Ph.D. DISSERTATION

SAMPLE EFFICIENT ROBOT LEARNING FOR OPTIMAL DECISION MAKING UNDER COMPLEX AND UNCERTAIN ENVIRONMENTS

복잡하고 불확실한 환경에서 최적 의사 결정을 위한
효율적인 로봇 학습

BY

KYUNGJAE LEE

FEBRUARY 2021

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING
COLLEGE OF ENGINEERING
SEOUL NATIONAL UNIVERSITY

Sample-Efficient Robot Learning for Optimal Decision Making under Complex and Uncertain Environments

복잡하고 불확실한 환경에서 최적 의사 결정을 위한 효율적인 로봇 학습

지도교수 오 성 회

이 논문을 공학박사 학위논문으로 제출함

2020 년 12 월

서울대학교 대학원

전기정보공학부

이 경 재

이경재의 공학박사 학위논문을 인준함

2020 년 12 월

위원장	최 진 영	(인)
부위원장	오 성 회	(인)
위 원	심 형 보	(인)
위 원	최 성 준	(인)
위 원	김 은 우	(인)

Abstract

The problem of sequential decision making under an uncertain and complex environment is a long-standing challenging problem in robotics. In this thesis, we focus on learning a policy function of robotic systems for sequential decision making under which is called a robot learning framework. In particular, we are interested in reducing the sample complexity of the robot learning framework. Hence, we develop three sample efficient robot learning frameworks. The first one is the maximum entropy reinforcement learning. The second one is a perturbation-based exploration. The last one is learning from demonstrations with mixed qualities.

For maximum entropy reinforcement learning, we employ a generalized Tsallis entropy regularization as an efficient exploration method. Tsallis entropy generalizes Shannon-Gibbs entropy by introducing a *entropic index*. By changing an entropic index, we can control the sparsity and multi-modality of policy. Based on this fact, we first propose a sparse Markov decision process (sparse MDP) which induces a sparse and multi-modal optimal policy distribution. In this MDP, the sparse entropy, which is a special case of Tsallis entropy, is employed as a policy regularization. We first analyze the optimality condition of a sparse MDP. Then, we propose dynamic programming methods for the sparse MDP and prove their convergence and optimality. We also show that the performance error of a sparse MDP has a constant bound, while the error of a soft MDP increases logarithmically with respect to the number of actions, where this performance error is caused by the introduced regularization term. Furthermore, we generalize sparse MDPs to a new class of entropy-regularized Markov decision processes (MDPs), which will be referred to as Tsallis MDPs, and analyzes different types of optimal policies with interesting properties related to the stochasticity of the optimal policy by controlling the entropic index.

Furthermore, we also develop perturbation based exploration methods to handle heavy-tailed noises. In many robot learning problems, a learning signal is often corrupted by noises such as sub-Gaussian noise or heavy-tailed noise. While most of the exploration strategies have been analyzed under sub-Gaussian noise assumption, there exist few methods for handling such heavy-tailed rewards. Hence, to overcome heavy-tailed noise, we consider stochastic multi-armed bandits with heavy-tailed rewards. First, we propose a novel robust estimator that does not require prior information about a noise distribution, while other existing robust estimators demand prior knowledge. Then, we show that an error probability of the proposed estimator decays exponentially fast. Using this estimator, we propose a perturbation-based exploration strategy and develop a generalized regret analysis scheme that provides upper and lower regret bounds by revealing the relationship between the regret and the cumulative density function of the perturbation. From the proposed analysis scheme, we obtain gap-dependent and gap-independent upper and lower regret bounds of various perturbations. We also find the optimal hyperparameters for each perturbation, which can achieve the minimax optimal regret bound with respect to total rounds.

For learning from demonstrations with mixed qualities, we develop a novel inverse reinforcement learning framework using leveraged Gaussian processes (LGP) which can handle negative demonstrations. In LGP, the correlation between two Gaussian processes is captured by a leveraged kernel function. By using properties, the proposed inverse reinforcement learning algorithm can learn from both positive and negative demonstrations. While most existing inverse reinforcement learning (IRL) methods suffer from the lack of information near low reward regions, the proposed method alleviates this issue by incorporating negative demonstrations. To mathematically formulate negative demonstrations, we

introduce a novel generative model which can generate both positive and negative demonstrations using a parameter, called *proficiency*. Moreover, since we represent a reward function using a leveraged Gaussian process which can model a nonlinear function, the proposed method can effectively estimate the structure of a nonlinear reward function.

Keywords: Multi-Armed Bandits, Online Learning, Reinforcement Learning, Inverse Reinforcement Learning, Learning from Demonstrations, Imitation Learning

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Conventional robotics have been developed under structured and controlled environments such as industrial factories without humans. Under such structured and unmanned environment, traditional robots have been manually programmed based on rules and have shown satisfactory performances for repetitive tasks. Furthermore, automation using industrial robots has improved efficiency and has increased safety of a risky task by replacing human labors. However, recently, robots are gradually permeating into our daily life to enhance our quality of life. Such robots are required to perform more complex operations, e.g., autonomous driving [2] and socially adaptive path planning [65]. In such cases, the traditional method of manually programming a robot based on rules has a clear limitation in that unexpected situations and complex tasks cannot be clearly represented by rules. In this regard, robot learning method, which can adapt unexpected situation and learn a new task autonomously, has been getting more attention. In particular, robot learning method has a great advantage over rule-based methods when it comes to learning a complex and unstructured task using less prior knowledge.

Robot learning is a technique for a robot to learn a optimal behavior from observations which have partial or full information about given tasks. Based on a type of given information, robot learning techniques can be categorized into two groups. The first group is called learning from rewards where a learning signal is given as a reward function. The second group is called learning from demonstrations where information about tasks is given as a demonstration of the expert.

Each group of methods has a clear benefit over rule-based methods. For example, learning from rewards can be applied to a personal service robot which requires high adaptability. Since every person has different preferences, it is hardly represented by rules. Then, learning from rewards is a nice approach for adapting a personal preference. For example of robotic grasping, conditions for successfully grasping an object depends on a friction and geometry of surface. However, it is difficult to define grasping rules depending on every type of surfaces. In this case, learning from rewards approach is required rather than a rule-based controller.

Learning from demonstrations also has a benefit over rule-based methods. Furthermore, demonstration-based approaches can cover the problem that cannot be solved by learning from rewards. In other words, learning from demonstrations has great advantages over both rule-based methods and learning from rewards approaches when it is difficult to model a reward function with multiple desiderata to be traded off [1]. Autonomous driving, for example, has to consider multiple criteria such as maintaining the center of a lane and avoiding collisions with other cars and pedestrians. While each of the criteria is easy to model, combining them into a single reward function is not straightforward. On the contrary, it is more natural to demonstrate driving behaviors and let the agent learn from demonstrations [1], which is also referred to as apprenticeship learning.

While robot learning is an important technique for future robotics industry, there still have lots of hurdles to apply this technique for the real-world problem. This dissertation focuses on improving sample efficiency Sample efficiency plays an important role for a success of robot learning. In learning from rewards, a reward-based robot learning find an optimal policy via trial and errors without the prior knowledge of the environment, such as the dynamics of environments and the structure of rewards. The absence of environmental information gives rise to an innate trade-off between exploration and exploitation during a learning process. If the algorithm decides to explore the environment, then, it will lose the chance to exploit the best decision based on collected experiences and vice versa. Such trade-off should be appropriately scheduled in order to learn an optimal policy through a small number of interactions with an environment. Especially, the efficiency of exploration becomes more important when training a robot since a hardware system of a robot can be damaged if the robot exceeds its durability. In this regard, improving sample efficiency is a main issue to be addressed in this dissertation.

In learning from demonstrations framework, optimality assumption for demonstrations makes data collecting process difficult. Learning from demonstrations aims to find the reward function which best explains demonstrations by experts. A key assumption is that experts follow the optimal policy induced by the underlying reward function. However, since demonstrations of experts are often distributed near high reward regions, the resulting robot behaviors cannot properly perform in low reward regions. For example, when learning how to drive, an autonomous vehicle occasionally encounters a risky situation, e.g. heading towards the side of the road. In order to avoid a catastrophic situation, the autonomous vehicle should recover back to the center of the road. However, such recovery behavior rarely appears in demonstrations from a good driver. In [109], Ross and Bargnell tackled this problem via continuous interaction with experts. However, it is not practical to rely on experts frequently. To handle lack of demonstrations near low reward regions, we incorporate demonstrations about both *what to do* and *what not to do*. As demonstrations about *what not to do* will be often distributed near low reward regions, we can obtain information to avoid catastrophic failures from such demonstrations. Demonstrations of failures had been considered before.

1.2 Organization of the Dissertation

This dissertation is organized as follows: In Chapter 2, backgrounds for robot learning methods are introduced. In particular, mathematical framework to formulate robot learning problems, such as multi-armed bandit, contextual bandit, or Markov decision processes, are mainly introduced. In Chapter 3 and 4, entropy based exploration method will be described. Especially, sparse entropy framework is proposed in Chapter 3 and it will be generalized to a unified framework using Tsallis entropy in Chapter 4. In Chapter 5, perturbation based exploration method will be described. in this chapter, we mainly address multi-armed bandit problem with stochastic noises and will introduce a novel exploration framework using random perturbation. Especially, we propose a . In Chapter 6, learning from negative demonstrations will be introduced. The framework which can handle a negative demonstrations is proposed and modeling data generating processes for negative demonstrations.

Chapter 2

Background

In this chapter, we introduce several frameworks for formulating a robot learning problem. In general, the main goal of a robot learning problem is to find a policy, also called a controller, to best perform a given task. Hence, a robot learning framework is often formulated as a sequential decision making problem for learning an optimal policy. The robot learning problem can be categorized into two groups based on which information is given: a reward signal or expert's demonstration. In reward-based learning framework, a reward signal provides a performance measure of a given task and the goal of a learning algorithm is to find the optimal policy maximizing the rewards (or cumulative rewards). On the contrary, in demonstration-based learning framework, expert demonstrations provide information about how to perform a given task. Hence, in this framework, the goal of a learning algorithm is to optimize a policy function to reproduce the expert's demonstrations.

2.1 Learning from Rewards

For reward-based learning, a robot learning problem can be formulated as a sequential decision making problem. In this framework, a robot chooses an action a_t consecutively for T rounds. T is called a horizon of the problem. Then, for each decision, a robot receives a reward signal R_t which is a real value indicating an instance performance of a given task for the decision a_t. Then, the goal of decision making is to maximize the expected reward $\mathbb{E}\left[R_t\right]$ (or the cumulative rewards). The basic assumption on this reward-based learning framework is the absence of prior knowledge about a reward signal. In other words, a robot does not know which action induces high rewards. A robot should verify an optimal action by consistently interacting with an environment while does not lose much rewards. Hence, a robot falls into a natural dilemma called exploration-exploitation trade-off. Many exploration strategies try to solve this dilemma by achieving the maximum rewards with the minimum trials.

The reward-based sequential decision making problems are categorized into three main frameworks based on how to represent an environment. Especially, representation of a reward function is a key difference. The first framework is an multi-armed bandit (MAB) which consists of rewards and actions. Hence, a reward only depends on an action. The second framework is a contextual bandit (CB) defined by adding a contextual space (or state space). In this framework, a reward depends on state and action. The final framework is the most general framework called a Markov decision process (MDP). An MDP consists of state and action space and a transition probability. Unlike MAB and CB, in an MDP, states are dependent through time. In other words, the next state depends on the current state where the Markov property makes the next state independent on the past states if the current state is conditioned. More detail definitions and notations are introduced in the following sections.

2.1.1 Multi-Armed Bandits

A stochastic multi-armed bandit problem is defined as a tuple $\{\mathcal{A}, r\}$ A learning agent plays T rounds. For each round $t \in \{1, \dots, T\}$, the agent takes an action (or arm) $a_t \in \mathcal{A}$ from an exploration strategy and obtains a stochastic reward

$$\mathbf{R}_{t,a_t} = r_{a_t} + \epsilon_t, \quad \mathbb{E}[\epsilon_t] = 0 \tag{2.1}$$

with deterministic mean function r_{a_t} and random noise ϵ_t. In this section, we assume $r_a \in [0, 1]$ which is widely used in bandit algorithms. The goal of the agent is to minimize the sum of pseudo regret over T rounds, which is defined as

$$\mathcal{R}_T := \sum_{t=1}^{T} r_{a^\star} - r_{a_t}, \quad a^\star := \operatorname*{argmax}_{a \in \mathcal{A}} r_a. \tag{2.2}$$

In our analysis, we often derive the upper bound of expectation of $\mathcal{R}_T$, which can be restated by,

$$\mathbb{E}\left[\mathcal{R}_T\right] = \sum_{a \in \mathcal{A}} \sum_{t=1}^{T} \Delta_a \mathbb{P}(a_t = a), \tag{2.3}$$

where $\Delta_a := r_{a^\star} - r_a$. $\mathcal{R}_T$ indicates how much rewards are lost during the exploration, hence, it means the efficiency of the exploration strategy.

2.1.2 Contextual Multi-Armed Bandits

A contextual bandit problem is defined by a tuple with three elements: $\{\mathcal{S}, \mathcal{A}, r\}$ where $\mathcal{S}$ is a context space, $\mathcal{A}$ is an action space, r is a reward which is a random variable indicating goodness of an action given a context. In this framework, the expected reward of pulling $a \in \mathcal{A}$ given $s \in \mathcal{S}$ is defined as a conditional expectation of the reward, $r_a(s) := \mathbb{E}\left[\mathbf{R}|s, a\right]$. Similarly to MAB problems, the goal of the contextual bandit problem is to find the best arm whose expected reward is the maximum by consecutively pulling arms and obtaining contexts and rewards every rounds.

An agent plays $1, \cdots, t, \cdots, T$. For each round, an arbitrary context s_t is given, then, a contextual bandit algorithm proposes a policy π_t based on s_t and sample an action a_t from π_t. The feedback of a_t is given as a reward $\mathbf{R}_t$. Since an expected reward $r_a(s_t)$ of each arm is unknown, rewards of each arm given s_t should be estimated. To estimate the expected rewards, $\hat{r}_a(s;\theta)$ is maintained where θ is the parameter of an estimator. $\hat{r}_a(s)$ is trained from the collected context and reward pairs. Generally, as the number of data increases, the error of reward estimations decreases. After estimators become accurate, the best arm can be selected based on $\hat{r}_a(s)$. Collecting more data to train $\hat{r}$ more accurately is called exploration and choosing the estimated best arm based on $\hat{r}$ is called exploitation. The main hurdle of bandit problem is balancing the exploration and exploration.

The efficiency of a bandit algorithm is often measured by the expected cumulative regret defined as

$$\mathcal{R}_T := \mathbb{E}_{s_{1:T}, a_{1:T}} \left[\sum_{t=1}^{T} \max_{a'} r_{a'}(s_t) - r_{a_t}(s_t) \right] \tag{2.4}$$

where $s_{1:T}$ indicates contexts given during T rounds and $a_{1:T}$ indicates actions selected during T rounds. If the algorithm focus on exploring arbitrary arms, $\mathcal{R}_T$ linearly increases. On the contrary, if the exploitation is focused, the estimation error of rewards is hardly reduced and $\mathcal{R}_T$ also linearly increases. When $\mathcal{R}_T$ sublinearly increases, such algorithms are called no regret and have the property that the error converges to zero as the number of rounds increases, i.e., $\lim_{T\to\infty} \frac{\mathcal{R}_T}{T} = 0$.

2.1.3 Markov Decision Processes

A Markov decision process (MDP) has been widely used to formulate a sequential decision making problem. An MDP can be characterized by a tuple $\mathbf{M} = \{\mathcal{S}, \mathcal{F}, \mathcal{A}, d, T, \gamma, r\}$, where $\mathcal{S}$ is the state space, $\mathcal{F}$ is the corresponding feature space, $\mathcal{A}$ is the action space, $d(s)$ is the distribution of an initial state, $T(s'|s,a)$ is the transition probability from $s \in \mathcal{S}$ to $s' \in \mathcal{S}$ by taking $a \in \mathcal{A}$, $\gamma \in (0,1)$ is a discount factor, and r is the reward function, i.e., $r(s,a,s') := \mathbb{E}\left[R|s,a,s'\right]$, and R is a noisy reward. The objective of an MDP is to find a policy which maximize $\mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t R_t \middle| \pi, d, T\right] = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t, s_{t+1}) \middle| \pi, d, T\right]$, where policy π is a mapping from the state space to the action space. For notational simplicity, we denote the expectation of a discounted summation of function $f(s,a)$, i.e., $\mathbb{E}[\sum_{t=0}^{\infty} \gamma^t f(s_t, a_t, s_{t+1})|\pi, d, T]$, by $\mathbb{E}_\pi[f(s,a,s')]$, where $f(s,a,s')$ is a function of state and action, such as a reward function $\mathbf{r}(s,a,s')$ or an indicator function $\mathbf{1}_{\{s_t = s\}}$. We also denote the expectation of a discounted summation of function $f(s,a,s')$ conditioned on the initial state, i.e., $\mathbb{E}[\sum_{t=0}^{\infty} \gamma^t f(s_t, a_t, s_{t+1})|\pi, s_0 = s, T]$, by $\mathbb{E}_\pi[f(s,a,s')|s_0 = s]$. Finding an optimal policy for an MDP can be formulated as follows:

$$
\begin{aligned}
\underset{\pi}{\text{maximize}} \quad & \mathbb{E}_\pi\left[\mathbf{r}(s_t, a_t, s_{t+1})\right] \\
\text{subject to} \quad & \forall\, s \ \sum_{a'} \pi(a'|s) = 1, \\
& \forall\, s, a \ \ \pi(a'|s) \geq 0.
\end{aligned}
\tag{2.5}
$$

The necessary condition for the optimal solution of (2.5) is called the Bellman equation. The Bellman equation is derived from the Bellman's optimality principal as follows:

$$
\begin{aligned}
Q_\pi(s,a) &= \sum_{s'} \left(\mathbf{r}(s,a,s') + \gamma V_\pi(s')\right) T(s'|s,a) \\
V_\pi(s) &= \max_{a'} Q(s,a') \\
\pi(s) &= \arg\max_{a'} Q(s,a'),
\end{aligned}
\tag{2.6}
$$

where $V_\pi(s)$ is a value function of π, which is the expected sum of discounted rewards when the initial state is given as s, and $Q_\pi(s,a)$ is a state-action value function of π, which is the expected sum of discounted rewards when the initial state and action are given as s and a, respectively. Note that the optimal solution is a deterministic function, which is referred to as a deterministic policy.

