers, we have, for the total group regret,

$$R_G(T) \leq \inf_{\mathcal{C}_1, \ldots, \mathcal{C}_k} \left\{ \sum_{k \in [K]} \sum_{c \in \mathcal{C}_k} 8\sigma^2 \ln(T) \left(\frac{1}{\Delta_k^{C_{\min}}} + \left(1 - \frac{1}{|C|}\right) \frac{\Delta_k^{C_{\max}}}{(\Delta_k^{C_{\min}} - 2\epsilon_k)^2} \right) + O(K) \right\}.$$

□

We see that asymptotically, the algorithm obtains an optimal regret order of $O(\ln T)$. Additionally, our bound depends on the number of cliques in the minimal cover of the connectivity graphs. Since this is equal to $\chi(\bar{\mathcal{G}}_k)$ denotes the chromatic number of the complement graph of $\mathcal{G}_k$, Our algorithm obtains a dependence of $O\left(\sum_{k \in [K]} \chi(\bar{\mathcal{G}}_k)\right)$.

The chromatic number determines the smallest number of colors required to color the graph without sharing colors between adjacent vertices. If the communication graph $\mathcal{G}_k$ is densely connected, the complement graph $\bar{\mathcal{G}}_k$ is sparsely connected, we will require few colors, which implies that the algorithm will incur small regret.

On the other hand, if the communication graph is sparsely connected, the complement graph will be dense, lending it a higher chromatic number. In effect, the algorithm will incur higher regret.

This behavior is expected, since in the case when the network is sparse, the algorithm will have weaker estimates from its neighborhood, and observations from agents will not lend much benefit. In denser graphs, the agents will have more communication, and hence the algorithm will converge sooner.

In the next chapter, we will discuss experimental results that verify the tight regret bounds achieved by our algorithm.

Chapter 7

Experiments

In this section, we discuss the experimentals results for the algorithms developed in the last two chapters. All our implementations are on multi-core Google Cloud virtual servers, and are implemented in Python using NumPy. Since generating clique covers is NP-Hard, we use approximate clique covers using the implementation in NetworkX.

7.1 Heavy-Tailed Bandits

To compare heavy-tailed reward settings, we select the following benchmark algorithms.

- **ε-greedy**. Here, an agent chooses the arm with the maximum mean reward with probability $1 - \varepsilon$, and chooses a random arm with probability ε.
- **Gaussian-TS**. Regular Thompson Sampling with Gaussian priors and a Gaussian assumption on the data. This is to compare the effect of a non-robust algorithm on heavy-tailed data.
- **α-UCB**. This is our version of Robust-UCB [17] for heavy-tailed distributions using a truncated mean estimator, adapted to α-stable densities.
- **α-TS**. Our first Bayesian algorithm for α-stable densities, this does not use a robust mean estimator. We set the iterations for sampling of λ (Q) as 50.

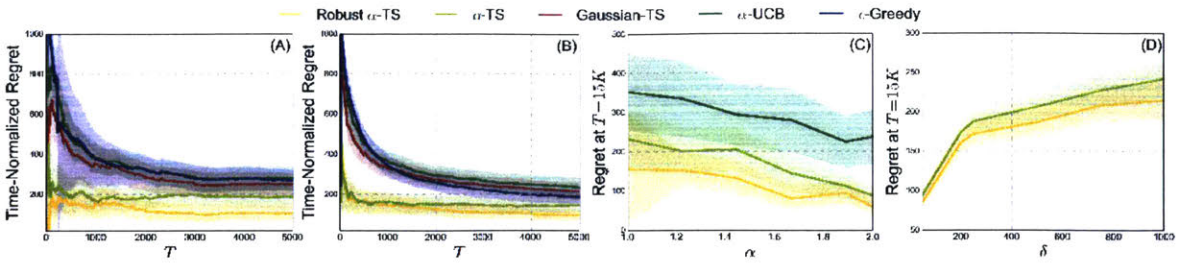

Figure 7-1: Empirical performance of the $\alpha-$UCB, $\alpha-$TS, and Robust $\alpha-$TS algorithms. (A) Competitive benchmarking for $\alpha = 1.3$, and (B) $\alpha = 1.8$; (C) Ablation studies for varying α, and (D) varying prior strength. Shaded areas denote variance scaled by 0.25 in (A) and (B), and scaled by 0.5 in (C) and (D).

- **Robust α-TS**. Our second Bayesian algorithm for α-stable densities, this uses a robust mean estimator. We set the iterations for sampling of λ (Q) as 50.

Setting Priors for TS: In all the Thompson Sampling benchmarks, setting the priors are crucial to the algorithm's performance. In the competitive benchmarking, we randomly set the priors for each arm from the same range we use for setting the mean rewards.