2.1.4 Soft Markov Decision Processes

An entropy-regularized MDP, also known as a soft MDP, has been widely used to represent a multi-modal policy function [134, 19, 115, 138]. In a soft MDP, causal entropy regularization over π is introduced to obtain a multi-modal policy distribution, i.e., $\pi(a|s)$. Since causal entropy regularization penalizes a deterministic distribution, it makes an optimal policy of a soft MDP to be a softmax distribution. A soft MDP is formulated as follows:

$$
\begin{aligned}
&\underset{\pi}{\text{maximize}} && \mathbb{E}_\pi\left[\mathbf{r}(s_t,a_t,s_{t+1})\right] + \alpha H(\pi) \\
&\text{subject to} && \forall\, s \ \sum_{a'} \pi(a'|s) = 1, \ \ \forall\, s,a \ \ \pi(a'|s) \geq 0,
\end{aligned}
\tag{2.7}
$$

where $H(\pi) \triangleq \mathbb{E}_\pi\left[-\log(\pi(a_t|s_t))\right]$ is a γ-discounted causal entropy and α is a regularization coefficient. This problem (A.2) has been extensively studied in [50, 19, 115]. In [19], a soft Bellman equation and the optimal policy distribution are derived from the Karush Kuhn Tucker (KKT) conditions as follows:

$$
\begin{aligned}
Q_{\pi}^{soft}(s,a) &= \sum_{s'} \left(r(s,a,s') + \gamma V_{\pi}^{soft}(s') \right) T(s'|s,a) \\
V_{\pi}^{soft}(s) &= \alpha \log \left(\sum_{a'} \exp \left(\frac{Q_{\pi}^{soft}(s,a')}{\alpha} \right) \right) \\
\pi(a|s) &= \frac{\exp\left(\frac{Q_{\pi}^{soft}(s,a)}{\alpha}\right)}{\sum_{a'} \exp\left(\frac{Q_{\pi}^{soft}(s,a')}{\alpha}\right)},
\end{aligned}
$$

where

$$
\begin{aligned}
V_{\pi}^{soft}(s) &= \mathbb{E}_{\pi}\left[r(s_t,a_t,s_{t+1}) - \alpha \log(\pi(a_t|s_t)) | s_0 = s \right] \\
Q_{\pi}^{soft}(s,a) &= \mathbb{E}_{\pi}\left[r(s_t,a_t,s_{t+1}) - \alpha \log(\pi(a_t|s_t)) | s_0 = s, a_0 = a \right].
\end{aligned}
$$

$V_{\pi}^{soft}(s)$ is a soft value of π indicating the expected sum of rewards including the entropy of a policy, obtained by starting at state s and $Q_{\pi}^{soft}(s,a)$ is a soft state-action value of π, which is the expected sum of rewards obtained by starting at state s by taking action a. Note that the optimal policy distribution is a softmax distribution. In [19], a soft value iteration method is also proposed and the optimality of soft value iteration is proved. By using causal entropy regularization, the optimal policy distribution of a soft MDP is able to represent a multi-modal distribution.

The causal entropy regularization has an effect of making the resulting policy of a soft MDP closer to a uniform distribution as the number of actions increases. To handle this issue, we propose a novel regularization method whose resulting policy distribution still has multiple modes (a stochastic policy) but the performance loss is less than a softmax policy distribution.

2.2 Learning from Demonstrations

For demonstration-based learning, there are two main streams: behavior cloning and inverse reinforcement learning. A behavior cloning learns a policy function (or distribution) of an expert from given demonstrations by using a supervised learning technique. An inverse reinforcement learning learns both reward function and policy of an expert from demonstrations.

Learning from demonstrations (LfD) is a technique of learning a skill from a set of expert's demonstrations which consist of state action pairs as follows,

$$\tau = \{s_0, a_0, \cdots, s_T, a_T, s_{T+1}\}$$

where T is a length of demonstrations. In LfD, it is assumed that a set of demonstrations are given ,i.e., $\mathcal{D} := \{\tau_i\}_{i=1}^{N}$ where N is the number of demonstrations. Furthermore, LfD also assumed that $\mathcal{D}$ is generated by an expert policy π_E which is an optimal policy for the underlying reward function r. The methods for LfD can be categorized into two groups: behavior cloning (BC) and inverse reinforcement learning (IRL). BC focuses on learning an optimal policy using a supervised learning method. On the contrary, an IRL method find not only optimal policy but also the underlying reward function of an experts. While IRL can lear both policy and rewards, the algorithm of IRL is often more complex and requires more computational resource than that of BC.

2.2.1 Behavior Cloning

Behavior cloning (BC) is a imitation learning method which focuses on finding an optimal policy from expert's demonstrations by using a supervised learning method. Since BC models an expert's policy using a function approximation, BC methods can be categorized into two groups based on an action space. First, for discrete action space, a classification method is generally used and the policy $\hat{\pi}$ is modeled as a classifier. Then, LfD problem can be formulated as the classification problem. The policy $\hat{\pi}$ is trained by minimizing a classification loss such as cross entropy loss,

$$\begin{aligned} &\underset{\phi}{\text{minimize}} && -\mathbb{E}_{s_t, a_t \in \mathcal{D}}\left[\ln\left(\pi_\phi(a_t|s_t)\right)\right] \\ &\text{subject to} && \forall\, s \ \ \sum_{a'} \pi_\phi(a'|s) = 1, \ \ \forall\, s, a \ \ \pi_\phi(a'|s) \geq 0, \end{aligned} \tag{2.8}$$

where ϕ is a parameter of classifier. Note that, since BC with discrete action space is formulated as a classification problem, other classification method also can be used. Second, for continuous action space, BC becomes a regression task. In this setting, BC is formulated as a regression problem, hence, the policy is optimized by minimizing the log-likelihood similarly to (2.8). However, the constraint is changed into the integral condition,

$$\forall\, s \ \ \int_{a \in \mathcal{A}} \pi_\phi(a|s)\mathbf{d}a = 1, \ \ \forall\, s, a \ \ \pi_\phi(a'|s) \geq 0.$$

The proposed methods in this dissertation is often focused on handling continuous action spaces.

2.2.2 Inverse Reinforcement Learning

Inverse reinforcement learning (IRL) problem aims to learn rewards function from expert's demonstrations by assuming that expert follows optimal policy induced by the underlying rewards function. The expert generates demonstrations $\mathcal{D} = \{\zeta_0, \cdots, \zeta_N\}$ where N is the number of demonstrations and ζ_i is a sequence of state and action pairs whose length is T, i.e., $\zeta_i = \{(s_0, a_0), \cdots, (s_T, a_T)\}$. Then, the goal of IRL is to recover both rewards $\mathbf{r}$ and corresponding optimal policy π from $\mathcal{D}$ when $\mathbf{M}/\mathbf{r}$ is given. However, IRL problem inherently contains ill-posedness due to the data inconsistency and ambiguity of rewards function.

Maximum Shannon-Gibbs Entropy Framework

Zeibart et al. [152] proposed the maximum causal entropy framework, which is also known as maximum entropy inverse reinforcement learning (MaxEnt IRL). MaxEnt IRL maximizes the causal entropy of a policy distribution while the feature expectation of the optimized policy distribution is matched with that of expert's policy. The maximum causal entropy framework is defined as follows:

$$\begin{aligned} \underset{\pi\in\Pi}{\text{maximize}} \quad & \alpha H(\pi) \\ \text{subject to} \quad & \mathbb{E}_{\pi}\left[\phi(s,a)\right] = \mathbb{E}_{\pi_E}\left[\phi(s,a)\right], \end{aligned} \tag{2.9}$$

where $H(\pi) \triangleq \mathbb{E}_{\pi}\left[-\log(\pi(a|s))\right]$ is the causal entropy of policy π, α is a scale parameter, π_E is the policy distribution of the expert, and ϕ is a feature mapping from a state action space to a feature space. Maximum causal entropy estimation finds the most uniformly distributed policy satisfying feature matching constraints. The feature expectation of the expert policy is used as a statistic to represent the behavior of an expert and is approximated from expert's demonstrations $\mathcal{D} = \{\zeta_0, \cdots, \zeta_N\}$, where N is the number of demonstrations and ζ_i is a sequence of state and action pairs whose length is T, i.e., $\zeta_i = \{(s_0, a_0), \cdots, (s_T, a_T)\}$. The principle of maximum causal entropy tells us that, if expert's demonstration is not enough to fully understand its underlying rewards, then, it is best to select a random action at the undemonstrated state in perspective of the worst case performance guarantee. In [151], it is shown that the optimal solution of (2.9) is a softmax distribution and the reward function is obtained as $\theta^{\intercal}\phi(s,a)$ where θ is the Lagrangian multiplier of the problem (2.9).

In [151], (2.9) was solved using Lagrangian method finding primal and dual variables alternatively. The dual problem of (2.9) is

$$\min_{\theta} \min_{\pi\in\Pi} \quad L_H(\theta, c, \lambda, \pi) \tag{2.10}$$

where L_H is a Lagrangian objective function deinfed as

$$\begin{aligned} L_H(\theta, c, \lambda, \pi) = &-\alpha H(\pi) - \mathbb{E}_\pi\left[\theta^\intercal \phi(s,a)\right] + \mathbb{E}_{\pi_E}\left[\theta^\intercal \phi(s,a)\right] \\ &+ \sum_s c_s \left(\sum_{a'} \pi(a'|s) - 1\right) - \sum_{s,a} \lambda_{sa} \pi(a|s), \end{aligned} \tag{2.11}$$

θ is a dual variable for feature matching constraints and has the same dimension with feature vector $\phi(s,a)$, $c \in \mathbb{R}^{|\mathcal{S}|}$ is a dual variable for sum to one constraints, and $\lambda \in \mathbb{R}^{|\mathcal{S}||\mathcal{A}|}$ is a dual variable for non-negative inequality. Note that the maximization is changed into the minimization with the negative objective function. From the Lagrangian objective function, [151] showed that $\underset{\pi}{\min} L_H(\theta, c, \lambda, \pi)$ is equivalent to solve the soft MDP under the rewards function $\theta^\intercal \phi(s,a)$ and the optimal policy is obtained as a softmax distribution, i.e., $\pi(a|s) = \frac{\exp(q(s,a))}{\sum_{a'} \exp(q(s,a'))}$ where $q(s,a) = \mathbb{E}\left[\sum_{t=0}^{\infty} \theta^\intercal \phi(s_t, a_t) | s_0 = s, a_0 = a, \pi\right]$.

Ho and Ermon further proposed Generative adversarial imitation learning (GAIL) by showing the equivalence between the unified IRL framework and generative adversarial networks (GANs) where the reward function and policy function in IRL correspond to discriminator and generator in GAN, respectively.

Generative Adversarial Framework For Model-Free Imitation Learning

In [55], Ho and Ermon have extended (2.9) to a unified framework for IRL by adding a reward regularization as follows:

$$\max_{c} \min_{\pi \in \Pi} \quad -\alpha H(\pi) + \mathbb{E}_\pi\left[c(s,a)\right] - \mathbb{E}_{\pi_E}\left[c(s,a)\right] - \psi(c), \tag{2.12}$$

where c is a cost function and ψ is a convex regularization for cost c. As shown in [55], many existing IRL methods can be interpreted with this framework, such as MaxEnt IRL [152], apprenticeship learning [1], and multiplicative weights apprenticeship learning [124]. Existing IRL methods based on (2.12) often require to solve the inner minimization over π for fixed c in order to compute the gradient of c. The inner minimization with cost c. In [151], Ziebart showed that the inner minimization is equivalent to a soft Markov decision process (soft MDP) under the reward $-c$ and proposed soft value iteration to solve the soft MDP. However, solving a soft MDP every iteration is often intractable for problems with large state and action spaces and also requires the transition probability which is not accessible in many cases. To address this issue, the generative adversarial imitation learning (GAIL) framework is proposed in [55] to avoid solving the soft MDP problem directly. It is proven that the minimization and maximization of (2.12) are interchangeable by the mini-max theorem [86]. Before explaining interchangability of (2.12), we introduce the settings about the rewards regularization and the feature space used in [152]. Ho and Ermon assumed that $\mathcal{F}$ is a set of $|\mathcal{S}||\mathcal{A}|$-dimensional unit vector $\mathbf{e}_{s'a'}$ whose element is determined as $[\mathbf{e}_{s'a'}]_{sa} = \mathbb{I}_{\{s'=s,a'=a\}}$ where $\mathbb{I}_{\{s'=s,a'=a\}}$ is an indicator function and ϕ is set to the function which maps a state action pair to the corresponding unit vector, i.e., $\phi(s,a) := \mathbf{e}_{sa}$, where the third input s' is ignored. Then, the feature expectation in (2.9) is equivalent to state-action visitation as follows:

$$\begin{aligned}\mathbb{E}_\pi\left[\left[\phi(s',a')\right]_{sa}\right] &= \mathbb{E}_\pi\left[\left[\mathbf{e}_{s'a'}\right]_{sa}\right] \\ &= \mathbb{E}_\pi\left[\mathbb{I}_{\{s'=s,a'=a\}}\right] = \rho_\pi(s,a).\end{aligned}$$

Consequently, the feature expectation constraints in (2.9) is converted into state-action visitation matching constraints as follows:

$$\forall\ s,a\ \ \rho_\pi(s,a) = \rho_{\pi_E}(s,a).$$

Then, the corresponding Lagrangian multiplier θ becomes a $|\mathcal{S}||\mathcal{A}|$-dimensional vector and is equivalent to the rewards function $\mathbf{r}$ since $\mathbf{r}(s,a) = \theta^\intercal\phi(s,a) = \theta^\intercal e_{sa} = \theta_{sa}$. Hence, in this problem, we only consider state action dependent rewards, i.e., $\mathbf{r}(s,a) = \mathbf{r}(s,a,s')$ for all s'. By using interchangability, unified framework can be converted as follows:

$$\underset{\pi}{\text{minimize}} \;\; \psi^*\left(\rho_\pi - \rho_{\pi_E}\right) - \alpha H(\pi)$$

where ψ^* is a conjugate function of the policy regularization, i.e., $\psi^*(x) = \max_{\mathbf{r}} -\psi(\mathbf{r}) + \sum_{s,a} \mathbf{r}(s,a)x_{sa}$ for any $x \in \mathbb{R}^{|\mathcal{S}||\mathcal{A}|}$. By using the interchangeability and using the specific rewards regularization,

$$\psi_{GA}(\mathbf{r}) \triangleq \begin{cases} \mathbb{E}_{\pi_E}\left[g(\mathbf{r}(s,a))\right], & \text{if } \mathbf{r} > 0 \\ \infty, & \text{otherwise} \end{cases}$$

where

$$g(x) \triangleq \begin{cases} x - \log(1 - \exp(-x)), & \text{if } x > 0 \\ \infty, & \text{otherwise,} \end{cases}$$

The unified imitation learning problem (2.12) can be converted into the GAIL framework as follows:

$$\min_{\pi \in \Pi} \max_{\mathbf{D}} \quad \mathbb{E}_\pi\left[\log(\mathbf{D}(s,a))\right] + \mathbb{E}_{\pi_E}\left[\log(1 - \mathbf{D}(s,a))\right] - \alpha H(\pi), \tag{2.13}$$

where $\mathbf{D} \in (0,1)^{|\mathcal{S}||\mathcal{A}|}$ indicates a discriminator, which returns the probability that a given demonstration is from a learner, i.e., 1 for learner's demonstrations and 0 for expert's demonstrations. Notice that we can interpret $\log(\mathbf{D})$ as cost c (or reward of $-c$).

Since existing IRL methods, including GAIL, are often based on the maximum causal entropy, they model the expert's policy using a softmax distribution, which can assign non-zero probability to non-expert actions in a discrete action space. Furthermore, in a continuous action space, expert's behavior is often modeled using a uni-modal Gaussian distribution, which is not proper to model multi-modal behaviors.

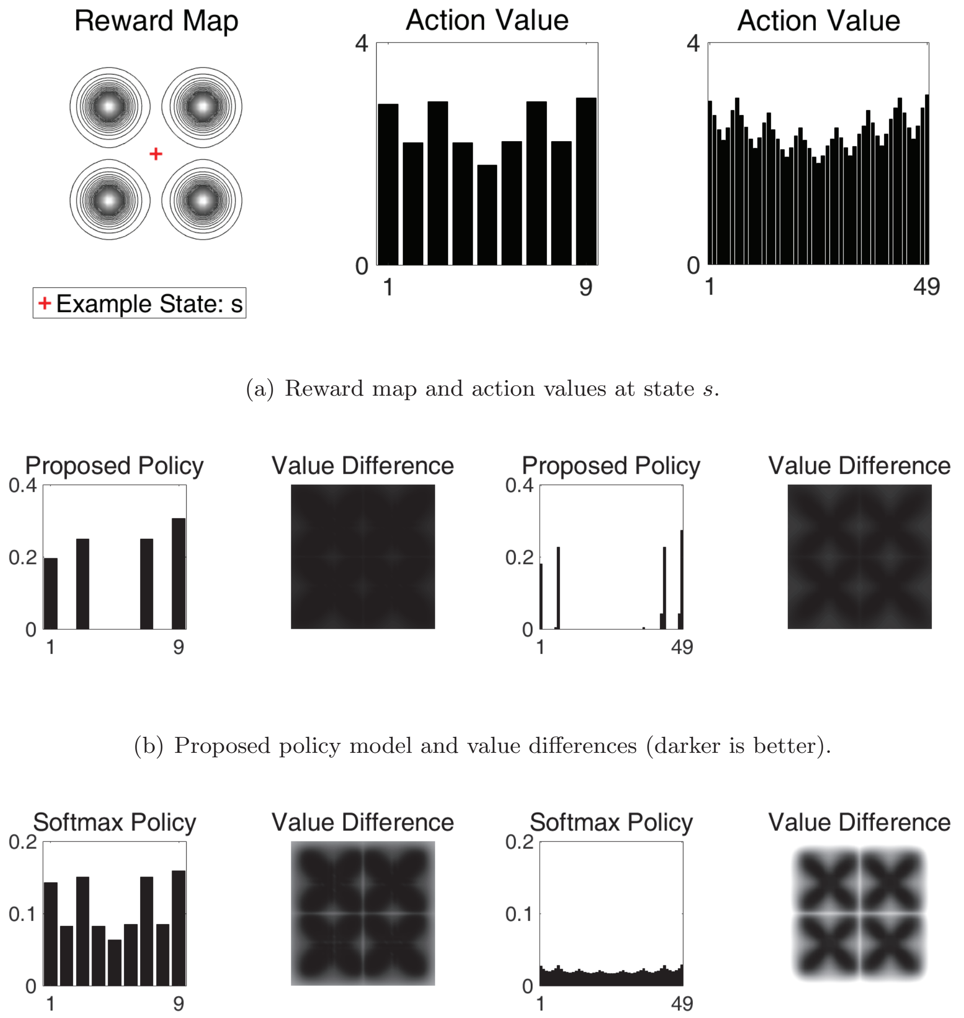
Chapter 3

Sparse Policy Learning

3.1 Sparse Policy Learning for Reinforcement Learning

Reinforcement learning (RL) has been widely used to solve stochastic sequential decision problems, such as autonomous driving [23], path planning [102], and quadrotor control [57]. In general, the goal of RL is to find the optimal policy function which maximizes the expected return. As we mentioned in Chapter 2.1, a Markov decision process (MDP) is often used to formulate reinforcement learning (RL) [68], which aims to find the optimal policy without the explicit specification of stochasticity of an environment, and inverse reinforcement learning (IRL) [92], whose goal is to search the proper reward function that can explain the behavior of an expert who follows the underlying optimal policy.

In Chapter 2.1, the optimal solution of an MDP is a deterministic policy. However, it is not desirable to apply an MDP to the problems with multiple optimal actions. In perspective of RL, the knowledge of multiple optimal actions makes it possible to cope with unexpected situations. For example, suppose that



(a) Reward map and action values at state s.

(b) Proposed policy model and value differences (darker is better).

(c) Softmax policy model and value differences (darker is better).

Figure 3.1: A 2-dimensional multi-objective environment with point mass dynamics.

an autonomous vehicle has multiple optimal routes to reach a given goal. If a traffic accident occurs at the currently selected optimal route, it is possible to avoid the accident by choosing another safe optimal route without additional computation of a new optimal route. For this reason, it is more desirable to learn all possible optimal actions in terms of robustness of a policy function. In perspective of IRL, since the experts often make multiple decisions in the same situation, a deterministic policy has a limitation in expressing the expert's behavior. For this reason, it is indispensable to model the policy function of an expert as a multi-modal distribution. These reasons give a rise to the necessity of a multi-modal policy model.

In order to address the issues with a deterministic policy function, a causal entropy regularization method has been utilized [50, 53, 115, 134, 138]. This is mainly due to the fact that the optimal solution of an MDP with causal entropy regularization becomes a softmax distribution of state-action values $Q(s, a)$, i.e., $\pi(a|s) = \frac{\exp(Q(s,a))}{\sum_{a'}\exp(Q(s,a'))}$, which is often referred to as a soft MDP [19]. While a softmax distribution has been widely used to model a stochastic policy, it has a weakness in modeling a policy function when the number of actions is large. In other words, the policy function modeled by a softmax distribution is prone to assign non-negligible probability mass to non-optimal actions even if state-action values of these actions are dismissible.

This tendency gets worse as the number of actions increases as demonstrated in Figure 3.1. In this example, the state is a location and the action is a velocity bounded with $[-3, 3] \times [-3, 3]$. Then, Figure 3.1(a) shows the reward map and action value functions for different discretization of action spaces. The left figure of Figure 3.1(a) shows the reward map with four maxima (multiple objectives). The action space is discretized into two levels: 9 (low resolution) and 49 (high resolution). The middle (resp., right) figure of Figure 3.1(a) shows the optimal action value at state s indicated as red cross point when the number of action is 9 (resp., 49). In Figure 3.1(b), the first and third figure indicate the proposed policy distributions at state s induced by the action values in Figure 3.1(a). The second and fourth figure of Figure 3.1(b) show a map of the performance difference between the proposed policy and the optimal policy at each state when the number of action is 9 and 49, respectively. The larger the error, the brighter the color of the state. Furthermore, all figures in Figure 3.1(c) are obtained in the same way as Figure 3.1(b) by replacing the proposed policy with a softmax policy. This example shows that the proposed policy model is less affected when the number of actions increases.

In this paper, we propose a sparse MDP by presenting a novel causal sparse Tsallis entropy regularization method, which can be interpreted as a special case of the Tsallis generalized entropy [135]. The proposed regularization method has a unique property in that the resulting policy distribution becomes a sparse distribution. In other words, the supporting action set which has a non-zero probability mass contains a sparse subset of the action space.

We provide a full mathematical analysis about the proposed sparse MDP. We first derive the optimality condition of a sparse MDP, which is named as a sparse Bellman equation. We show that the sparse Bellman equation is an approximation of the original Bellman equation. Interestingly, we further find the connection between the optimality condition of a sparse MDP and the probability simplex projection problem [140]. We present a sparse value iteration method for solving a sparse MDP problem, where the optimality and convergence are proven using the Banach fixed point theorem [120]. We further analyze the performance gaps of the expected return of the optimal policies obtained by a sparse MDP and a soft MDP compared to that of the original MDP. In particular, we prove that the performance gap between the proposed sparse MDP and the original MDP has a constant bound as the number of actions increases, whereas the performance gap between a soft MDP and the original MDP grows logarithmically. From this property, sparse MDPs have benefits over soft MDPs when it comes to solving problems in robotics with a continuous action space.

To validate effectiveness of a sparse MDP, we apply the proposed method to the exploration strategy and the update rule of Q-learning and compare to the ϵ-greedy method and softmax policy [134]. The proposed method is also compared to the deep deterministic policy gradient (DDPG) method [79], which is designed to operate in a continuous action space without discretization. The proposed method shows the state of the art performance compared to other methods as the discretization level of an action space increases.

3.1.1 Sparse Markov Decision Processes

We propose a sparse Markov decision process by introducing a novel causal sparse Tsallis entropy regularizer:

$$\begin{aligned} W(\pi) &:= \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t \frac{1}{2}(1-\pi(a_t|s_t)) \middle| \pi, d, T\right] \\ &= \mathbb{E}_{\pi}\left[\frac{1}{2}(1-\pi(a|s))\right]. \end{aligned} \tag{3.1}$$

By adding $W(\pi)$ to the objective function of (2.5), we aim to solve the following optimization problem:

$$\begin{aligned} &\underset{\pi}{\text{maximize}} && \mathbb{E}_{\pi}\left[\mathbf{r}(s,a,s')\right] + \alpha W(\pi) \\ &\text{subject to} && \forall\, s \ \sum_{a'} \pi(a'|s) = 1, \ \ \forall\, s, a \ \ \pi(a'|s) \geq 0, \end{aligned} \tag{3.2}$$

where $\alpha > 0$ is a regularization coefficient. We will first derive the sparse Bellman equation from the necessary condition of (3.2). Then by observing the connection between the sparse Bellman equation and the probability simplex projection, we show that the optimal policy becomes a *sparsemax distribution*, where the sparsity can be controlled by α. In addition, we present a sparse value iteration algorithm where the optimality is guaranteed using the Banach's fixed point theorem. The detailed derivations of lemmas and theorems in this paper can be found in the supplementary material.

Notations

Before explaining the proposed MDP, let us introduce some useful notations:

$$\begin{aligned}
J_{\pi}^{sp} &:= \mathbb{E}_{\pi}\left[r(s,a,s') + \frac{\alpha}{2}(1-\pi(a|s))\right] \\
V_{\pi}^{sp}(s) &:= \mathbb{E}_{\pi}\left[r(s,a,s') + \frac{\alpha}{2}(1-\pi(a|s))\Big|s_0 = s\right] \\
Q_{\pi}^{sp}(s,a) &:= \mathbb{E}_{\pi}\left[r(s,a,s') + \frac{\alpha}{2}(1-\pi(a|s))\Big|s_0 = s, a_0 = a\right] \\
r_{\pi}^{sp}(s) &:= \sum_{a}\left(r(s,a,s') + \frac{\alpha}{2}(1-\pi(a|s))\right)\pi(a|s),
\end{aligned}$$

where J_{π}^{sp} is the objective function of a sparse MDP, V_{π}^{sp} and Q_{π}^{sp} are a value function and an action value function of a sparse MDP, respectively, and r_{π}^{sp} is the expectation of rewards at state s. Here, the superscript sp indicates a sparse MDP problem.

Sparse Bellman Equation from Karush-Kuhn-Tucker conditions

The sparse Bellman equation can be derived from the necessary conditions of an optimal solution of a sparse MDP. We carefully investigate the Karush Kuhn Tucker (KKT) conditions, which indicate necessary conditions for a solution to be optimal when some regularity conditions about the feasible set are satisfied. The feasible set of a sparse MDP satisfies linearity constraint qualification [146] since the feasible set consists of linear affine functions. In this regards, the optimal solution of a sparse MDP necessarily satisfy KKT conditions as follows.

Theorem 1. *If a policy distribution π is the optimal solution of a sparse MDP (3.2), then π and the corresponding sparse value function V_{π}^{sp} necessarily satisfy the following equations for all state and action pairs:*

$$
\begin{aligned}
Q_{\pi}^{sp}(s,a) &= \sum_{s'} \left(\mathbf{r}(s,a,s') + \gamma V_{\pi}^{sp}(s')\right) T(s'|s,a) \\
V_{\pi}^{sp}(s) &= \alpha \left[\frac{1}{2}\sum_{a\in S(s)} \left(\left(\frac{Q_{\pi}^{sp}(s,a)}{\alpha}\right)^2 - \tau\left(\frac{Q_{\pi}^{sp}(s,\cdot)}{\alpha}\right)^2\right) + \frac{1}{2}\right] \\
\pi(a|s) &= \max\left(\frac{Q_{\pi}^{sp}(s,a)}{\alpha} - \tau\left(\frac{Q_{\pi}^{sp}(s,\cdot)}{\alpha}\right), 0\right),
\end{aligned}
\tag{3.3}
$$

where $\tau\left(\frac{Q_{\pi}^{sp}(s,\cdot)}{\alpha}\right) = \frac{\sum_{a\in S(s)} \frac{Q_{\pi}^{sp}(s,a)}{\alpha} - 1}{K_s}$, $S(s)$ is a set of actions satisfying $1 + i\frac{Q_{\pi}^{sp}(s,a_{(i)})}{\alpha} > \sum_{j=1}^{i} \frac{Q_{\pi}^{sp}(s,a_{(j)})}{\alpha}$ with $a_{(i)}$ indicating the action with the ith largest action value $Q_{\pi}^{sp}(s,a_{(i)})$, and K_s is the cardinality of $S(s)$.

The full proof of Theorem 1 is provided in the supplementary material. The proof depends on the KKT condition where the derivative of a Lagrangian objective function with respect to policy $\pi(a|s)$ becomes zero at the optimal solution, the stationary condition. From (3.3), it can be shown that the optimal solution obtained from the sparse MDP assigns zero probability to the action whose action value $Q^{sp}(s,a)$ is below the threshold $\tau\left(\frac{Q_{\pi}^{sp}(s,\cdot)}{\alpha}\right)$ and the optimal policy assigns positive probability to near optimal actions in proportion to their action values, where the threshold $\tau\left(\frac{Q_{\pi}^{sp}(s,\cdot)}{\alpha}\right)$ determines the range of near optimal actions. This property makes the optimal policy to have a sparse distribution and prevents the performance drop caused by assigning non-negligible positive probabilities to non-optimal actions, which often occurs in a soft MDP.

From the definitions of $S(s)$ and $\pi(a|s)$, we can further observe an interesting connection between the sparse Bellman equation and the probability simplex projection problem [140].

Probability Simplex Projection and SparseMax Operation

The probability simplex projection [140] is a well known problem of projecting a d-dimensional vector into a $d-1$ dimensional probability simplex in a Euclidean metric sense. A probability simplex projection problem is defined as follows:

$$\begin{aligned} \underset{p}{\text{minimize}} \quad & \frac{1}{2}||p-z||_2^2 \\ \text{subject to} \quad & \sum_{i=1}^{d} p_i = 1, \quad p_i \geq 0,\ \forall i = 1, \cdots, d, \end{aligned} \tag{3.4}$$

where z is a given d-dimensional vector, d is the dimension of p and z, and p_i is the ith element of p. Let $z_{(i)}$ be the ith largest element of z and $\text{supp}(z)$ be the supporting set of the optimal solution as defined by $\text{supp}(z) = \{z_{(i)} | 1 + iz_{(i)} > \sum_{j=1}^{i} z_{(j)}\}$. It is a well known fact that the problem (3.4) has a closed form solution which is $p_i^*(z) = \max(z_i - \tau(z), 0)$, where i indicates the ith dimension, $p_i^*(z)$ is the ith element of the optimal solution for fixed z, and $\tau(z) = \frac{\sum_{i=1}^{K} z_{(i)} - 1}{K}$ with $K = |\text{supp}(z)|$ [140, 85].

Interestingly, the optimal solution $p^*(\cdot)$, $\tau(\cdot)$ and the supporting set $\text{supp}(\cdot)$ of (3.4) can be precisely matched to those of the sparse Bellman equation (3.3). From this observation, it can be shown that the optimal policy distribution of a sparse MDP is the projection of $Q_\pi^{sp}(s, \cdot)$ into a probability simplex. Note that we refer to $p^*(\cdot)$ as a *sparsemax distribution* and denote it as $\text{spdist}(\cdot)$.

More surprisingly, V_π^{sp} can be represented as an approximation of the *max* operation derived from $p^*(z)$. A differentiable approximation of the *max* operation is defined as follows:

$$\text{spmax}(z) \triangleq \frac{1}{2}\sum_{i=1}^{K}\left(z_{(i)}^2 - \tau(z)^2\right) + \frac{1}{2} \tag{3.5}$$

We call $\text{spmax}(\cdot)$ as a *sparsemax operation.* In [85], it is proven that $\text{spmax}(z)$ is an indefinite integral of $p^*(z)$, i.e., $\text{spmax}(z) = \int (p^*(z))^\intercal \mathbf{d}z + C$, where C is a constant and, in our case, $C = \frac{1}{2}$. We provide simple upper and lower bounds of spmax(z) with respect to max(z)

$$\max(z) \leq \alpha \text{spmax}\left(\frac{z}{\alpha}\right) \leq \max(z) + \alpha\frac{d-1}{2d}. \tag{3.6}$$

The lower bound of spmax($\cdot$) is shown in [85]. However, we provide another proof of the lower bound and the proof for the upper bound in the supplementary material .

The bounds (3.6) show that spmax($\cdot$) is a bounded and smooth approximation of *max* and, from this fact, (3.3) can be interpreted as an approximation of the original Bellman equation. Using this notation, V_π^{sp} can be rewritten as,

$$V_\pi^{sp}(s) = \alpha \text{spmax}\left(\frac{Q_\pi^{sp}(s,\cdot)}{\alpha}\right). \tag{3.7}$$

Supporting Set of Sparse Optimal Policy

The supporting set $S(s)$ of a sparse MDP is a set of actions with nonzero probabilities and the cardinality of $S(s)$ can be controlled by regularization coefficient α, while the supporting set of a soft MDP is always the same as the entire action space. In a sparse MDP, actions assigned with non-zero probability must satisfy the following inequality:

$$\alpha + iQ_\pi^{sp}(s, a_{(i)}) > \sum_{j=1}^{i} Q_\pi^{sp}(s, a_{(j)}), \tag{3.8}$$

where $a_{(i)}$ indicates the action with the ith largest action value. From this inequality, it can be shown that α controls the margin between the largest action value and the others included in the supporting set. In other words, as α increases, the cardinality of a supporting set increases since the action values that satisfy (3.8) increase. Conversely, as α decreases, the supporting set decreases. In extreme cases, if α goes zero, only optimal actions will be included in $S(s)$ and if

α goes infinity, the entire actions will be included in $S(s)$. On the other hand, in a soft MDP, the supporting set of a softmax distribution cannot be controlled by the regularization coefficient α even if the sharpness of the softmax distribution can be adjusted. This property makes sparse MDPs have an advantage over soft MDPs, since we can give a zero probability to non-optimal actions by controlling α.

Connection to Tsallis Generalized Entropy

The notion of the Tsallis entropy was introduced by C. Tsallis as a general extension of entropy [135] and the Tsallis entropy has been widely used to describe thermodynamic systems and molecular motions. Surprisingly, the proposed regularization is closely related to a special case of the Tsallis entropy. The Tsallis entropy is defined as follows:

$$S_q(p) = \frac{1}{q-1}\left(1 - \sum_i p_i^q\right),$$

where p is a probability mass function and q is a parameter called *entropic-index*. Note that, if $q \to 1$, $S_1(p)$ is the same as entropy, i.e., $-\sum_i p_i \log(p_i)$. In [151, 19], it is shown that $H(\pi)$ is an extension of $S_1(\pi(\cdot|s))$ since $H(\pi) = \mathbb{E}_\pi[S_1(\pi(\cdot|s))] = \sum_{s,a} -\pi(a|s)\log(\pi(a|s))\rho(s)$ where $\rho(s)$ is the state visitation of s defined as $\mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t \mathbb{I}(s_t = s)\right]$.

We discover the connection between the Tsallis entropy and the proposed regularization when $q = 2$.

Theorem 2. *The proposed policy regularization $W(\pi)$ is an extension of the Tsallis entropy with parameters $q = 2$ to the version of causal entropy, i.e.,*

$$W(\pi) = \frac{1}{2}\mathbb{E}_\pi[S_2(\pi(\cdot|s))].$$

From this theorem, $W(\pi)$ can be interpreted as an extension of $\frac{1}{2}S_2(p)$ to the case of a causally conditioned distribution, similarily to the causal entropy.

3.1.2 Sparse Value Iteration

In this section, we propose an algorithm for solving a causal sparse Tsallis entropy regularized MDP problem. Similar to the original MDP and a soft MDP, the sparse version of value iteration can be induced from the sparse Bellman equation. We first define a sparse Bellman operation $U^{sp} : \mathbb{R}^{|\mathcal{S}|} \rightarrow \mathbb{R}^{|\mathcal{S}|}$ as follows: for all s,

$$U^{sp}(x)(s) := \alpha \text{spmax}\left(\frac{\sum_{s'}\left(r(s,\cdot,s') + \gamma x(s')\right)T(s'|s,\cdot)}{\alpha}\right),$$

where x is a vector in $\mathbb{R}^{|\mathcal{S}|}$ and $U^{sp}(x)$ is the resulting vector after applying U^{sp} to x and $U^{sp}(x)(s)$ is the element for state s in $U^{sp}(x)$. Then, the sparse value iteration algorithm can be described simply as

$$x_{i+1} = U^{sp}(x_i),$$

where i is the number of iterations. In the following section, we show the convergence and the optimality of the proposed sparse value iteration method.