7.1.1 Performance against Competitive Benchmarks

We run 100 MAB experiments each for all 5 benchmarks for $\alpha = 1.8$ and $\alpha = 1.3$, and $K = 50$ arms, and for each arm, the mean is drawn from $[0, 2000]$ randomly for each experiment, and $\sigma = 2500$. Each experiment is run for $T = 5000$ iterations, and we report the regret averaged over time, i.e. $\mathbb{R}(t)/t$ at any time t.

In Figures 7-1A and 7-1B, we see the results of the regret averaged over all 100 experiments. We observe that for $\alpha = 1.3$, there are more substantial variations in the regret (low α implies heavy outliers), yet both algorithms comfortably outperform the other baselines.

In the case of $\alpha = 1.8$, the variations are not that substantial, the performance follows the same trend. It is important to note that regular Thompson Sampling (Gaussian-TS) performs competitively, although in our experiments, we observed that when K is large, the algorithm often concentrates on the incorrect arm, and subsequently earns a larger regret.

Intuitively, we can see that whenever arms have mean rewards close to each other (compared to the variance), ϵ-greedy will often converge to the incorrect arm. α-UCB, however, makes very weak assumptions on the data distributions, and hence has a much larger exploration phase, leading to larger regret. Compared to α-TS, Robust α-TS is less affected by outlying rewards (as is more evident in $\alpha = 1.3$ vs. $\alpha = 1.8$) and hence converges faster.

7.1.2 Ablation Studies

Effect of α

First, we compare the performance of α-TS on the identical set up as before (same K and reward distributions), but with varying values of $\alpha \in (1, 2)$. We report the expected time-normalized regret averaged over 100 trials in Figure 7-1C, and observe that (i) as α increases, the asymptotic regret decreases faster, and (ii) as expected, for lower values of α there is a substantial amount of variation in the regret.

Effect of Closeness in the Priors for TS

Secondly, we compare the effect of the closeness of the priors – i.e., how close the prior beliefs are to the true average rewards. In the previous experiments, the priors are drawn randomly from the same range as the means, without any additional information. However, by introducing more information about the means through the priors, we can expect better performance. In this experiment, for each mean μ_k, we randomly draw the prior mean μ_k^0 from $[\mu_k - \delta, \mu_k + \delta]$, and observe the regret after $T = 15K$ trials for δ from 50 to 1000. The results for this experiment are summarized in Figure 7-1D for $K = 10$ and $\sigma = 25$, and results are averaged over 25 trials each. We see that with uninformative priors, α-TS performs competitively, and only gets better as the priors get sharper.

7.2 Generalized Network Bandits

For the generalized bandit problem, we examine our Net-UCB algorithm against the following benchmarks.

- ε-**greedy**. Here, each agent chooses the arm with the maximum mean reward with probability $1 - \varepsilon$, or chooses a random arm with probability ε.
- **Net-UCB**. This is the proposed algorithm from the previous section. We supply each agent with the largest clique they are part of, and run the algorithm without any additional parameters.
- **UCB-LP**. This is a variant of the bandits with side observations algorithm proposed in [21], where each agent runs the single-agent UCB-LP algorithm with side observations coming from one neighboring arm in their clique randomly. While the algorithm requires rewards coming from all neighboring arms at each iteration, since this is not applicable to our setting, we replace it with one 'clique' arm that provides one observation each from the clique itself.
- **Coop-UCB**. This is the Cooperative UCB algorithm, which is a multiagent algorithm proposed in [51]. This algorithm assumes equal rewards for arms across agents (cooperative multi-agent bandit setting), which we do not alter, and we provide the union of the communication graphs for each arm as the overall communication graph.
- **UCB-1**. This is the naive UCB-1 algorithm, proposed first in [10]. Each agent independently runs the UCB-1 algorithm without utilizing network observations at all.

These benchmarks were chosen to compare across all aspects of the problem:

- We compare with the baseline ε-greedy algorithm to compare naive greedy decision-making in the presence of multiple side observations from the communication network.

Algorithm 4 Generating Rewards for Each Arm

Input: Arms $k \in [K]$, Graphs $\mathcal{G}_1, ..., \mathcal{G}_K$, Constraints $\epsilon_1, ..., \epsilon_K$, Range for μ: $[-M, M]$.
for each graph $\mathcal{G}_k \in \{G_1, ..., \mathcal{G}_K\}$ **do**
  Compute set of connected components $\mathcal{Z}$ in $\mathcal{G}_k$.
  for each connected component $Z \in \mathcal{Z}$ **do**
    Draw μ^z_{base} uniformly from $[-M, M]$.
    for each vertex m in Z **do**
      Draw δ^m_k uniformly from $[-\epsilon_k/2, -\epsilon_k/2]$.
      Set $\mu^m_k = \mu^z_{base} + \delta^m_k$.
    end for
  end for
end for

- We compare with the UCB-LP algorithm since it does not natively incorporate the multi-agent setting, and relies on additional observations entirely. In contrast to their problem setting, here, we may have different agents behaving differently based on their own optimal arm, and hence the probability of getting samples from a desired arm consistently is lower.