Optimality of Sparse Value Iteration

In this section, we prove the convergence and optimality of the sparse value iteration method. We first show that U^{sp} has monotonic and discounting properties and, by using those properties, we prove that U^{sp} is a contraction. Then, by the Banach fixed point theorem, with repeated applications of U^{sp}, it always converges to a unique fixed point from an arbitrary initial point.

Lemma 1. *U^{sp} is monotone: for $x, y \in \mathbb{R}^{|\mathcal{S}|}$, if $x \leq y$, then $U^{sp}(x) \leq U^{sp}(y)$, where $\leq$ indicates an element-wise inequality.*

Lemma 2. *For any constant $c \in \mathbb{R}$, $U^{sp}(x + c\mathbf{1}) = U^{sp}(x) + \gamma c\mathbf{1}$, where $\mathbf{1} \in \mathbb{R}^{|\mathcal{S}|}$ is a vector of all ones.*

The full proofs can be found in the supplementary material. The proofs of Lemma 1 and Lemma 2 rely on the bounded property of the sparsemax operation. It is possible to prove that the sparse Bellman operator U^{sp} is a contraction using Lemma 1 and Lemma 2 as follows:

Lemma 3. *U^{sp} is a γ-contraction mapping and have a unique fixed point, where γ is in $(0, 1)$ by definition.*

Using Lemma 1, Lemma 2, and Lemma 3, the optimality and convergence of sparse value iteration can be proven.

Theorem 3. *Sparse value iteration converges to the optimal value of (3.2).*

The proof can be found in the supplementary material. Theorem 3 is proven using the uniqueness of the fixed point of U^{sp} and the sparse Bellman equation.

3.1.3 Performance Error Bounds for Sparse Value Iteration

We prove the bounds of the performance gap between the policy obtained by a sparse MDP and the policy obtained by the original MDP, where this performance error is caused by regularization. The boundedness of (3.6) plays a crucial role to prove the error bounds. The performance bounds can be derived from bounds of *sparsemax*. A similar approach can be applied to prove the error bounds of a soft MDP since a log-sum-exp function is also a bounded approximation of the *max* operation.

Before explaining the performance error bounds, we introduce two useful propositions which are employed to prove the performance error bounds of a sparse MDP and a soft MDP. We first prove an important fact which shows that the optimal values of sparse value iteration and soft value iteration are greater than that of the original MDP.

Lemma 4. *Let U and U^{soft} be the Bellman operations of an original MDP and soft MDP, respectively, such that, for state s and $x \in \mathbb{R}^{|\mathcal{S}|}$,*

$$\begin{aligned} U(x)(s) &= \max_{a'} \sum_{s'} \left(r(s,a',s') + \gamma x(s') T(s'|s,a') \right) \\ U^{soft}(x)(s) &= \alpha \log \sum_{a'} \exp\left(\frac{\sum_{s'} \left(r(s,a',s') + \gamma x(s') \right) T(s'|s,a')}{\alpha} \right). \end{aligned}$$

Then following inequalities hold for every integer $n > 0$:

$$U^n(x) \leq (U^{sp})^n(x), \;\; U^n(x) \leq (U^{soft})^n(x),$$

where U^n (resp., $(U^{sp})^n$) is the result after applying U (resp., U^{sp}) n times. In addition, let x_*, x_*^{sp} and x_*^{soft} be the fixed points of U, U^{sp} and U^{soft}, respectively. Then, following inequalities also hold:

$$x_* \leq x_*^{sp}, \;\; x_* \leq x_*^{soft}.$$

Lemma 4 shows that the optimal values, V_π^{sp} and V_π^{soft}, obtained by sparse value iteration and soft value iteration are always greater than the original optimal value V_π. Intuitively speaking, the reason for this inequality is due to the regularization term, i.e., $W(\pi)$ or $H(\pi)$, added to the objective function.

Now, we discuss other useful properties about the proposed causal sparse Tsallis entropy regularization $W(\pi)$ and causal entropy regularization $H(\pi)$.

Lemma 5. *$W(\pi)$ and $H(\pi)$ have following upper bounds:*

$$W(\pi) \leq \frac{1}{1-\gamma} \frac{|\mathcal{A}|-1}{2|\mathcal{A}|}, \;\; H(\pi) \leq \frac{\log(|\mathcal{A}|)}{1-\gamma}$$

where $|\mathcal{A}|$ is the cardinality of the action space $\mathcal{A}$.

Using Lemma 4 and Lemma 5, the performance bounds for a sparse MDP and a soft MDP can be derived as follows.

Theorem 4. *Following inequalities hold:*

$$\mathbb{E}_{\pi^*}\left[\mathbf{r}(s,a,s')\right] - \frac{\alpha}{1-\gamma}\frac{|\mathcal{A}|-1}{2|\mathcal{A}|} \leq \mathbb{E}_{\pi^{sp}}\left[\mathbf{r}(s,a,s')\right] \leq \mathbb{E}_{\pi^*}\left[\mathbf{r}(s,a,s')\right],$$

where π^* and π^{sp} are the optimal policy obtained by the original MDP and a sparse MDP, respectively.

Theorem 5. *Following inequalities hold:*

$$\mathbb{E}_{\pi^*}\left[\mathbf{r}(s,a,s')\right] - \frac{\alpha}{1-\gamma}\log(|\mathcal{A}|) \leq \mathbb{E}_{\pi^{soft}}\left[\mathbf{r}(s,a,s')\right] \leq \mathbb{E}_{\pi^*}\left[\mathbf{r}(s,a,s')\right]$$

where π^* and π^{soft} are the optimal policy obtained by the original MDP and a soft MDP, respectively.

These error bounds show us that the expected return of the optimal policy of a sparse MDP has always tighter error bounds than that of a soft MDP. Moreover, it can be also known that the bounds for the proposed sparse MDP converges to a constant $\frac{\alpha}{2(1-\gamma)}$ as the number of actions increases, whereas the error bounds of soft MDP grows logarithmically.

This property has a clear benefit when a sparse MDP is applied to a robotic problem with a continuous action space. To apply an MDP to a continuous action space, a discretization of the action space is essential and a fine discretization is required to obtain a solution which is closer to the underlying continuous optimal policy. Accordingly, the number of actions becomes larger as the level of discretization increases. In this case, a sparse MDP has advantages over a soft MDP in that the performance error of a sparse MDP is bounded by a constant factor as the number of actions increases, whereas performance error of optimal policy of a soft MDP grows logarithmically.

3.1.4 Sparse Exploration and Update Rule for Sparse Deep Q-Learning

In this section, we first propose sparse Q-learning and further extend to sparse deep Q-learning, where a sparsemax policy and the sparse Bellman equation are employed as a exploration method and update rule. Sparse Q-learning is a model free method to solve the proposed sparse MDP without the knowledge of transition probabilities. In other words, when the transition probability $T(s'|a, s)$ is unknown but sampling from $T(s'|a, s)$ is possible, sparse Q-learning estimates an optimal Q^{sp} of the sparse MDP using sampling, as Q-learning finds an approximated value of an optimal Q of the conventional MDP. Similar to Q-learning, the update equation of sparse Q-learning is derived from the sparse Bellman equation,

$$\begin{aligned}&Q^{sp}(s_i, a_i) \leftarrow Q^{sp}(s_i, a_i) + \\ &\eta(i)\left[\mathbf{r}(s_i, a_i, s_{i+1}) + \gamma\alpha\text{spmax}\left(\frac{Q^{sp}(s_{i+1}, \cdot)}{\alpha}\right) - Q^{sp}(s_i, a_i)\right],\end{aligned}$$

where i indicates the number of iterations and $\eta(i)$ is a learning rate. If the learning rate $\eta(i)$ satisfies $\sum_{i=0}^{\infty} \eta(i) = \infty$ and $\sum_{i=0}^{\infty} \eta(i)^2 < \infty$, then, as the number of samples increases to infinity, sparse Q-learning converges to the optimal solution of a sparse MDP. The proof of the convergence and optimality of sparse Q-learning is the same as that of the standard Q-learning [142].

The proposed sparse Q-learning can be easily extended to sparse deep Q-learning using a deep neural network as an estimator of the sparse Q value. In each iteration, sparse deep Q-learning performs a gradient descent step to minimize the squared loss $(y - Q(s, a; \theta))^2$, where θ is the parameter of the Q network. Here, y is the target value defined as follows:

$$y = \mathbf{r}(s, a, s') + \gamma\alpha\text{spmax}\left(\frac{Q(s', \cdot; \theta)}{\alpha}\right),$$

where s' is the next state sampled by taking action a at the state s and θ are network parameters.

Moreover, we employ the sparsemax distribution as a exploration strategy. To guarantee that all state and action pairs are sufficiently visited, we propose sparsemax exploration by combining a sparsemax distribution and the ϵ-greedy method as follows:

$$\pi_t(\cdot|s) = (1 - \epsilon_t)\text{spdist}\left(\frac{Q(s, \cdot; \theta)}{\alpha}\right) + \frac{\epsilon_t}{|\mathcal{A}|}\mathbf{1}, \tag{3.9}$$

where π_t indicates the exploration policy at iteration t. $\epsilon_t \in (0, 1]$ is the proportion of ϵ-greedy, which can be scheduled similar to the learning rate. When ϵ_t is close to one, sparsemax exploration acts like a uniform distribution, similarly to ϵ-greedy. After ϵ_t is reduced to a sufficiently small number, sparsemax exploration converges to a sparemax distribution. Hence, sparsemax exploration selectively re-explore the meaningful actions when ϵ_t is small. The effectiveness of the sparsemax exploration is investigated in Section 3.1.5.

For stable convergence of a Q network, we utilize double Q-learning [139], which prevents instability of deep Q-learning by slowly updating the target value. Prioritized experience replay [112] is also applied using weighted loss function for training Q network. The whole process of sparse deep Q-learning is summarized in Algorithm 1.

3.1.5 Experiments

We first verify Theorem 4, Theorem 5 and the effect of (3.8) in simulation. For the verification of Theorem 4 and Theorem 5, we measure the performance of the expected return while increasing the number of actions, $|\mathcal{A}|$. For the verification of the effect of (3.8), the cardinality of the supporting set of optimal policies of sparse and soft MDP are compared at different values of α.

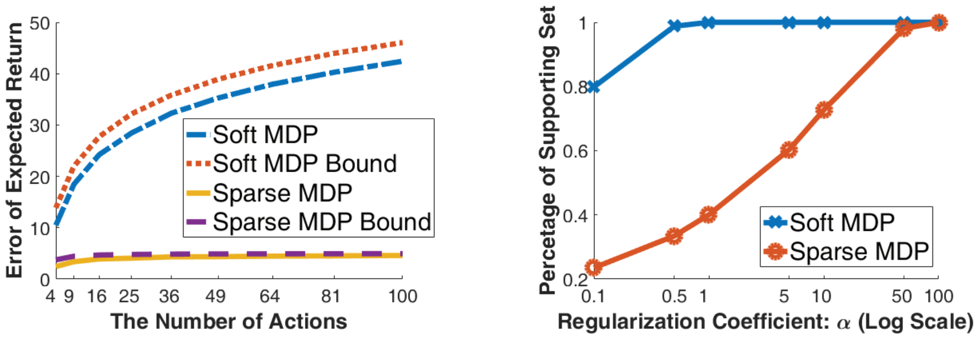
Algorithm 1 Sparse Deep Q-Learning

```
Initialize prioritized replay memory $M = \emptyset$, Q network parameters $\theta$ and $\theta^{-}$, $\epsilon_0 = 1$
for $i = 0$ to $N$ do
    Sample initial state $s_0 \sim d_0(s)$
    for $t = 0$ to $T$ do
        Sample action $a_t \sim \pi_t(a|s_t)$ (3.9)
        Execute $a_t$ and observe next state $s_{t+1}$ and reward $\mathbf{r}_t$
        Add experiences to replay memory $M$ with an initial importance weight, $M \leftarrow (s_t, a_t, \mathbf{r}_t, s_{t+1}, w_0) \cup M$
        Sample mini-batch $B$ from $M$ with importance weight
        Set a target value $y_j$ of $(s_j, a_j, \mathbf{r}_j, s_{j+1}, w_j)$ in $B$, $y_j = \mathbf{r}_j + \gamma\alpha\mathrm{spmax}\left(\frac{Q(s_{j+1},\cdot;\theta^{-})}{\alpha}\right)$
        Minimize $\sum_j w_j \left(y_j - Q(s_j, a_j; \theta)\right)^2$ using a gradient descent method
        Update $\epsilon_t$ and importance weights $\{w_j\}$ based on temporal difference error $\delta_j = |y_j - Q(s_j, a_j; \theta)|$ [112]
    end for
    Update $\theta^{-} = \theta$ every $c$ iteration
end for
```

To investigate effectiveness of the proposed method, we test sparsemax exploration and the sparse Bellman update rule on reinforcement learning with a continuous action space. To apply Q-learning to a continuous action space, a fine discretization is necessary to obtain a solution which is closer to the original continuous optimal policy. As the level of discretization increases, the number of actions to be explored becomes larger. In this regards, an efficient exploration method is required to obtain high performance. We compare our method to other exploration methods with respect to the convergence speed and the expected sum of rewards. We further check the effect of the update rule.

Experiments on Performance Bounds and Supporting Set

To verify our theorem about performance error bounds, we create a transition model T by discretization of unicycle dynamics defined in a continuous state and action space and solve the original MDP, a soft MDP and a sparse MDP under predefined rewards while increasing the discretization level of the action space. The reward function is defined as a linear combination of two squared exponential functions, i.e., $\mathbf{r}(x) = \exp\left(\frac{||x-x_1||^2}{2\sigma_1^2}\right) - \exp\left(\frac{||x-x_2||^2}{2\sigma_2^2}\right)$, where x is a location of a unicycle, x_1 is a goal point, x_2 is the point to avoid, and σ_1 and σ_2 are scale parameters. The reward function is designed to let an agent to navigate towards x_1 while avoiding x_2. The absolute value of differences between the expected return of the original MDP and that of sparse MDP (or soft MDP) is measured. As shown in Figure 3.2(a), the performance gap of sparse MDP converges to a constant bound while the performance of the soft MDP grows logarithmically. Note that the performance gaps of the sparse MDP and soft MDP are always smaller than their error bounds. Supporting set experiments are conducted using discretized unicycle dynamics. The cardinality of optimal policies are measured while α varies from 0.1 to 100. In Figure 3.2(b), while the ratio of the supporting set for a soft MDP is changed from 0.79 to 1.00, the ratio for a sparse MDP is changed from 0.24 to 0.99, demonstrating the sparseness of the proposed sparse MDPs compared to soft MDPs.



(a) Performance Bounds

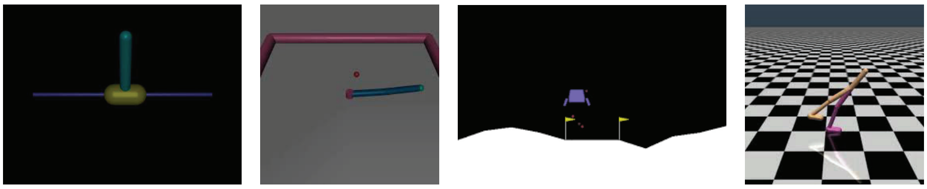
(b) Supporting Set Comparison

Figure 3.2: (a) The performance gap is calculated as the absolute value of the difference between the performance of sparse MDP or soft MDP and the performance of an original MDP. (b) The ratio of the number of supporting actions to the total number of actions is shown. The action space of unicycle dynamics is discretized into 25 actions.

Reinforcement Learning in a Continuous Action Space

We test our method in OpenAI Gym and MuJoCo [133], a physics-based simulator, using four problems with a continuous action space: *inverted pendulum, reacher, lunar lander*, and *Walker2D*. The action space is discretized to apply Q-learning to a continuous action space and experiments are conducted with fine discretization to validate the effectiveness of sparsemax exploration and the sparse Bellman update rule.

We compare the sparsemax exploration method to the ϵ-greedy and softmax exploration [138] and further compare the sparse Bellman update rule to the original Bellman [142] and the soft Bellman [19] update rule. In total, we test 9 combinations of variants of deep Q-learning by combining three exploration methods and three update rules. The deep deterministic policy gradient (DDPG)



(a) Inverted Pendulum (b) Reacher (c) Lunar Lander (d) Walker2D

Figure 3.3: (a) An inverted pendulum is mounted on a cart. The cart can only move horizontally. The goal of this task is to control a cart to keep the pendulum upright. (b) Two joint arm is fixed at the center point. The goal of this task is to control the end effector of arm to reach the goal point. (c) The point mass is dropped by gravity. The goal of this task is to land safely inside the landing pad (between two flags) and use less fuel. (d) 7 joint bipedal robot should move forward as quickly as possible.

method [79], which operates in a continuous action space without discretization of the action space, is also compared[1]. Hence, a total of 10 algorithms are tested. The experiments are repeated with five different random seeds. We find the best α and ϵ value for each algorithm using a brute force search. A Q network with two 512 dimensional hidden layers is used for inverted pendulum and Walker2D problems and a Q network with four 256 dimensional hidden layers is used for reacher and lunar lander problems. Each Q-learning algorithm utilizes the same network topology for the same problem. More details about the problems and experiment settings can be found in the supplementary material .

Results are shown in Table 3.1 and 3.2, where the maximum average return and the number of episodes to reach a given threshold return value are shown. For the inverted pendulum problem, every algorithm achieves the maximum av-

[1] To test DDPG, we used the code from Open AI available at `https://github.com/openai/baselines`.

Task	$\lvert\mathcal{A}\rvert$	Sps+SpsB	Sps+SftB	Sps+B	Stf+SpsB	Stf+StfB	Stf+B	Eps+SpsB	Eps+StfB	Eps+B	DDPG
Inv. Pend.	2001^1	1000.0	1000.0	1000.0	1000.0	1000.0	1000.0	1000.0	1000.0	1000.0	1000.0
Reacher	51^2	**-4.9**	-5.5	-5.0	-5.5	-5.6	-5.5	-5.6	-5.6	-5.5	-5.9
Lun. Lan. Cont.	51^2	216.5	**223.4**	215.5	214.7	211.2	212.9	-324.7	-337.1	-349.5	216.5
Walker2D	3^6	1218.9	1189.8	**1853.6**	1625.8	1222.4	1269.5	1416.7	1244.7	690.5	1312.2

Table 3.1: Maximum average return with consecutive 100 episodes for five different random seeds. Sps, Sft, and Eps stand for Sparsemax, Softmax, and ϵ-greedy exploration, respectively, and SpsB, SftB, B stand for the sparse, soft, and standard Bellman update rule, respectively. Algorithms are named as `<exploration method>+<update rule>`. (The best result is shown in bold.)

erage return of 1000. However, algorithms with sparsemax exploration reach the threshold value, 980, slightly faster than softmax exploration. For the reacher problem, algorithms with sparsemax exploration outperform other exploration methods with respect to the maximum average expected return while softmax exploration converges faster than other methods. For the lunar lander, sparsemax exploration shows the best expected return and the smallest number of episodes needed to reach the threshold value. For the Walker2D problem, the method combining sparsemax exploration and Bellman update rule shows the best expected return value while its convergence speed is the second best. DDPG shows a slower convergence speed than sparsemax and softmax exploration since training actor and critic networks requires more episodes. The experimental results show that the sparsemax exploration method has an advantage over softmax exploration, ϵ-greedy method and DDPG with respect to the number of episodes to reach the optimal performance.