- We compare with the Coop-UCB algorithm to visualize the effects of having different individual reward distributions for each agent. This generalization of the co-operative problem setting may introduce non co-operative behavior between agents that have different optimal arms.

- Finally, we compare with UCB-1 to observe the improvement in performance from the most naive baseline algorithm, where the clique observations are ignored entirely.

Graph Generation

Since we do not have a connectedness requirement in the connectivity graphs, we generate graphs randomly for each arm $k \in [K]$ from the following families of random graphs.

- **Erdos-Renyi Graphs**. This family of graphs are produced by first setting the number of nodes, and then randomly inserting links between pairs of nodes with

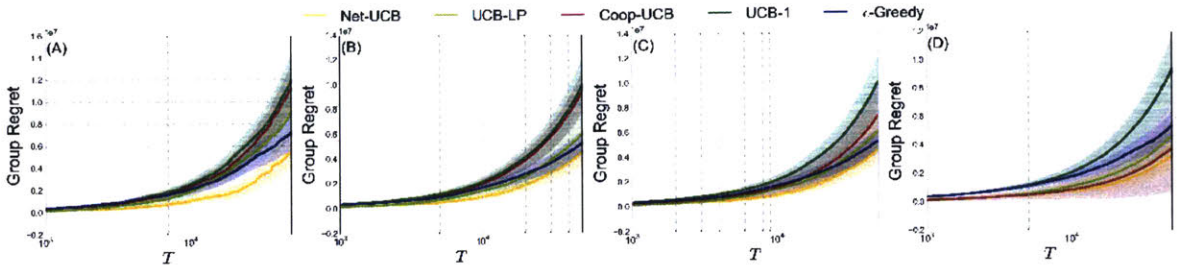

Figure 7-2: Empirical performance of the Net-UCB algorithm against competitive benchmarks across different communication graph types: (A) Erdos-Renyi Graphs with $p = 0.8$, (B) Erdos-Renyi Graphs with $p = 0.2$, (C) Scale-Free Graphs, (D) Small-World Graphs. All networks have $M = 100$ agents, and the time axis is scaled logarithmically.

probability $p \in (0, 1)$. We consider $p = 0.2$ and $p = 0.8$ for our experiments.

- **Scale-Free Graphs**. This family of graphs follows a power-law distribution of edges, which closely resemble properties of real-world social networks. We follow the pipeline in [26], where the power-law factor is set to 2.5.

- **Small-World Graphs**. This family of graphs ensure that the maximum distance between any two nodes is roughly logarithmic in the number of total nodes. These graphs are typically not dense since most nodes simply have only two edges, but no two nodes are a large distance apart. We follow the pipeline for generation outlined by [64].

Setting Rewards for Each Arm

Given the closeness constraints imposed by the generalized bandit problem, the process of setting rewards for each arm is non-trivial. We outline our algorithm in Algorithm 4. Primarily, for each arm k, we consider the associated connectivity graph $\mathcal{G}_k$, and compute its connected components. For each connected component, we first randomly sample a baseline mean from the given range, and then for each node within that connected component, we sample a change factor within $[-\epsilon_k/2, \epsilon_k/2]$. Then for each node, we construct the mean reward by adding the baseline mean to the change factor. This provides us with a network such that no two nodes that are neighbors in $\mathcal{G}_k$ have a difference in means larger than ϵ_k.

We now describe the individual experiments for each graph type. Since there are no tunable parameters for our algorithm, there are no specific ablation studies that we can perform.

7.2.1 Effect of Type of Connectivity Graph

In Figure 7-2, we describe the performance against competitive benchmarks across different families of networks. For each experiment, we set $\sigma = 500$, $M = 1000$ and randomly sample ϵ_k from the range $(0, 200]$. We repeat each experiment a total of 200 times across randomly generated graphs. We observe the following:

1. In all graph types, we observe that our algorithm performs substantially better (lower regret) among all algorithms. This behavior is expected, since our algorithm accounts for both i) varying individual rewards, and ii) multiagent communication, which none of the baselines do. Additionally, across all experiments, UCB-1 performs the worst, which is also expected since it does not account for multi-agent communication at all.

2. In Scale-Free graphs, we observe that the margins of improvement are smaller. This can be attributed to the power-law network structure, which introduces a large variation in the size of cliques. This implies that some nodes in the network are very sparsely connected, and hence take less advantage of multi-agent communication.