Task	Threshold	Sps+SpsB	Sps+SftB	Sps+B	Stf+SpsB	Stf+StfB	Stf+B	Eps+SpsB	Eps+StfB	Eps+B	DDPG
Inv. Pend.	980	673	**573**	957	583	835	700	1693	1717	1488	1009
Reacher	-7.0	1155	1205	1256	1363	**1064**	2636	2502	2588	2298	2298
Lun. Lan. Cont.	170.0	574	**397**	480	494	753	529	-	-	-	2213
Walker2D	1000.0	1341	1447	1194	**1119**	1403	1429	1333	1440	-	1388

Table 3.2: The number of episodes required to cross a given threshold of the average expected return. '-' indicates that the algorithm has not reached the given threshold within the given training episode from all runs. The performance is measured after exploring 4000, 10000, 3000, and 1500 episodes for inverted pendulum, reacher, lunar lander, and Walker2D, respectively. (The best result is shown in bold.)

Task	Sps+SpsB	Sps+SftB	Sps+B	Stf+SpsB	Stf+StfB	Stf+B	Eps+SpsB	Eps+StfB	Eps+B
Inv. Pend.	$\alpha : 0.1$	$\alpha : 0.01$	$\alpha : 0.01$	$\alpha : 0.01$	$\alpha : 0.1$	$\alpha : 0.1$	$\alpha : 0.1$	$\alpha : 1$	-
	$\epsilon : 0.995$	$\epsilon : 0.995$	$\epsilon : 0.995$	-	-	-	$\epsilon : 0.995$	$\epsilon : 0.995$	$\epsilon : 0.995$
Reacher	$\alpha : 0.1$	$\alpha : 0.1$	$\alpha : 0.1$	$\alpha : 0.01$	$\alpha : 0.01$	$\alpha : 0.01$	$\alpha : 0.01$	$\alpha : 1$	-
	$\epsilon : 0.99$	$\epsilon : 0.99$	$\epsilon : 0.99$	-	-	-	$\epsilon : 0.995$	$\epsilon : 0.995$	$\epsilon : 0.995$
Lun. Lan.	$\alpha : 0.1$	$\alpha : 1$	$\alpha : 1$	$\alpha : 0.1$	$\alpha : 0.1$	$\alpha : 0.1$	$\alpha : 1$	$\alpha : 0.1$	-
	$\epsilon : 0.999$	$\epsilon : 0.999$	$\epsilon : 0.999$	-	-	-	$\epsilon : 0.99$	$\epsilon : 0.99$	$\epsilon : 0.99$
Walker2D	$\alpha : 0.01$	$\alpha : 0.1$	$\alpha : 0.5$	$\alpha : 0.1$	$\alpha : 0.1$	$\alpha : 0.1$	$\alpha : 1.0$	$\alpha : 1.0$	-
	$\epsilon : 0.9995$	$\epsilon : 0.9995$	$\epsilon : 0.9995$	-	-	-	$\epsilon : 0.9995$	$\epsilon : 0.9995$	$\epsilon : 0.9995$

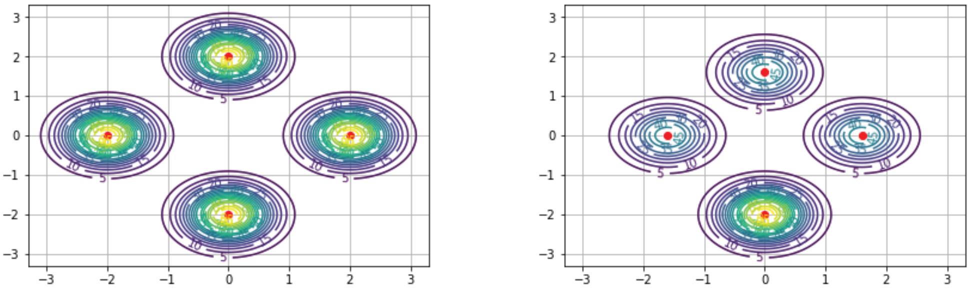
Table 3.3: The best performed parameters for each algorithm in each problem.

Task	Deep Q learning	DDPG
Inv. Pend. Walker2D	Two 512 ReLU layers	Two 128 ReLU layers
Reacher Lun. Lan. Cont.	Four 256 ReLU layers	Two 64 ReLU layers

Table 3.4: Network Structures. ReLU indicates a rectified linear unit.

Multi-Objective Exploration

In order to verify that sparsemax exploration can successfully learn multi-modal optimal actions, we designed a simple multi-objective environment where an agent

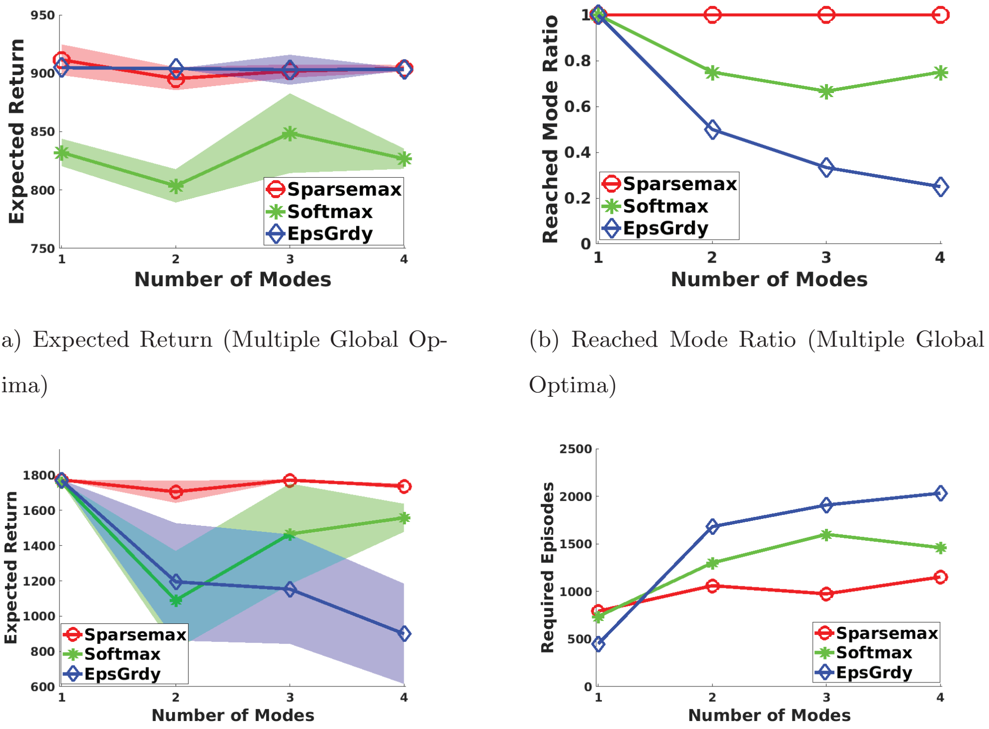


(a) Multiple Global Optima

(b) Multiple Local Optima

Figure 3.4: (a) Reward maps with multiple global optima and (b) multiple local optima. Red dot indicates goal points. The agent always starts at the center.

follows point mass dynamics and tries to reach one of equally distributed multiple modes. The reward function is defined as a mixture of squared exponential functions whose centers are placed at the goal positions (see Figure 3.4(a)). If the exploration method successfully explores the environment, then the resulting policy distribution will equally reach the multiple modes. We compare the sparsemax, softmax, and ϵ-greedy methods and measure the average return and ratio of reached modes to given modes while changing the number of global optima with five different random seeds. The values of α and ϵ are found by a brute force search. α is set to 3 and 5 for softmax and sparsemax policy, respectively, and the decay rate of ϵ is set to 0.9995. The results are shown in Figure 3.5. The average performance and the reached mode ratio are shown in Figure 3.5(a). The resulting sparsemax policy can reach every modes while maintaining its performance when the number of global optima increases. However, the softmax policy shows a performance drop since it assigns nonzero probability to non-optimal actions to explore the every optimal points during the exploration phase and this effect hampers the convergence of Q network to the multiple modes. The example of



(a) Expected Return (Multiple Global Optima)

(b) Reached Mode Ratio (Multiple Global Optima)

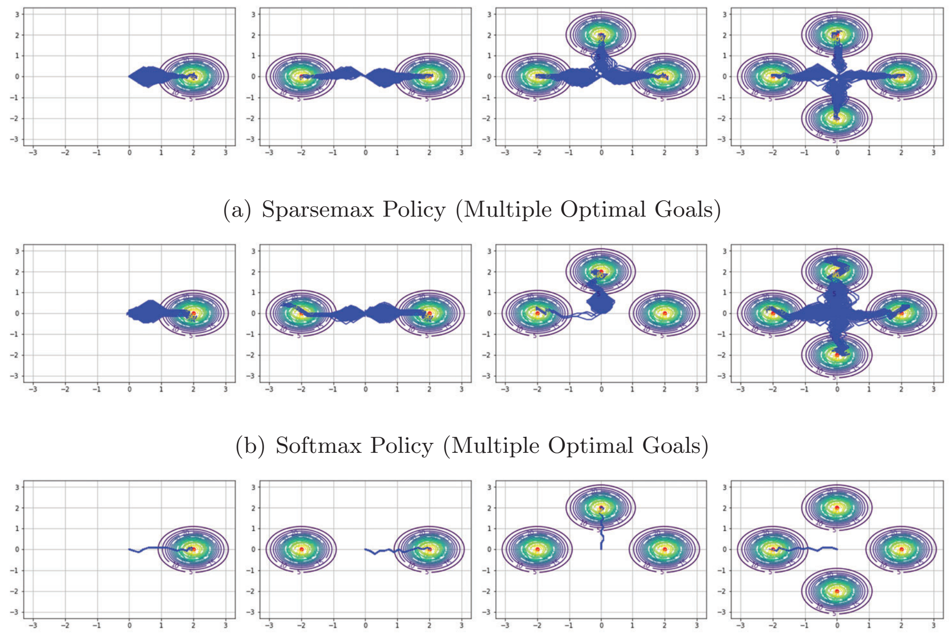
(c) Expected Return (Multiple Local Optima)

(d) Required Episodes (Multiple Local Optima)

Figure 3.5: (a) The average performance of each algorithm averaging over 500 test episodes after training with 3000 episodes. (b) The ratio of the number of given goals to the number of the goals reached by a trained policy. (c) The average performance of each algorithm averaging over 500 test episodes sampled from a greedy policy after training with 3000 episodes. (d) The number of episodes to reach the threshold value (800) during the exploration phase.

sampled trajectories are shown in Figure 3.6.

We test our algorithm on a more difficult problem to verify that our method can find the global optimum when multiple local optima exist (see Figure 3.4(b)). We designed a reward function with single global optimum and multiple local op-



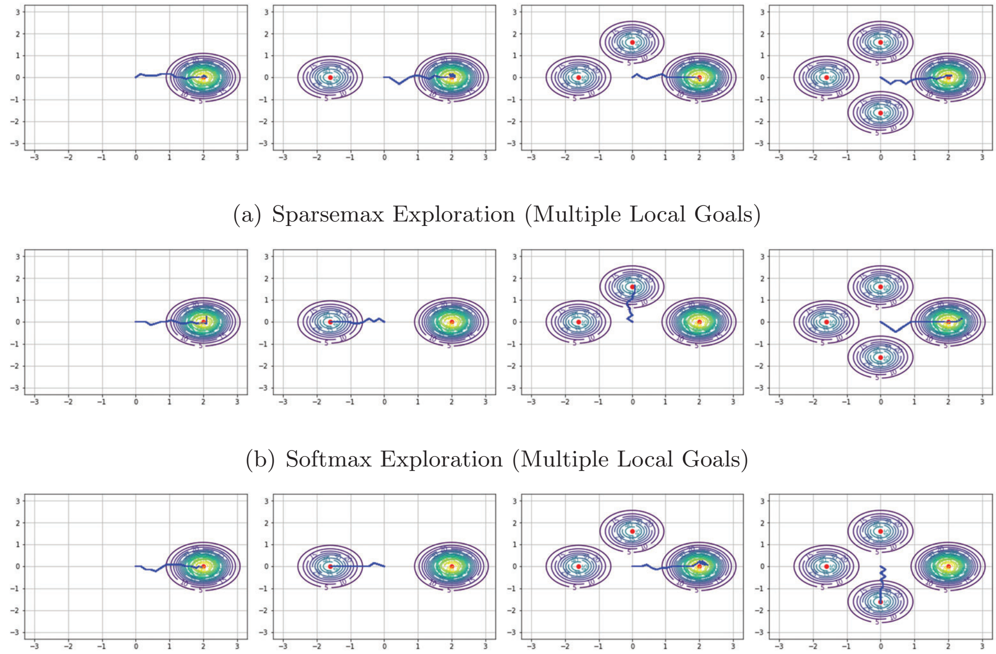
(a) Sparsemax Policy (Multiple Optimal Goals)

(b) Softmax Policy (Multiple Optimal Goals)

(c) Greedy Policy (Multiple Optimal Goals)

Figure 3.6: (a) Example trajectories sampled from the sparsemax policy distribution trained by sparsemax exploration. (b) Example trajectories sampled from the softmax policy distribution trained by softmax exploration. (c) Example trajectories sampled from the greedy policy distribution trained by ϵ-greedy exploraiton (when we sample the trajectory, ϵ is set to zero).

tima. The local optima are located near the initial state and the global optimum is farther from the initial state than the local optima are. Hence, in order to reach the global optimum, an agent should keep exploring with wide directions. We train a Q network with sparsemax, softmax, and ϵ-greedy explorations and evaluate the trained Q network with the greedy policy, i.e., $\arg\max Q(s, a)$. If exploration method can search the golobal optimum within limited episodes and the Q network converges into the global optimum, then the greedy policy will reach the global optimal point. For each algorithm, α is selected from the best value



(a) Sparsemax Exploration (Multiple Local Goals)

(b) Softmax Exploration (Multiple Local Goals)

(c) ϵ-Greedy Exploration (Multiple Local Goals)

Figure 3.7: (a) Example trajectories sampled by greedy policy trained by sparsemax exploration. (b) Example trajectories sampled by greedy policy trained by softmax exploration. (c) Example trajectories sampled by greedy policy trained by ϵ-greedy exploraiton.

among $[0.1, 1, 5, 10, 100]$ and ϵ decaying rate is also selected from the best value among $[0.99, 0.999, 0.9995, 0.9999]$ with the minimum ϵ at 0.001. The experiments are repeated with five different random seeds and the test average return and required episodes to reach the threshold average return are shown in Figure 3.5. To compute the number of episodes to reach the threshold average return, we measure the average return over the consecutive 100 episodes during the exploration phase and find the first point to cross the specific threshold average return which is set to 800. In the given problem, the expected value of local optima and global optimum are 600 and 1800, respectively. Therefore, the threshold average return, 800, indicates that some of 100 episodes reach the global optimum. In terms of the number of episodes, sparsemax exploration shows the fastest convergence to the global optima than the other methods as shown in Figure 3.5(d). As a result, it can be shown that sparsemax exploration escapes the local optima faster than the other explorations. When it comes to peformance evaluation, the Q network trained by sparsemax exploration outperforms softmax and ϵ-greedy exploration as shown in Figure 3.5(c), since sparsemax eploration reaches the global optima faster than the others.

3.1.6 Summary

In this section, we have proposed a new MDP with novel causal sparse Tsallis entropy regularization which induces a sparse and multi-modal optimal policy distribution. In addition, we have provided the full mathematical analysis of the proposed sparse MDPs, including the sparse Bellman equation, the convergence and optimality of sparse value iteration, and the performance bound between a sparse MDP and the original MDP. We have also shown that the performance gap of a sparse MDP is strictly smaller than that of a soft MDP. In experiments, we have verified that the theoretical performance gaps of a sparse MDP and soft MDP from the original MDP are correct. We have applied the sparsemax policy and sparse Bellman equation to deep Q-learning as an exploration strategy and update rule, respectively, and shown that the proposed exploration method shows significantly better performance compared to ϵ-greedy, softmax exploration, and DDPG, when the number of actions is large. From the analysis and experiments, we have demonstrated that the proposed sparse MDP can be an efficient alternative to problems with a large number of possible actions and even a continuous action space.

3.2 Sparse Policy Learning for Imitation Learning

In this section, we focus on the problem of imitating demonstrations of an expert who behaves non-deterministically depending on the situation. In imitation learning, it is often assumed that the expert's policy is deterministic. However, there are instances, especially for complex tasks, where multiple action sequences perform the same task equally well. We can model such non-deterministic behavior of an expert using a stochastic policy. For example, expert drivers normally show consistent behaviors such as keeping lane or keeping the distance from a frontal car, but sometimes they show different actions for the same situation, such as overtaking a car and turning left or right at an intersection, as suggested in [152]. Furthermore, learning multiple optimal action sequences to perform a task is desirable in terms of robustness since an agent can easily recover from failure due to unexpected events [50, 74]. In addition, a stochastic policy promotes exploration and stability during learning [53, 50, 138]. Hence, modeling experts' stochasticity can be a key factor in imitation learning.

To this end, we propose a novel maximum causal Tsallis entropy (MCTE) framework for imitation learning, which can learn from a uni-modal to multi-modal policy distribution by adjusting its supporting set. We first show that the optimal policy under the MCTE framework follows a *sparsemax* distribution [85], which has an adaptable supporting set in a discrete action space. Traditionally, the maximum causal entropy (MCE) framework [152, 19] has been proposed to model stochastic behavior in demonstrations, where the optimal policy follows a softmax distribution. However, it often assigns non-negligible probability mass to non-expert actions when the number of actions increases [74, 34]. On the contrary, as the optimal policy of the proposed method can adjust its supporting set, it can model various expert's behavior from a uni-modal distribution to a multi-modal distribution.