3. In Small-World graphs, the Coop-UCB algorithm performs at par with our algorithm. This is because the Coop-UCB algorithm, while assuming identical rewards, relies on the complete communication network and not local communication. Since small-world graphs have low average path lengths, this boosts the algorithm's performance, whereas our algorithm does not benefit from this structure. However, the identical reward assumption also makes the Coop-UCB algorithm less stable, and hence it has a much higher variance in its group regret compared to other algorithms.

Algorithm	Per-Agent Regret at T=50K			
	M=10	M=50	M=100	M=200
UCB-1	12581.24 (± 876.23)	12448.96 (± 797.65)	12699.44 (± 815.94)	12348.06 (± 801.15)
UCB-Coop	10801.68 (± 704.52)	10054.24 (± 698.54)	9945.32 (± 677.10)	8556.46 (± 608.35)
LP-UCB	8779.56 (± 359.96)	6099.31 (± 401.59)	5145.12 (± 340.03)	4669.35 (± 277.42)
Net-UCB	**6080.49 (± 308.85)**	**3369.35(± 301.96)**	**2708.59(± 279.09)**	**2195.29(± 276.55)**

Table 7.1: Comparison of different multi-agent and side-observation algorithms with Net-UCB as agents are varied. Note: reported metric is per-agent regret, so lower is better.

In conclusion, our algorithm consistently provides better performance on the generalized network bandit problem. This is in line with our regret bounds, which guarantee a smaller regret than the individual algorithms discussed here.

7.2.2 Effect of Number of Agents

For any multi-agent algorithm, it is imperative that behavior with increasing number of agents grows at least sub-linearly with the number of agents, as any naive single-agent algorithm can provide linear growth with the number of agents.

In the experiment summarized in Table 7.1, we compare the per-agent regret (group regret normalized by total number of agents) for the various UCB algorithms on Erdos-Renyi graphs with $p = 0.8$ with the identical generation setup as the previous one. We select Erdos-Renyi graphs specifically since we can explicitly control the average number of edges in these networks via the density parameter p.

We repeat the comparative experiment for 200 trials, and report averaged numbers. There are two primary observations:

1. We see that the performance of the Net-UCB algorithm provides an average increase of 7.2× in the group regret when the number of agents is increased by 190. Comparatively, the next most competitive algorithm, LP-UCB, provides an increase of 10.1×. This confirms the predicted performance as guaranteed by our regret bound.

2. In addition to average regret, we observe that our algorithm consistently exhibits a lower variance in average regret compared to the other baselines. This suggests

Algorithm	Group Regret $\times 10^6$ at T=50K			
	p=0.1	p=0.4	p=0.8	p=1.0
UCB-1	1.257 (± 0.095)	1.234 (± 0.087)	1.269 (± 0.081)	1.238 (± 0.080)
UCB-Coop	1.181 (± 0.072)	0.951 (± 0.068)	0.853 (± 0.060)	0.725 (± 0.052)
LP-UCB	0.704 (± 0.035)	0.608 (± 0.042)	0.514 (± 0.034)	0.388 (± 0.027)
Net-UCB	**0.648 (± 0.030)**	**0.456(± 0.030)**	**0.276(± 0.028)**	**0.195(± 0.021)**

Table 7.2: Comparison of different multi-agent and side-observation algorithms with Net-UCB as the average number of edges are increased. Note: reported metric is per-agent regret, so lower is better.

that our network is robust to configurations of edges in networks as long as the number of edges is constant (in expectation).

Finally, we discuss the effects of network density on group regret in the next section.

7.2.3 Effect of Average Degree of Communication

The performance of our algorithm depends critically on the number of minimal cliques of the connectivity graphs. This dependence arises from the formulation of the regret bound over the minimal clique cover. This is a function of the chromatic number $\chi(\bar{\mathcal{G}}_k)$ of the complement graph of each connectivity graph. This function decreases (in expectation) as the number of edges increase in the graph.

More intuitively, as the number of edges increase, the average size of the cliques will increase, since more vertices will be connected to each other. This implies that the number of cliques will decrease, and hence, the maximum regret will as well. From the point of view of network structure, when the number of edges increases, each agent will (in expectation), have more neighbors to leverage additional observations from, and hence will learn faster.