To apply the MCTE framework to a complex and model-free problem, we propose a maximum causal Tsallis entropy imitation learning (MCTEIL) with a sparse mixture density network (sparse MDN) whose mixture weights are modeled as a sparsemax distribution. By modeling expert's behavior using a sparse MDN, MCTEIL can learn varying stochasticity depending on the state in a continuous action space. Furthermore, we show that the MCTEIL algorithm can be obtained by extending the MCTE framework to the generative adversarial setting, similarly to generative adversarial imitation learning (GAIL) by Ho and Ermon [55], which is based on the MCE framework. The main benefit of the generative adversarial setting is that the resulting policy distribution is more robust than that of a supervised learning method since it can learn recovery behaviors from less demonstrated regions to demonstrated regions by exploring the state-action space during training. Interestingly, we also show that the Tsallis entropy of a sparse MDN has an analytic form and is proportional to the distance between mixture means. Hence, maximizing the Tsallis entropy of a sparse MDN encourages exploration by providing bonus rewards to wide-spread mixture means and penalizing collapsed mixture means, while the causal entropy [152] of an MDN is less effective in terms of preventing the collapse of mixture means since there is no analytical form and its approximation is used in practice instead. Consequently, maximizing the Tsallis entropy of a sparse MDN has a clear benefit over the causal entropy in terms of exploration and mixture utilization.

To validate the effectiveness of the proposed method, we conduct two simulation studies. In the first simulation study, we verify that MCTEIL with a sparse

MDN can successfully learn multi-modal behaviors from expert's demonstrations. A sparse MDN efficiently learns a multi-modal policy without performance loss, while a single Gaussian and a softmax-based MDN suffer from performance loss. The second simulation study is conducted using four continuous control problems in MuJoCo [133]. MCTEIL outperforms existing methods in terms of the average cumulative return. In particular, MCTEIL shows the best performance for the *reacher* problem with a smaller number of demonstrations while GAIL often fails to learn the task.

3.2.1 Related Work

The early researches on IRL [152, 1, 105, 103, 76, 150, 31, 30, 144] can be categorized into two groups: a margin based and entropy based method. A margin based method maximizes the margin between the value of the expert's policy and all other policies [1, 105]. In [1], Abbeel and Ng proposed an apprenticeship learning where the rewards function is estimated to maximize the margin between the expert's policy and randomly sampled policies. In [105], Ratliff et al. proposed the maximum margin planning (MMP) where Bellman-flow constraints are introduced to consider the margin between the experts' policy and all other possible policies. On the contrary, an entropy based method is first proposed in [152] to handle the stochastic behavior of the expert. Ziebart et al. [152] proposed a maximum entropy inverse reinforcement learning (MaxEnt IRL) using the principle of maximum (Shannon) entropy to handle ambiguity issues of IRL. Ramachandran et al. [103] proposed Bayesian inverse reinforcement learning (BIRL) where the Bayesian probabilistic model over demonstrations is proposed and the expert policy and rewards are inferred by using a Metropolis-Hastings (MH) method. In[152, 103], the expert behavior is modeled as a softmax distribution of an action value which is the optimal solution of the maximum entropy problem. We also note that [76, 150, 31, 30, 144] are variants based on [152, 103].

In [55], Ho and Ermon have extended [152] to a unified framework for two groups by adding a reward regularization. Most existing IRL methods can be interpreted as the unified framework with different reward regularization. Those methods including the aforementioned algorithms [152, 1, 105, 103, 76, 150, 31, 30, 144] require to solve an MDP problem every iterations to update a reward function. In model-free case, reinforcement learning (RL) method should be applied to solve the MDP, which leads to high computational costs and huge amounts of samples. To address this issue, Ho and Ermon proposed the generative adversarial imitation learning (GAIL) method where the policy function is updated to maximize the reward function and the reward function is updated to assign high values to expert's demonstrations and low values to trained policy's demonstrations. GAIL achieves sample efficiency by avoiding the need to solve RL as a subroutine and alternatively updating policy and reward functions.

Recently, several variants of GAIL [52, 141, 77] have been developed based on the maximum entropy framework. These methods [52, 141, 77] focus on handling the multi-modality in demonstrations by learning the latent structure. In [52], Hausman et al. proposed an imitation learning method to learn policies using unlabeled demonstrations collected from multiple different tasks where the latent intention is introduced in order to separate mixed demonstrations. Similarly, in [141], a robust imitation learning method is proposed, which separates unlabeled demonstrations by assigning the latent code using a variational autoencoder. The encoding network assigns the latent code to the input demonstration. Then, the policy network is trained to mimic the input demonstration given the latent code and the encoding network is trained to recover the given latent code from the generated trajectory. In [77], the latent code is also proposed to handle multi-modal demonstrations. The latent structure in [77] is learned by maximizing the lower bound of mutual information between the latent code and the corresponding demonstrations. Consequently, existing imitation learning methods which can handle the multi-modal behavior have common features in that they are developed based on the maximum entropy framework and capture the multi-modality of demonstrations by learning the mapping from demonstrations to the latent space.

Unlikely to recent methods for multi-modal demonstrations, the proposed method is established on the maximum causal Tsallis entropy framework which induces a sparse distribution whose supporting set can be adjusted, instead of the original maximum entropy. Furthermore, a policy is modeled as a sparse mixture density network (sparse MDN) which can learn multi-modal behavior directly instead of learning the latent structure.

3.2.2 Principle of Maximum Causal Tsallis Entropy

In this section, we formulate a maximum causal Tsallis entropy imitation learning (MCTEIL) and show that MCTE induces a sparse and multi-modal distribution which has an adaptable supporting set. The problem of maximizing the causal Tsallis entropy $W(\pi)$ can be formulated as follows:

$$\begin{aligned} \underset{\pi\in\Pi}{\text{maximize}} \quad & \alpha W(\pi) \\ \text{subject to} \quad & \mathbb{E}_{\pi}\left[\phi(s,a)\right] = \mathbb{E}_{\pi_E}\left[\phi(s,a)\right]. \end{aligned} \tag{3.10}$$

In order to derive optimality conditions, we will first change the optimization variable from a policy distribution to a state-action visitation measure. Then, we prove that the MCTE problem is concave with respect to the visitation measure. The necessary and sufficient conditions for an optimal solution are derived from the Karush-Kuhn-Tucker (KKT) conditions using the strong duality and the optimal policy is shown to be a sparsemax distribution. Furthermore, we also provide an interesting interpretation of the MCTE framework as robust Bayes estimation in terms of the Brier score. Hence, the proposed method can be viewed as maximization of the worst case performance in the sense of the Brier score [24].

We can change the optimization variable from a policy distribution to a state-action visitation measure based on the following theorem.

Theorem 6 (Theorem 2 of Syed et al. [125])**.** *Let* $\mathbf{M}$ *be a set of state-action visitation measures, i.e.,* $\mathbf{M} \triangleq \{\rho | \forall s,\ \ a,\ \ \rho(s,a) \geq 0,\ \ \sum_a \rho(s,a) = d(s) + \gamma \sum_{s',a'} T(s|s',a')\rho(s',a')\}$*. If* $\rho \in \mathbf{M}$*, then it is a state-action visitation measure for* $\pi_\rho(a|s) \triangleq \frac{\rho(s,a)}{\sum_a \rho(s,a)}$*, and* π_ρ *is the unique policy whose state-action visitation measure is* ρ*.*

The proof of Theorem 6 can be found in [125] or in Puterman [98]. Theorem 6 guarantees the one-to-one correspondence between a policy distribution and state-action visitation measure. Then, the objective function $W(\pi)$ is converted into the function of ρ as follows.

Theorem 7. *Let* $\bar{W}(\rho) = \frac{1}{2}\sum_{s,a} \rho(s,a)\left(1 - \frac{\rho(s,a)}{\sum_{a'} \rho(s,a')}\right)$*. Then, for any stationary policy* $\pi \in \Pi$ *and any state-action visitation measure* $\rho \in \mathbf{M}$*,* $W(\pi) = \bar{W}(\rho_\pi)$ *and* $\bar{W}(\rho) = W(\pi_\rho)$ *hold.*

The proof is provided in the supplementary material. Theorem 7 tells us that if $\bar{W}(\rho)$ has the maximum at ρ^*, then $W(\pi)$ also has the maximum at π_{ρ^*}. Based on Theorem 6 and 7, we can freely convert the problem (3.10) into

$$\begin{aligned} \underset{\rho \in \mathbf{M}}{\text{maximize}} \quad & \alpha \bar{W}(\rho) \\ \text{subject to} \quad & \sum_{s,a} \rho(s,a)\phi(s,a) = \sum_{s,a} \rho_E(s,a)\phi(s,a), \end{aligned} \tag{3.11}$$

where ρ_E is the state-action visitation measure corresponding to π_E.

Optimality Condition of Maximum Causal Tsallis Entropy

We show that the optimal policy of the problem (3.11) is a sparsemax distribution using the KKT conditions. In order to use the KKT conditions, we first show that the MCTE problem is concave.

Theorem 8. *$\bar{W}(\rho)$ is strictly concave with respect to $\rho \in \mathbf{M}$.*

The proof of Theorem 8 is provided in the supplementary material. Since all constraints are linear and the objective function is concave, (3.11) is a concave problem and, hence, strong duality holds. The dual problem is defined as follows:

$$\begin{aligned} &\max_{\theta,c,\lambda} \ \min_{\rho} \quad L_W(\theta, c, \lambda, \rho) \\ &\text{subject to} \quad \forall\, s, a \ \ \lambda_{sa} \geq 0, \end{aligned} \tag{3.12}$$

where $L_W(\theta, c, \lambda, \rho) = -\alpha \bar{W}(\rho) - \sum_{s,a} \rho(s,a)\theta^\intercal \phi(s,a) + \sum_{s,a} \rho_E(s,a)\theta^\intercal\phi(s,a) - \sum_{s,a} \lambda_{sa}\rho(s,a) + \sum_s c_s \left(\sum_a \rho(s,a) - d(s) - \gamma \sum_{s',a'} T(s|s',a')\rho(s',a')\right)$ and θ, c, and λ are Lagrangian multipliers and the constraints come from $\mathbf{M}$. Then, the optimal solution of primal and dual variables necessarily and sufficiently satisfy the KKT conditions.

Theorem 9. *The optimal solution of (3.11) sufficiently and necessarily satisfies the following conditions:*

$$\begin{aligned} q_{sa} :=& \sum_{s'} \left(\theta^\intercal \phi(s,a) + \gamma c_{s'}\right) T(s'|s,a), \\ c_s =& \alpha \left[\frac{1}{2} \sum_{a \in S(s)} \left(\left(\frac{q_{sa}}{\alpha}\right)^2 - \tau\left(\frac{q_s}{\alpha}\right)^2 \right) + \frac{1}{2} \right], \\ \pi_\rho(a|s) =& \max\left(\frac{q_{sa}}{\alpha} - \tau\left(\frac{q_s}{\alpha}\right), 0 \right), \end{aligned}$$

where $\pi_\rho(a|s) = \frac{\rho(s,a)}{\sum_a \rho(s,a)}$, q_{sa} *is an auxiliary variable, and* $q_s = [q_{sa_1} \cdots q_{sa_{|\mathcal{A}|}}]^\intercal$.

The optimality conditions of the problem (3.11) tell us that the optimal policy is a sparsemax distribution which assigns zero probability to an action whose auxiliary variable q_{sa} is below the threshold τ, which determines a supporting set. If expert's policy is multi-modal at state s, the resulting $\pi_{\rho}(\cdot|s)$ becomes multi-modal and induces a multi-modal distribution with a large supporting set. Otherwise, the resulting policy has a sparse and smaller supporting set. Therefore, a sparsemax policy has advantages over a softmax policy for modeling sparse and multi-modal behaviors of an expert whose supporting set varies according to the state.

Furthermore, we also discover an interesting connection between the optimality condition of an MCTE problem and the sparse Bellman optimality condition in Theorem 1. Since the optimality condition is equivalent to the sparse Bellman optimality equation [74], we can compute the optimal policy and Lagrangian multiplier c_s by solving a sparse MDP under the reward function $\mathbf{r}(s,a) = \theta^{*\intercal}\phi(s,a)$, where θ^{*} is the optimal dual variable. In addition, c_s and q_{sa} can be viewed as a state value and state-action value for the reward $\theta^{*\intercal}\phi(s,a)$, respectively.

Interpretation as Robust Bayes

In this section, we provide an interesting interpretation about the MCTE framework. In general, maximum entropy estimation can be viewed as a minimax game between two players. One player is called a decision maker and the other player is called the nature, where the nature assigns a distribution to maximize the decision maker's misprediction while the decision maker tries to minimize it [45]. The same interpretation can be applied to the MCTE framework. We show that the proposed MCTE problem is equivalent to a minimax game with the Brier score [24].

Theorem 10. *The maximum causal Tsallis entropy distribution minimizes the* *worst case prediction Brier score,*

$$\begin{aligned}
&\min_{\pi\in\Pi}\max_{\tilde{\pi}\in\Pi}\ \mathbb{E}_{\tilde{\pi}}\left[\sum_{a'}\frac{1}{2}\left(\mathbb{I}_{\{a'=a\}}-\pi(a|s)\right)^{2}\right] \\
&\text{subject to }\ \mathbb{E}_{\pi}\left[\phi(s,a)\right]=\mathbb{E}_{\pi_E}\left[\phi(s,a)\right]
\end{aligned} \tag{3.13}$$

where $\sum_{a'}\frac{1}{2}\left(\mathbb{I}_{\{a'=a\}}-\pi(a|s)\right)^{2}$ *is the Brier score.*

Note that minimizing the Brier score minimizes the misprediction ratio while we call it a score here. Theorem 10 is a straightforward extension of the robust Bayes results in [45] to sequential decision problems. This theorem tells us that the MCTE problem can be viewed as a minimax game between a sequential decision maker π and the nature $\tilde{\pi}$ based on the Brier score. In this regards, the resulting estimator can be interpreted as the best decision maker against the worst that the nature can offer.

3.2.3 Maximum Causal Tsallis Entropy Imitation Learning

In this section, we propose a maximum causal Tsallis entropy imitation learning (MCTEIL) algorithm to solve a model-free IL problem in a continuous action space. In many real-world problems, state and action spaces are often continuous and transition probability of a world cannot be accessed. To apply the MCTE framework for a continuous space and model-free case, we follow the extension of GAIL [55], which trains a policy and reward alternatively, instead of solving RL at every iteration. We extend the MCTE framework to a more general case with reward regularization and it is formulated by replacing the causal entropy $H(\pi)$ in the problem (2.12) with the causal Tsallis entropy $W(\pi)$ as follows:

$$\max_{\theta}\min_{\pi\in\Pi}\quad -\alpha W(\pi)-\mathbb{E}_{\pi}\left[\theta^{\intercal}\phi(s,a)\right]+\mathbb{E}_{\pi_E}\left[\theta^{\intercal}\phi(s,a)\right]-\psi(\theta). \tag{3.14}$$

Similarly to [55], we convert the problem (3.14) into the generative adversarial setting as follows.

Theorem 11. *The maximum causal sparse Tsallis entropy problem (3.14) is equivalent to the problem:*

$$\min_{\pi\in\Pi} \; \psi^{*}\left(\mathbb{E}_{\pi}\left[\phi(s,a)\right] - \mathbb{E}_{\pi_E}\left[\phi(s,a)\right]\right) - \alpha W(\pi),$$

where $\psi^{*}(x) = \sup_{y}\{y^{\intercal}x - \psi(y)\}$*.*

The proof is detailed in the supplementary material. The proof of Theorem 11 depends on the fact that the objective function of (3.14) is concave with respect to ρ and is convex with respect to θ. Hence, we first switch the optimization variables from π to ρ and, using the minimax theorem [86], the maximization and minimization are interchangeable and the generative adversarial setting is derived. Similarly to [55], Theorem 11 says that a MCTE problem can be interpreted as minimization of the distance between expert's feature expectation and training policy's feature expectation, where $\psi^{*}(x_1 - x_2)$ is a proper distance function since $\psi(x)$ is a convex function. Let $e_{sa} \in \mathbb{R}^{|\mathcal{S}||\mathcal{A}|}$ be a feature indicator vector, such that the sath element is one and zero elsewhere. If we set ψ to $\psi_{GA}(\theta) \triangleq \mathbb{E}_{\pi_E}[g(\theta^{\intercal}e_{sa})]$, where $g(x) = -x - \log(1-e^{x})$ for $x < 0$ and $g(x) = \infty$ for $x \geq 0$, we can convert the MCTE problem into the following generative adversarial setting:

$$\min_{\pi\in\Pi}\max_{\mathbf{D}} \quad \mathbb{E}_{\pi}\left[\log(\mathbf{D}(s,a))\right] + \mathbb{E}_{\pi_E}\left[\log(1-\mathbf{D}(s,a))\right] - \alpha W(\pi), \tag{3.15}$$

where $\mathbf{D}$ is a discriminator. The problem (3.15) can be solved by MCTEIL which consists of three steps. First, trajectories are sampled from the training policy π_ν and discriminator $\mathbf{D}_\omega$ is updated to distinguish whether the trajectories are generated by π_ν or π_E. Finally, the training policy π_ν is updated with a policy optimization method under the sum of rewards $\mathbb{E}_{\pi}\left[-\log(\mathbf{D}_\omega(s,a))\right]$ with a causal Tsallis entropy bonus $\alpha W(\pi_\nu)$. The algorithm is summarized in Algorithm 2.

Algorithm 2 Maximum Causal Tsallis Entropy Imitation Learning

```
Expert's demonstrations $\mathcal{D}$ are given
Initialize policy and discriminator parameters $\nu, \omega$
while until convergence do
    Sample trajectories $\{\zeta\}$ from $\pi_\nu$
    Update $\omega$ with the gradient of $\sum_{\{\zeta\}} \log(\mathbf{D}_\omega(s,a)) + \sum_{\mathcal{D}} \log(1 - \mathbf{D}_\omega(s,a))$.
    Update $\nu$ using a policy optimization method with reward function $-\mathbb{E}_{\pi_\nu}\left[\log(\mathbf{D}_\omega(s,a))\right] + \alpha W(\pi_\nu)$
end while
```

Sparse Mixture Density Network We further employ a novel mixture density network (MDN) with sparsemax weight selection, which can model sparse and multi-modal behavior of an expert, which is called a sparse MDN. In many imitation learning algorithms, a Gaussian network is often employed to model expert's policy in a continuous action space. However, a Gaussian distribution is inappropriate to model the multi-modality of an expert since it has a single mode. An MDN is more suitable for modeling a multi-modal distribution. In particular, a sparse MDN is a proper extension of a sparsemax distribution for a continuous action space. The input of a sparse MDN is state s and the output of a sparse MDN is components of K mixtures of Gaussians: mixture weights $\{w_i\}$, means $\{\mu_i\}$, and covariance matrices $\{\Sigma_i\}$. A sparse MDN policy is defined as

$$\pi(a|s) = \sum_{i}^{K} w_i(s)\mathcal{N}(a; \mu_i(s), \Sigma_i(s)),$$

where $\mathcal{N}(a; \mu, \Sigma)$ indicates a multivariate Gaussian density at point a with mean μ and covariance Σ. In our implementation, $w(s)$ is computed as a sparsemax distribution, while most existing MDN implementations utilize a softmax distribution. Modeling the expert's policy using an MDN with K mixtures can be interpreted as separating continuous action space into K representative actions.

Since we model mixture weights using a sparsemax distribution, the number of mixtures used to model the expert's policy can vary depending on the state. In this regards, the sparsemax weight selection has an advantage over the soft weight selection since the former utilizes mixture components more efficiently as unnecessary components will be assigned with zero weights.

Tsallis Entropy of Mixture Density Network An interesting fact is that the causal Tsallis entropy of an MDN has an analytic form while the Gibbs-Shannon entropy of an MDN is intractable.

Theorem 12. *Let* $\pi(a|s) = \sum_i^K w_i(s)\mathcal{N}(a; \mu_i(s), \Sigma_i(s))$ *and* $\rho_\pi(s) = \sum_a \rho_\pi(s, a)$*. Then,*

$$W(\pi) = \frac{1}{2}\sum_s \rho_\pi(s)\left(1 - \sum_i^K \sum_j^K w_i(s)w_j(s)\mathcal{N}\left(\mu_i(s); \mu_j(s), \Sigma_i(s) + \Sigma_j(s)\right)\right). \quad (3.16)$$

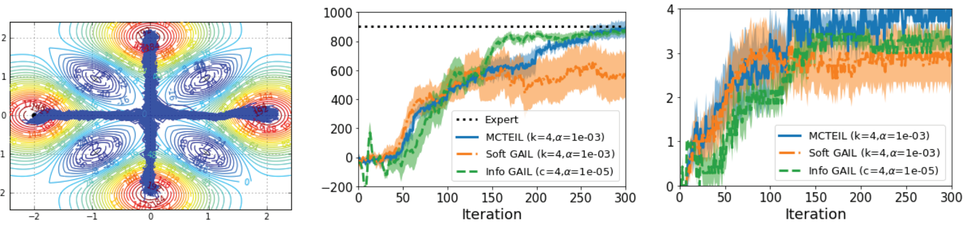
The proof is included in the supplementary material. The analytic form of the Tsallis entropy shows that the Tsallis entropy is proportional to the distance between mixture means. Hence, maximizing the Tsallis entropy of a sparse MDN encourages exploration of diverse directions during the policy optimization step of MCTEIL. In imitation learning, the main benefit of the generative adversarial setting is that the resulting policy is more robust than that of supervised learning since it can learn how to recover from a less demonstrated region to a demonstrated region by exploring the state-action space during training. Maximum Tsallis entropy of a sparse MDN encourages efficient exploration by giving bonus rewards when mixture means are spread out. (3.16) also has an effect of utilizing mixtures more efficiently by penalizing for modeling a single mode using several mixtures. Consequently, the Tsallis entropy $W(\pi)$ has clear benefits in terms of both exploration and mixture utilization.

3.2.4 Experiments

To verify the effectiveness of the proposed method, we compare MCTEIL with several other imitation learning methods. First, we use behavior cloning (BC) as a baseline. Second, generative adversarial imitation learning (GAIL) with a single Gaussian distribution is compared. We also compare a straightforward extension of GAIL for a multi-modal policy by using a softmax weighted mixture density network (soft MDN) in order to validate the efficiency of the proposed sparsemax weighted MDN. In soft GAIL, due to the intractability of the causal entropy of a mixture of Gaussians, we approximate the entropy term by adding $-\alpha \log(\pi(a_t|s_t))$ to $-\log(\mathbf{D}(s_t, a_t))$ since $\mathbb{E}_\pi\left[-\log(\mathbf{D}(s,a))\right] + \alpha H(\pi) = \mathbb{E}_\pi\left[-\log(\mathbf{D}(s,a)) - \alpha\log(\pi(a|s))\right]$. We also compare info GAIL [77] which learns simultaneously both policy and the latent structure of experts' demonstrations. In info GAIL, a posterior distribution of a latent code is learned to cluster multi-modal demonstrations. The posterior distribution is trained to consistently assign the latent code to similar demonstrations and Once the latent codes are assigned to the demonstrations, the policy function conditioned on a latent code is trained to generate the corresponding demonstrations. Different modes in demonstrations are captured by assigning different latent codes.

Multi-Goal Environment

To validate that the proposed method can learn multi-modal behavior of an expert, we design a simple multi-goal environment with four attractors and four repulsors, where an agent tries to reach one of attractors while avoiding all repulsors as shown in Figure 3.8(a). The agent follows the point-mass dynamics and get a positive reward (resp., a negative reward) when getting closer to an attractor (resp., repulsor). Intuitively, this problem has multi-modal optimal ac-

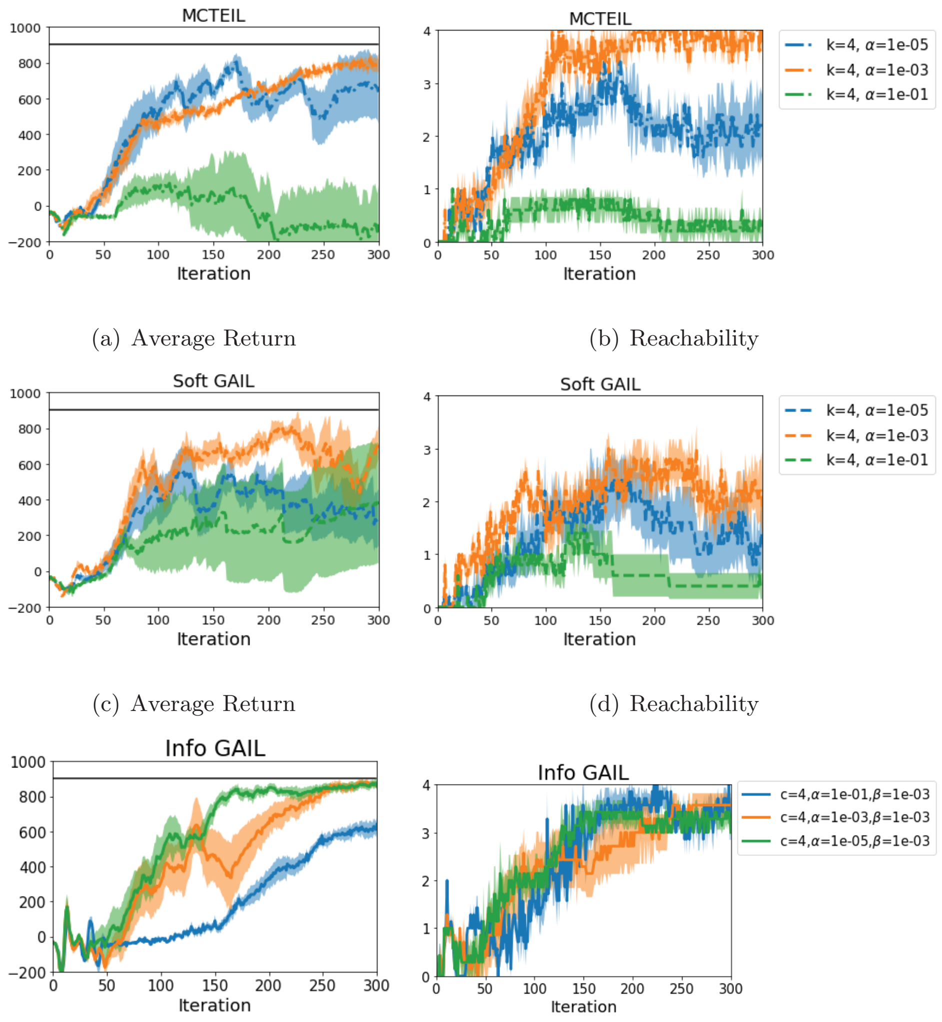


(a) Multi-Goal Environment (b) Average Return (c) Reachability

Figure 3.8: (a) The environment and multi-modal demonstrations are shown. The contour shows the underlying reward map. (b) The average return during training. (c) The reachability during training, where k is the number of mixtures, c is a dimension of the latent code, and α is a regularization coefficient.

tions at the center. We first train the optimal policy using [74] and generate 300 demonstrations from the expert's policy. For tested methods, 500 episodes are sampled at each iteration. In every iteration, we measure the average return using the underlying rewards and the reachability which is measured by counting how many goals are reached. If the algorithm captures the multi-modality of expert's demonstrations, then, the resulting policy will show high reachability. All algorithms run repeatedly with seven different random seeds.

The results are shown in Figure 3.8(b) and 3.8(c). Since the rewards are multi-modal, it is easy to get a high return if the algorithm learns only uni-modal behavior. Hence, the average returns of soft GAIL, info GAIL and MCTEIL increases similarly. However, when it comes to the reachability, MCTEIL outperforms other methods when they use the same number of mixtures. In particular, MCTEIL can learn all modes in demonstrations at the end of learning while soft GAIL and info GAIL suffer from collapsing modes. This advantage clearly comes from the maximum Tsallis entropy of a sparse MDN since the analytic form of the Tsallis entropy directly penalizes collapsed mixture means while $-\log(\pi(a|s))$



(a) Average Return

(b) Reachability

(c) Average Return

(d) Reachability

(e) Average Return

(f) Reachability

Figure 3.9: Average return and reachability of MCTEIL, soft GAIL, and info GAIL, respectively. k indicates the number of mixtures, α indicates an entropy regularization coefficient, and c indicates a dimension of the latent code of Info GAIL.

indirectly prevents modes collapsing in soft GAIL. Furthermore, info-GAIL also shows mode collapsing while the proposed method can learn every modes. Since info-GAIL has to train a posterior distribution over the latent code to separate demonstrations, it requires more iterations for reaching all modes as well as prone to the mode collapsing problems. Consequently, we can conclude that the MCTEIL efficiently utilizes each mixture for wide-spread exploration. The experimental results with other hyperparameters are shown in Figure 3.9.

Continuous Control Environment

We test MCTEIL with a sparse MDN on MuJoCo [133], which is a physics-based simulator, using *Halfcheetah, Walker2d, Reacher*, and *Ant*. We train the expert policy distribution using trust region policy optimization (TRPO) [113] under the true reward function and generate 50 demonstrations from the expert policy. We run algorithms with varying numbers of demonstrations, $4, 11, 18$, and 25, and all experiments have been repeated three times with different random seeds. To evaluate the performance of each algorithm, we sample 50 episodes from the trained policy and measure the average return value using the underlying rewards. For methods using an MDN, we use the best number of mixtures using a brute force search.

The results are shown in Figure 3.10. For three problems, except Walker2d, MCTEIL outperforms the other methods with respect to the average return as the number of demonstrations increases. For Walker2d, MCTEIL and soft GAIL show similar performance. Especially, in the reacher problem, we obtain the similar results reported in [55], where BC works better than GAIL. However, our method shows the best performance for all demonstration counts. It is observed that the MDN policy tends to show high performance consistently since MCTEIL and soft GAIL are consistently ranked within the top two high performing algorithms. From these results, we can conclude that an MDN policy explores better than

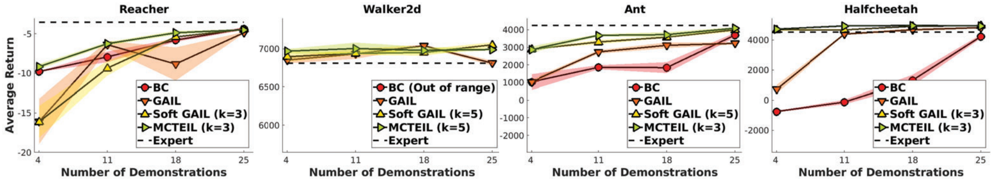


Figure 3.10: Average returns of trained policies. For soft GAIL and MCTEIL, k indicates the number of mixture and α is an entropy regularization coefficient. A dashed line indicates the performance of an expert.

a single Gaussian policy since an MDN can keep searching multiple directions during training. In particular, since the maximum Tsallis entropy makes each mixture mean explore in different directions and a sparsemax distribution assigns zero weight to unnecessary mixture components, MCTEIL efficiently explores and shows better performance compared to soft GAIL with a soft MDN. Consequently, we can conclude that MCTEIL outperforms other imitation learning methods and the causal Tsallis entropy has benefits over the causal Gibbs-Shannon entropy as it encourages exploration more efficiently.

3.2.5 Summary

In this section, we have proposed a novel maximum causal Tsallis entropy (MCTE) framework, which induces a sparsemax distribution as the optimal solution. We have also provided the full mathematical analysis of the proposed framework, including the concavity of the problem, the optimality condition, and the interpretation as robust Bayes. We have also developed the maximum causal Tsallis entropy imitation learning (MCTEIL) algorithm, which can efficiently solve a MCTE problem in a continuous action space since the Tsallis entropy of a mixture of Gaussians encourages exploration and efficient mixture utilization. In experiments, we have verified that the proposed method has advantages over existing methods for learning the multi-modal behavior of an expert since a sparse MDN can search in diverse directions efficiently. Furthermore, the proposed method has outperformed BC, GAIL, and GAIL with a soft MDN on the standard IL problems in the MuJoCo environment. From the analysis and experiments, we have shown that the proposed MCTEIL method is an efficient and principled way to learn the multi-modal behavior of an expert.

Chapter 4

Entropy-based Exploration

4.1 Generalized Tsallis Entropy Reinforcement Learning

Reinforcement learning (RL) combined with a powerful function approximation technique like a neural network has shown noticeable successes on challenging sequential decision making problems, such as playing a video game [87], learning complex control [37, 46], and realistic motion generation [96]. A model-free RL algorithm aims to learn a policy to effectively perform a given task through the trial and error without the prior knowledge about the environment, where the performance of policy is often measured by the sum of rewards. The absence of the environmental information gives a rise to innate trade-off between exploration and exploitation during a learning process. If the algorithm decides to explore the environment, then, it will lose the chance to exploit the best decision based on collected experiences and vice versa. Such trade-off should be appropriately scheduled to learn an optimal policy through a small number of interactions with an environment.

For the sake of efficient exploration, many RL algorithms employ maximization of the Shannon-Gibbs (SG) entropy of a policy distribution [88, 39, 50, 115, 90, 51, 94, 35]. Maximizing the SG entropy penalizes a greedy policy and encourages the exploration, thus, it helps to find a global optimal policy and to learn diverse or multi-modal behaviors where the resulting policy is robust against unexpected changes in the environment. However, it is also known that the maximum SG entropy causes a performance loss since it hinders exploiting the best action to maximize the reward [74]. To handle this issue, an alternative way has been proposed using a sparse Tsallis (ST) entropy [74, 34], which is a special case of the Tsallis entropy [135]. The ST entropy encourages exploration while penalizing less on a greedy policy, compared to the SG entropy. However, unlike the SG entropy, the ST entropy may discover a sub-optimal policy since it enforces the algorithm to explore the environment less [74, 34].

In this section, we present a unified framework for maximum entropy RL problems with various types of entropy. The proposed framework is formulated as a new class of Markov decision processes with Tsallis entropy maximization, which is called Tsallis MDPs. The Tsallis entropy generalizes the standard SG entropy and can represent various types of entropy, including the SG and ST entropies by controlling a parameter, called an *entropic index*. A Tsallis MDP presents a unifying view on the use of various entropies in RL. We provide a comprehensive analysis of how a different value of the entropic index can provide a different type of optimal policies and different Bellman optimality equations. Our theoretical result allows us to interpret the effects of various entropies for an RL problem.

The proposed Tsallis RL framework contains both SG and ST entropy maximization as special cases and allows more diverse range of exploration-exploitation trade-off behaviors for a learning agent, which is a highly desirable feature since the problem complexity is different for each task at hand. We empirically show that there exists an appropriate entropic index for each task and using a proper entropic index outperforms existing actor-critic methods. Furthermore, we also propose a scheduling method for an entropic index to alleviate the demand on finding a suitable entropic index, and demonstrate that the proposed method with the scheduled entropic index achieves the performance of TAC with the best entropic index.

Furthermore, we apply Tsallis entropy reinforcement learning for learning a controller of a soft mobile robot. soft mobile robots have the potential to overcome challenging navigation tasks that conventional rigid robots are hard to achieve, such as exploring complex and unstructured environments, by using their high adaptability and robustness against changes around them [67]. Especially, a soft mobile robot using pneumatic actuators, which provide relatively high force-to-weight ratios, have been widely developed [97, 66].

Despite the fact that the pneumatic actuators combined with soft materials are beneficial to the adaptability and robustness of soft mobile robots, their behaviors are often hard to be modeled or controlled using a traditional method such as a feedback control [131], due to their inherent stochasticity. Furthermore, the efficiency of exploration becomes more important when training a soft mobile robot, as the properties of soft material can be changed or degraded if a robot exceeds its durability. In this application, we empirically prove that TAC with linear curriculum outperforms existing methods and can learn the controller of the soft mobile robot within moderate time.

Related Work

Recently, regularization on a policy function has been widely investigated in RL [14, 88, 115, 90, 50, 91, 51, 39, 35, 74, 34, 41, 16]. The main purpose of regularizing a policy is to encourage exploration by inducing a stochastic policy from regularization. If a policy converges to a greedy policy before collecting enough information about an environment, its behavior can be sub-optimal. This issue can be efficiently handled by a stochastic policy induced from a regularization.

The SG entropy has been widely used as a policy regularization. It has been empirically shown that maximizing the SG entropy of a policy along with reward maximization encourages exploration since the entropy maximization penalizes a greedy behavior [88]. In [39], it was also demonstrated that maximizing the SG entropy helps to learn diverse and useful behaviors. This penalty from the SG entropy also helps to capture the multi-modal behavior where the resulting policy is robust against unexpected changes in the environment [50]. Theoretically, [115, 90, 50, 51] have shown that the optimal solution of maximum entropy RL has a softmax distribution of state-action values, not a greedy policy. [51] showed that the SG entropy has the benefits over exploring a continuous action space, however, the performance of SAC is sensitive to a regularization coefficient. Furthermore, the maximum SG entropy in RL provides the connection between policy gradient and value-based learning [115, 94]. [35] have also shown that maximum entropy induces a smoothed Bellman operator and it helps stable convergence of value function estimation.