We use Erdos-Renyi graphs for this experiment as well, since we have direct control over the expected number of edges via the connectivity parameter p. In fact the expected number of edges for an M-vertex Erdos-Renyi graph with parameter p is given by Mp, and hence the relationship is linear. We compare the performance of all algorithms as we increase the connectedness of the communication graphs, and

summarize the results in Table 7.2. Parameters for this experiment are identical to the previous ones, with $M = 100$. The primary takeaways of this experiment are as follows:

1. Our algorithm consistently outperforms all other baselines, as in all earlier experiments. Additionally, we observe a $4\times$ decrease in regret as the network goes from sparsely-connected ($p = 0.1$) to fully-connected ($p = 1.0$), which is also larger than any other baseline.

2. Our algorithm consistently exhibits lower variance in regret, which can be attributed to the robustness of the algorithm to individual edge configurations of the graphs.

In summary, we conclude the proficiency of our algorithm across a variety of evaluative criteria. We do not compare the performance with variations in the standard deviation of rewards (σ^2) or the number of arms since the dependence of our algorithm on these quantities are identical in complexity to the baseline UCB algorithm of [10]. In the next section, we provide closing remarks and discuss the future avenues of work inspired by this thesis.

Chapter 8

Closing Remarks

In this thesis, we first designed a framework for efficient posterior inference for the α-stable family, which has largely been ignored in the bandits literature owing to its intractable density function. We formulated the first polynomial problem-independent regret bounds for Thompson Sampling with α-stable densities, and subsequently improved the regret bound to achieve the optimal regret identical to the sub-Gaussian case, providing an efficient framework for decision-making for these distributions.

Next, we considered the problem of network bandits with local communication and unique individual preferences. This problem fuses the seemingly disparate problem domains of multi-agent bandits and bandits with side-observations. We introduce Net-UCB, an optimal algorithm for this problem, and provide rigorous performance guarantees for the same.

We believe that the general nature of this problem setting will usher in new theoretical and applied research in multi-agent network bandits, leading to better algorithms for advertising, recommendation systems, and personalized machine learning.

Additionally, our intermediary concentration results provide a starting point for other machine learning problems that may be investigated in α-stable settings. There is ample evidence to support the existence of α-stability in various modeling problems across economics [34], finance [14] and behavioral studies [61].

With tools such as ours, we hope to usher scientific conclusions in problems that cannot make sub-Gaussian assumptions, and can lead to more robust empirical find-

ings. Some future work may include viewing more involved decision-making processes, such as MDPs, in the same light, leading to more (distributionally) robust algorithms.

Our work on stochastic bandits can lay the foundations for the development of several more sophisticated algorithms, such as linear and contextual bandits, and reinforcement learning in heavy-tailed and networked settings.

As stated earlier, the ultimate goal of this thesis is to establish the need for algorithms that can account for network structure and extreme uncertainty that is extant in real-world environments, and usher in the development of robust multi-agent systems that can eventually integrate seamlessly into a collaborative human-machine future.

Bibliography

[1] Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. In *Advances in Neural Information Processing Systems*, pages 2312–2320, 2011.

[2] Dhaval Adjodah, Dan Calacci, Abhimanyu Dubey, Anirudh Goyal, Peter Krafft, Esteban Moro, and Alex Pentland. Communication topologies between learning agents in deep reinforcement learning. *CoRR*, abs/1902.06740, 2019.

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

[4] Shipra Agrawal and Navin Goyal. Analysis of thompson sampling for the multi-armed bandit problem. In *Conference on Learning Theory*, pages 39–1, 2012.

[5] Shipra Agrawal and Navin Goyal. Further optimal regret bounds for thompson sampling. In *Artificial Intelligence and Statistics*, pages 99–107, 2013.

[6] Noga Alon, Nicolo Cesa-Bianchi, Ofer Dekel, and Tomer Koren. Online learning with feedback graphs: Beyond bandits. In *Annual Conference on Learning Theory*, volume 40. Microtome Publishing, 2015.

[7] Noga Alon, Nicolo Cesa-Bianchi, Claudio Gentile, Shie Mannor, Yishay Mansour, and Ohad Shamir. Nonstochastic multi-armed bandits with graph-structured feedback. *SIAM Journal on Computing*, 46(6):1785–1826, 2017.

[8] Noga Alon, Yossi Matias, and Mario Szegedy. The space complexity of approximating the frequency moments. *Journal of Computer and system sciences*, 58(1):137–147, 1999.

[9] David F Andrews and Colin L Mallows. Scale mixtures of normal distributions. *Journal of the Royal Statistical Society: Series B (Methodological)*, 36(1):99–102, 1974.

[10] Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2-3):235–256, 2002.

[11] Peter J Bickel et al. On some robust estimates of location. *The Annals of Mathematical Statistics*, 36(3):847–858, 1965.

[12] Szymon Borak, Wolfgang Härdle, and Rafał Weron. Stable distributions. In *Statistical tools for finance and insurance*, pages 21–44. Springer, 2005.

[13] Gauvain Bourgne, Amal El Fallah Segrouchni, and Henry Soldano. Smile: Sound multi-agent incremental learning. In *Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems*, page 38. ACM, 2007.

[14] Brendan O Bradley and Murad S Taqqu. Financial risk and heavy tails. In *Handbook of heavy tailed distributions in finance*, pages 35–103. Elsevier, 2003.

[15] Peter F Brown, Vincent J Della Pietra, Stephen A Della Pietra, and Robert L Mercer. The mathematics of statistical machine translation: Parameter estimation. *Computational linguistics*, 19(2):263–311, 1993.

[16] Ivan Brugere, Brian Gallagher, and Tanya Y Berger-Wolf. Network structure inference, a survey: Motivations, methods, and applications. *ACM Computing Surveys (CSUR)*, 51(2):24, 2018.

[17] Sébastien Bubeck, Nicolo Cesa-Bianchi, and Gábor Lugosi. Bandits with heavy tail. *IEEE Transactions on Information Theory*, 59(11):7711–7717, 2013.

[18] Sébastien Bubeck and Che-Yu Liu. Prior-free and prior-dependent regret bounds for thompson sampling. In *Advances in Neural Information Processing Systems*, pages 638–646, 2013.

[19] Sébastien Bubeck, Gilles Stoltz, and Jia Yuan Yu. Lipschitz bandits without the lipschitz constant. In *International Conference on Algorithmic Learning Theory*, pages 144–158. Springer, 2011.

[20] Swapna Buccapatnam, Atilla Eryilmaz, and Ness B Shroff. Multi-armed bandits in the presence of side observations in social networks. In *52nd IEEE Conference on Decision and Control*, pages 7309–7314. IEEE, 2013.

[21] Swapna Buccapatnam, Atilla Eryilmaz, and Ness B Shroff. Stochastic bandits with side observations on networks. *ACM SIGMETRICS Performance Evaluation Review*, 42(1):289–300, 2014.

[22] Apostolos N Burnetas and Michael N Katehakis. Optimal adaptive policies for sequential allocation problems. *Advances in Applied Mathematics*, 17(2):122–142, 1996.

[23] Robert R Bush and Frederick Mosteller. A stochastic model with applications to learning. *The Annals of Mathematical Statistics*, pages 559–585, 1953.

[24] Olivier Cappé, Aurélien Garivier, Odalric-Ambrym Maillard, Rémi Munos, Gilles Stoltz, et al. Kullback–leibler upper confidence bounds for optimal sequential allocation. *The Annals of Statistics*, 41(3):1516–1541, 2013.

[25] Stéphane Caron, Branislav Kveton, Marc Lelarge, and Smriti Bhagat. Leveraging side observations in stochastic bandits. *arXiv preprint arXiv:1210.4839*, 2012.

[26] Michele Catanzaro, Marián Boguná, and Romualdo Pastor-Satorras. Generation of uncorrelated random scale-free networks. *Physical review e*, 71(2):027103, 2005.

[27] Olivier Catoni. Challenging the empirical mean and empirical variance: a deviation study. In *Annales de l'IHP Probabilités et statistiques*, volume 48, pages 1148–1185, 2012.

[28] John M Chambers, Colin L Mallows, and BW Stuck. A method for simulating stable random variables. *Journal of the american statistical association*, 71(354):340–344, 1976.

[29] Olivier Chapelle and Lihong Li. An empirical evaluation of thompson sampling. In *Advances in neural information processing systems*, pages 2249–2257, 2011.

[30] Andreas Christmann, Ingo Steinwart, and Arnout van Messem. On consistency and robustness properties of support vector machines for heavy-tailed distributions. *Statistics and Its Interface*, 2(3):311–327, 2009.

[31] Vu C Dinh, Lam S Ho, Binh Nguyen, and Duy Nguyen. Fast learning rates with heavy-tailed losses. In *Advances in neural information processing systems*, pages 505–513, 2016.

[32] Nathan Eagle, Alex Sandy Pentland, and David Lazer. Inferring friendship network structure by using mobile phone data. *Proceedings of the national academy of sciences*, 106(36):15274–15278, 2009.

[33] Alhussein Fawzi, Omar Fawzi, and Pascal Frossard. Analysis of classifiers' robustness to adversarial perturbations. *Machine Learning*, 107(3):481–508, 2018.

[34] John C Frain. *Studies on the Application of the Alpha-stable Distribution in Economics.*

[35] Man Gao, Ling Chen, Bin Li, Yun Li, Wei Liu, and Yong-cheng Xu. Projection-based link prediction in a bipartite network. *Information Sciences*, 376:158–171, 2017.

[36] Claudio Gentile, Shuai Li, and Giovanni Zappella. Online clustering of bandits. In *International Conference on Machine Learning*, pages 757–765, 2014.

[37] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.

[38] Simon Godsill and Ercan E Kuruoglu. Bayesian inference for time series with heavy-tailed symmetric α-stable noise processes. 1999.

[39] Alfonso González-Briones, Gabriel Villarrubia, Juan F De Paz, and Juan M Corchado. A multi-agent system for the classification of gender and age from images. *Computer Vision and Image Understanding*, 172:98–106, 2018.

[40] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[41] Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 855–864. ACM, 2016.

[42] Peter D Grünwald and Nishant A Mehta. Fast rates for general unbounded loss functions: from erm to generalized bayes. *arXiv preprint arXiv:1605.00252*, 2016.

[43] Martin Hellman and Josef Raviv. Probability of error, equivocation, and the chernoff bound. *IEEE Transactions on Information Theory*, 16(4):368–372, 1970.

[44] Daniel Hsu and Sivan Sabato. Heavy-tailed regression with a generalized median-of-means. In *International Conference on Machine Learning*, pages 37–45, 2014.

[45] Levente Kocsis and Csaba Szepesvári. Bandit based monte-carlo planning. In *European conference on machine learning*, pages 282–293. Springer, 2006.

[46] Stanley Kok and Pedro Domingos. Extracting semantic networks from text via relational clustering. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 624–639. Springer, 2008.

[47] Nathaniel Korda, Emilie Kaufmann, and Remi Munos. Thompson sampling for 1-dimensional exponential family bandits. In *Advances in Neural Information Processing Systems*, pages 1448–1456, 2013.

[48] Tze Leung Lai and Herbert Robbins. Asymptotically efficient adaptive allocation rules. *Advances in applied mathematics*, 6(1):4–22, 1985.

[49] Peter Landgren, Vaibhav Srivastava, and Naomi Ehrich Leonard. Distributed cooperative decision-making in multiarmed bandits: Frequentist and bayesian algorithms. In *2016 IEEE 55th Conference on Decision and Control (CDC)*, pages 167–172. IEEE, 2016.

[50] Peter Landgren, Vaibhav Srivastava, and Naomi Ehrich Leonard. On distributed cooperative decision-making in multiarmed bandits. In *2016 European Control Conference (ECC)*, pages 243–248. IEEE, 2016.

[51] Peter Landgren, Vaibhav Srivastava, and Naomi Ehrich Leonard. Social imitation in cooperative multiarmed bandits: Partition-based algorithms with strictly local information. In *2018 IEEE Conference on Decision and Control (CDC)*, pages 5239–5244. IEEE, 2018.

[52] P Lévy. Calcul des probabilités, vol. 9. *Gauthier-Villars Paris*, 1925.

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

[54] Keqin Liu and Qing Zhao. Distributed learning in cognitive radio networks: Multi-armed bandit with distributed multiple players. In *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 3010–3013. IEEE, 2010.

[55] Keqin Liu and Qing Zhao. Distributed learning in multi-armed bandit with multiple players. *IEEE Transactions on Signal Processing*, 58(11):5667–5681, 2010.

[56] Keqin Liu, Qing Zhao, and Bhaskar Krishnamachari. Decentralized multi-armed bandit with imperfect observations. In *2010 48th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 1669–1674. IEEE, 2010.

[57] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

[58] Setareh Maghsudi and Ekram Hossain. Multi-armed bandits with application to 5g small cells. *IEEE Wireless Communications*, 23(3):64–73, 2016.

[59] Setareh Maghsudi and Ekram Hossain. Distributed user association in energy harvesting dense small cell networks: A mean-field multi-armed bandit approach. *IEEE Access*, 5:3513–3523, 2017.

[60] Setareh Maghsudi and Sławomir Stańczak. Joint channel selection and power control in infrastructureless wireless networks: A multiplayer multiarmed bandit framework. *IEEE Transactions on Vehicular Technology*, 64(10):4565–4578, 2015.

[61] Aniket Mahanti, Niklas Carlsson, Anirban Mahanti, Martin Arlitt, and Carey Williamson. A tale of the tails: Power-laws in internet measurements. *IEEE Network*, 27(1):59–64, 2013.

[62] Muneya Matsui, Zbyněk Pawlas, et al. Fractional absolute moments of heavy tailed distributions. *Brazilian Journal of Probability and Statistics*, 30(2):272–298, 2016.

[63] Andres Munoz Medina and Scott Yang. No-regret algorithms for heavy-tailed linear bandits. In *International Conference on Machine Learning*, pages 1642–1650, 2016.

[64] Mozart BC Menezes, Seokjin Kim, and Rongbing Huang. Constructing a watts-strogatz network from a small-world network with symmetric degree distribution. *PloS one*, 12(6):e0179120, 2017.

[65] Raymond J Mooney. Learning for semantic parsing. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 311–324. Springer, 2007.

[66] Seth Myers and Jure Leskovec. On the convexity of latent social network inference. In *Advances in neural information processing systems*, pages 1741–1749, 2010.

[67] MEJ Newman. Network structure from rich but noisy data. *Nature Physics*, 14(6):542, 2018.

[68] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. Learning deconvolution network for semantic segmentation. In *Proceedings of the IEEE international conference on computer vision*, pages 1520–1528, 2015.

[69] Franz Josef Och. Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, pages 160–167. Association for Computational Linguistics, 2003.

[70] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics, 2002.

[71] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 701–710. ACM, 2014.

[72] Hoifung Poon and Pedro Domingos. Unsupervised semantic parsing. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1*, pages 1–10. Association for Computational Linguistics, 2009.

[73] Richardson Ribeiro, André P Borges, and Fabricio Enembreck. Interaction models for multiagent reinforcement learning. In *2008 International Conference on Computational Intelligence for Modelling Control & Automation*, pages 464–469. IEEE, 2008.

[74] Daniel Russo and Benjamin Van Roy. Learning to optimize via posterior sampling. *Mathematics of Operations Research*, 39(4):1221–1243, 2014.

[75] Shahin Shahrampour, Alexander Rakhlin, and Ali Jadbabaie. Multi-armed bandits in multi-agent networks. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2786–2790. IEEE, 2017.

[76] Han Shao, Xiaotian Yu, Irwin King, and Michael R Lyu. Almost optimal algorithms for linear stochastic bandits with heavy-tailed payoffs. In *Advances in Neural Information Processing Systems*, pages 8430–8439, 2018.

[77] Min Shao and Chrysostomos L Nikias. Signal processing with fractional lower order moments: stable processes and their applications. *Proceedings of the IEEE*, 81(7):986–1010, 1993.

[78] Jacopo Staiano, Bruno Lepri, Nadav Aharony, Fabio Pianesi, Nicu Sebe, and Alex Pentland. Friends don't lie: inferring personality traits from social network structure. In *Proceedings of the 2012 ACM conference on ubiquitous computing*, pages 321–330. ACM, 2012.

[79] Ambuj Tewari and Susan A Murphy. From ads to interventions: Contextual bandits in mobile health. In *Mobile Health*, pages 495–517. Springer, 2017.

[80] William R Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3/4):285–294, 1933.

[81] Sattar Vakili, Keqin Liu, and Qing Zhao. Deterministic sequencing of exploration and exploitation for multi-armed bandit problems. *IEEE Journal of Selected Topics in Signal Processing*, 7(5):759–767, 2013.

[82] Peng Wang, BaoWen Xu, YuRong Wu, and XiaoYu Zhou. Link prediction in social networks: the state-of-the-art. *Science China Information Sciences*, 58(1):1–38, 2015.

[83] Yu-Xiong Wang, Ross Girshick, Martial Hebert, and Bharath Hariharan. Low-shot learning from imaginary data. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7278–7286, 2018.

[84] Yu-Xiong Wang, Deva Ramanan, and Martial Hebert. Learning to model the tail. In *Advances in Neural Information Processing Systems*, pages 7029–7039, 2017.

[85] Gerhard Weiß. Adaptation and learning in multi-agent systems: Some remarks and a bibliography. In *International Joint Conference on Artificial Intelligence*, pages 1–21. Springer, 1995.

[86] Wikipedia. Stable distribution — Wikipedia, the free encyclopedia, 2019. [Online; accessed 25-February-2019].

[87] Yifan Wu, András György, and Csaba Szepesvári. Online learning with gaussian payoffs and side observations. In *Advances in Neural Information Processing Systems*, pages 1360–1368, 2015.

[88] Yi Xu, Shenghuo Zhu, Sen Yang, Chi Zhang, Rong Jin, and Tianbao Yang. Learning with non-convex truncated losses by sgd. *arXiv preprint arXiv:1805.07880*, 2018.

[89] Mo Yu, Xiaoxiao Guo, Jinfeng Yi, Shiyu Chang, Saloni Potdar, Yu Cheng, Gerald Tesauro, Haoyu Wang, and Bowen Zhou. Diverse few-shot text classification with multiple metrics. *arXiv preprint arXiv:1805.07513*, 2018.

[90] Xiaotian Yu, Han Shao, Michael R Lyu, and Irwin King. Pure exploration of multi-armed bandits with heavy-tailed payoffs.

[91] Boyao Zhu and Yongxiang Xia. An information-theoretic model for link prediction in complex networks. *Scientific reports*, 5:13707, 2015.

[92] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017.