While the SG entropy in RL provides better exploration, numerical stability, and capturing multiple optimal actions, it is known that the maximum SG entropy causes a performance loss since it hinders exploiting the best action to maximize the reward [74, 34]. Such drawback is often handled by scheduling a coefficient of the SG entropy to progressively vanish [29]. However, designing a proper decaying schedule is still a demanding task in that it often requires an additional validation step in practice. [44] handled the same issue by automatically manipulating the importance of actions using mutual information. On the other hand, [74] and [34] have proposed an alternative way to handle the exploitation issue of the SG entropy using a sparse Tsallis (ST) entropy, which is a special case of the Tsallis entropy [135]. The ST entropy encourages exploration while penalizing less on a greedy policy, compared to the SG entropy. However, unlike the SG entropy, the ST entropy may discover a sub-optimal policy since it enforces the algorithm to explore the environment less [74, 34].

Recently, an analysis of general concave regularization of a policy function has been investigated [14, 91, 41]. [14] proposes dynamics programming for regularized MDPs and provides theoretical guarantees for finite state-action spaces. While the theory was derived for general concave regularizer, only SG entropy-based algorithm is demonstrated on a simple grid world example [14]. [91] also applied an SG entropy-based algorithm to a simple discrete action space. In contrast to prior work [14, 91, 41], we focus on analyzing the Tsallis entropy in MDPs and RL[1]. We derive unique properties of the Tsallis entropy such as performance bounds. We also propose two dynamic programming algorithms and extend it to a continuous actor-critic method and empirically show that the proposed method outperforms the SG entropy-based method.

q-Exponential, q-Logarithm, and Tsallis Entropy

Before defining the Tsallis entropy, let us first introduce variants of exponential and logarithm functions, which are called q-exponential and q-logarithm, respec-

[1] Note that the Tsallis entropy also provides concave regularization.

tively. They are used to define the Tsallis entropy and defined as follows[2]:

$$\begin{aligned} \exp_q(x) &:= [1+(q-1)x]_+^{\frac{1}{q-1}}, \\ \ln_q(x) &:= (x^{q-1}-1)/(q-1), \end{aligned} \tag{4.1}$$

where $[x]_+ = \max(x, 0)$ and q is a real number. Note that, for $q = 1$, q-logarithm and q-exponential are defined as their limitations, i.e., $\ln_1(x) \triangleq \lim_{q\to1} \ln_q(x) = \ln(x)$ and $\exp_1(x) \triangleq \lim_{q\to1} \exp_q(x) = \exp(x)$. Furthermore, when $q = 2$, $\exp_2(x)$ and $\ln_2(x)$ become a linear function. This property gives some clues that the entropy defined using $\ln_q(x)$ will generalize the SG (or ST) entropy and, furthermore, the proposed method can generalize an actor critic method using SG entropy [51] and ST entropy [74, 34].

Now, we define the Tsallis entropy using $\ln_q(x)$.

Definition 1 (Tsallis Entropy [9])**.** *The Tsallis entropy of a random variable X with the distribution P is defined as*

$$S_q(P) \triangleq \mathop{\mathbb{E}}_{X\sim P}\left[-\ln_q(P(X))\right].$$

q is called an entropic-index.

The Tsallis entropy can represent various types of entropy by varying the *entropic index*. For example, when $q \to 1$, $S_1(P)$ becomes the Shannon-Gibbs entropy and when $q = 2$, $S_2(P)$ becomes the sparse Tsallis entropy [74]. Furthermore, when $q \to \infty$, $S_q(P)$ converges to zero. We would like to emphasize that, for $q > 0$, the Tsallis entropy is a concave function with respect to the density function, but, for $q \leq 0$, the Tsallis entropy is a convex function. Detail proofs are included in the supplementary material. In this section, we only consider the

[2] Note that the definition of $\exp_q, \ln_q$, and the Tsallis entropy are different from the original one [9] but those settings can be recovered by setting $q = 2 - q'$, where q' is the entropic index used in [9].

case when $q > 0$ since our purpose of using the Tsallis entropy is to give a bonus reward to a stochastic policy.

4.1.1 Maximum Generalized Tsallis Entropy in MDPs

In this section, we formulate MDPs with Tsallis entropy maximization, which will be named Tsallis MDPs. We mainly focus on deriving the optimality conditions and algorithms generalized for the entropic index so that a wide range of q values can be used for a learning agent. First, we extend the definition of the Tsallis entropy so that it can be applicable for a policy distribution in MDPs. The Tsallis entropy of a policy distribution π is defined by $S_q^\infty(\pi) \triangleq \mathbb{E}_{\tau\sim P,\pi}\left[\sum_{t=0}^\infty \gamma^t S_q(\pi(\cdot|s_t))\right]$. Using S_q^∞, the original MDPs can be converted into Tsallis MDPs by adding $S_q^\infty(\pi)$ to the objective function as follows:

$$\max_{\pi\in\Pi} \mathop{\mathbb{E}}_{\tau\sim P,\pi}\left[\sum_t^\infty \gamma^t \mathbf{R}_t\right] + \alpha S_q^\infty(\pi), \tag{4.2}$$

where $\alpha > 0$ is a coefficient. A state value and state-action value are redefined for Tsallis MDPs as follows:

$$V_q^\pi(s) := \mathop{\mathbb{E}}_{\tau\sim P,\pi}\left[\sum_{t=0}^\infty \gamma^t \left(\mathbf{R}_t + \alpha S_q(\pi(\cdot|s_t)\right)\middle| s_0 = s\right]$$

and

$$Q_q^\pi(s,a) := \mathop{\mathbb{E}}_{\tau\sim P,\pi}\left[\mathbf{R}_0 + \sum_{t=1}^\infty \gamma^t \left(\mathbf{R}_t + \alpha S_q(\pi(\cdot|s_t)\right)\middle| s_0 = s, a_0 = a\right],$$

where q is the entropic index. The goal of a Tsallis MDP is to find an optimal policy distribution that maximizes both the sum of rewards and the Tsallis entropy whose importance is determined by α. The solution of the problem (4.2) is denoted as $\pi_q^\star$ and its value functions are denoted as $V_q^\star = V_q^{\pi_q^\star}$ and $Q_q^\star = Q_q^{\pi_q^\star}$, respectively. In our analysis, α is set to one, however one can easiliy generalize the case of $\alpha \neq 1$ by replacing $\mathbf{r}, V$, and Q with $\mathbf{r}/\alpha, V/\alpha$, and Q/α, respectively.

In the following sections, we first derive the optimality condition of (4.2), which will be called the Tsallis-Bellman optimality (TBO) equation. Second, dynamic programming to solve Tsallis MDPs is proposed with convergence and optimality guarantees. Finally, we provide the performance error bound of the optimal policy of the Tsallis MDP, where the error is caused by the additional entropy regularization term. The theoretical results derived in this section are extended to a practical actor-critic algorithm in Section 4.1.3.

q-Maximum Operator

Before analyzing an MDP with the Tsallis entropy, we define an operator, which is called q-maximum. A q-maximum operator is a bounded approximation of the maximum operator. For a function $f(x)$, q-maximum is defined as follows:

$$q\text{-}\max_{x}\left(f(x)\right) := \max_{P\in\Delta}\left[\mathop{\mathbb{E}}_{X\sim P}\left[f(X)\right] + S_q(P)\right], \tag{4.3}$$

where Δ is a probability simplex whose element is a probability. The following theorem shows the relationship between q-maximum and maximum operators.

Theorem 13. *For any function $f(x)$ defined on a finite input space $\mathcal{X}$, the q-maximum satisfies the following inequalities.*

$$q\text{-}\max_{x}\left(f(x)\right) + \ln_q\left(1/|\mathcal{X}|\right) \leq \max_{x}(f(x)) \leq q\text{-}\max_{x}\left(f(x)\right), \tag{4.4}$$

where $|\mathcal{X}|$ is the cardinality of $\mathcal{X}$.

The proof can be found in the supplementary material. The proof of Theorem 13 utilizes the definition of q-maximum. This boundedness property will be used to analyze the performance bound of an MDP with the maximum Tsallis entropy. The solution of q-maximum is obtained as $P(x) = \exp_q\left(f(x)/q - \psi_q\left(f/q\right)\right)$, where $\psi_q(\cdot)$ is called a q-potential function [9], which is uniquely determined by the normalization condition:

$$\sum_{x \in X} P(x) = \sum_{x \in X} \exp_q \left(f(x)/q - \psi_q \left(f/q \right) \right) = 1. \tag{4.5}$$

A detail derivation can be found in the supplementary material. The property of q-maximum and the solution of q-maximum plays an important role in the optimality condition of Tsallis MDPs.

Tsallis Bellman Optimality Equation

Using the q-maximum operator, the optimality condition of a Tsallis MDP can be obtained as follows.

Theorem 14. *For $q > 0$, an optimal policy $\pi_q^\star$ and optimal value $V_q^\star$ sufficiently and necessarily satisfy the following Tsallis-Bellman optimality (TBO) equations:*

$$\begin{aligned} Q_q^\star(s,a) &= \mathop{\mathbb{E}}_{s' \sim P}[\mathbf{r}(s,a,s') + \gamma V_q^\star(s')|s,a], \\ V_q^\star(s) &= q\text{-}\max_a(Q_q^\star(s,a)), \\ \pi_q^\star(a|s) &= \exp_q \left(Q_q^\star(s,a)/q - \psi_q \left(Q_q^\star(s,\cdot)/q \right) \right), \end{aligned} \tag{4.6}$$

where ψ_q is a q-potential function.

The proof can be found in the supplementary material. The TBO equation differs from the original Bellman equation in that the maximum operator is replaced by the q-maximum operator. The optimal state value $V_q^\star$ is the q-maximum of the optimal state-action value $Q_q^\star$ and the optimal policy $\pi_q^\star$ is the solution of q-maximum (4.3). Thus, as q changes, $\pi_q^\star$ can represent various types of q-exponential distributions. We would like to emphasize that the TBO equation becomes the original Bellman equation as q diverges into infinity. This is a reasonable tendency since, as $q \to \infty$, S_∞ tends zero and the Tsallis MDP becomes the original MDP. Furthermore, when $q \to 1$, q-maximum becomes the log-sum-exponential operator and the Bellman equation of maximum SG entropy RL, (a.k.a. soft Bellman equation) [50] is recovered. When $q = 2$, the Bellman equation of maximum ST entropy RL, (a.k.a. sparse Bellman equation) [74] is also recovered. Moreover, our result guarantees that the TBO equation holds for all $q > 0$.

4.1.2 Dynamic Programming for Tsallis MDPs

In this section, we develop dynamic programming algorithms for a Tsallis MDP: Tsallis policy iteration (TPI) and Tsallis value iteration (TVI). These algorithms can compute an optimal value and policy. TPI is a policy iteration method which consists of policy evaluation and policy improvement. TVI is a value iteration method that computes the optimal value directly. In the dynamic programming of the original MDPs, the convergence is derived from the maximum operator. Similarly, in the MDP with the SG entropy, log-sum-exponential plays a crucial role for the convergence. In TPI and TVI, we generalize such maximum or log-sum-exponential operators by the q-max operator, which is a more abstract notion and available for all $q > 0$. Note that proofs of all theorems in this section are provided in the supplementary material.

Tsallis Policy Iteration

We first discuss the policy evaluation method in a Tsallis MDP, which computes V_q^π and Q_q^π for fixed policy π. Similar to the original MDP, a value function of a Tsallis MDP can be computed using the expectation equation defined by

$$\begin{aligned} Q_q^\pi(s,a) &= \mathop{\mathbb{E}}_{s'\sim P}[\mathbf{r}(s,a,s') + \gamma V_q^\pi(s')|s,a], \\ V_q^\pi(s) &= \mathop{\mathbb{E}}_{a\sim\pi}[Q_q^\pi(s,a) - \ln_q(\pi(a|s))], \end{aligned} \tag{4.7}$$

where $s' \sim P$ indicates $s' \sim P(\cdot|s,a)$ and $a \sim \pi$ indicates $a \sim \pi(\cdot|s)$. Equation (4.7) will be called the Tsallis Bellman expectation (TBE) equation and it is derived from the definition of V_q^π and Q_q^π. Based on the TBE equation, we can define the operator for an arbitrary function $F(s,a)$ over $\mathcal{S} \times \mathcal{A}$, which is called the TBE operator,

$$\begin{aligned} \left[\mathcal{T}_q^\pi F\right](s,a) &\triangleq \mathop{\mathbb{E}}_{s'\sim P}[\mathbf{r}(s,a,s') + \gamma V_F(s')|s,a], \\ V_F(s) &\triangleq \mathop{\mathbb{E}}_{a\sim\pi}[F(s,a) - \ln_q(\pi(a|s))]. \end{aligned} \tag{4.8}$$

Then, the policy evaluation method for a Tsallis MDP can be simply defined as repeatedly applying the TBE operator to an initial function F_0,i.e., $F_{k+1} = \mathcal{T}_q^\pi F_k$.

Theorem 15 (Tsallis Policy Evaluation)**.** *For fixed π and $q > 0$, consider the TBE operator $\mathcal{T}_q^\pi$, and define Tsallis policy evaluation as $F_{k+1} = \mathcal{T}_q^\pi F_k$ for an arbitrary initial function F_0 over $\mathcal{S} \times \mathcal{A}$. Then, F_k converges to Q_q^π and satisfies the TBE equation (4.7).*

The proof of Theorem 15 relies on the contraction property of $\mathcal{T}_q^\pi$. The contraction property guarantees the sequence of F_k converges to a fixed point F_* of $\mathcal{T}_q^\pi$, i.e., $F_* = \mathcal{T}_q^\pi F_*$ and the fixed point F_* is the same as Q_q^π. The value function evaluated from Tsallis policy evaluation can be employed to update the policy distribution. In the policy improvement step, the policy is updated to maximize

$$\forall s,\, \pi_{k+1}(\cdot|s) = \arg\max_{\pi(\cdot|s)} \mathop{\mathbb{E}}_{a\sim\pi}[Q_q^{\pi_k}(s,a) - \ln_q(\pi(a|s))|s]. \tag{4.9}$$

Theorem 16 (Tsallis Policy Improvement)**.** *For $q > 0$, let π_{k+1} be the updated policy from (4.9) using $Q_q^{\pi_k}$. For all $(s,a) \in \mathcal{S} \times \mathcal{A}$, $Q_q^{\pi_{k+1}}(s,a)$ is greater than or equal to $Q_q^{\pi_k}(s,a)$.*

Theorem 16 tells us that the policy obtained by the maximization (4.9) has performance no worse than the previous policy. From Theorem 15 and 16, it is guaranteed that the Tsallis policy iteration gradually improves its policy as the number of iterations increases and it converges to the optimal solution.

Theorem 17 (Optimality of TPI)**.** *When $q > 0$, define the Tsallis policy iteration as alternatively applying (4.8) and (4.9), then π_k converges to the optimal policy.*

The proof is done by checking if the converged policy satisfies the TBO equation. In the next section, Tsallis policy iteration is extended to a Tsallis actor-critic method which is a practical algorithm to handle continuous state and action spaces in complex environments.

Tsallis Value Iteration

Tsallis value iteration is derived from the optimality condition. From (4.6), the TBO operator is defined by

$$\begin{aligned} \left[\mathcal{T}_q F\right](s,a) &\triangleq \mathop{\mathbb{E}}_{s'\sim P}\left[\mathbf{r}(s,a,s') + \gamma V_F(s) \middle| s,a\right], \\ V_F(s) &\triangleq q\text{-}\max_{a'}\left(F(s,a')\right). \end{aligned} \tag{4.10}$$

Then, Tsallis value iteration (TVI) is defined by repeatedly applying the TBO operator,i.e., $F_{k+1} = \mathcal{T}_q F_k$.

Theorem 18. *For $q > 0$, consider the TBO operator $\mathcal{T}_q$, and define Tsallis value iteration as $F_{k+1} = \mathcal{T}_q F_k$ for an arbitrary initial function F_0 over $\mathcal{S} \times \mathcal{A}$. Then, F_k converges to $Q_q^\star$.*

Similar to Tsallis policy evaluation, the convergence of Tsallis value iteration depends on the contraction property of $\mathcal{T}_q$, which makes F_k converges to a fixed point of $\mathcal{T}_q$. Then, the fixed point can be shown to satisfy the TBO equation.

Performance Error Bounds and q-Scheduling

We provide the performance error bounds of the optimal policy of a Tsallis MDP which can be obtained by TPI or TVI. The error is caused by the regularization term used in Tsallis entropy maximization. We compare the performance between the optimal policy of a Tsallis MDP and that of the original MDP. The performance error bounds are derived as follows.

Theorem 19. *Let $J(\pi)$ be the expected sum of rewards of a given policy π, $\pi^{\star}$ be the optimal policy of an original MDP, and $\pi_q^{\star}$ be the optimal policy of a Tsallis MDP with an entropic index q. Then, the following inequality holds: $J(\pi^{\star}) + (1-\gamma)^{-1}\ln_q(1/|\mathcal{A}|) \leq J(\pi_q^{\star}) \leq J(\pi^{\star})$, where $|\mathcal{A}|$ is the cardinality of $\mathcal{A}$ and $q > 0$.*

The proof of Theorem 19 is included in the supplementary material. Here, we can observe that the performance gap shows a similar property of the TBO equation. We further verify Theorem 19 on a simple grid world problem. We compute the expected sum of rewards of $\pi_q^{\star}$ obtained from TVI by varying q values and compare them to the bounds in Theorem 19. Notice that $\ln_q(1/|\mathcal{A}|) \propto 1/|\mathcal{A}|^{q-1}$ converges to zero as $q \rightarrow \infty$. This fact supports that $\pi_q^{\star}$ converges to the greedy optimal policy in the original Bellman equation when $q \rightarrow \infty$. Inspired by Theorem 19, we develop a scheduled TPI by linearly increasing q_k from zero to infinity during Tsallis policy iteration. From the following theorem, we can guarantee that it converges to the optimal policy of the original MDP.

Theorem 20 (Scheduled TPI). *Let $\mathcal{TPI}_q$ be the Tsallis policy iteration operator with an entropic index q. Assume that a diverging sequence q_k is given, such that $\lim_{k\rightarrow\infty} q_k = \infty$. For given q_k, scheduled TPI is defined as $\mathcal{TPI}_{q_k}$, i.e., $\pi_{k+1} = \mathcal{TPI}_{q_k}(\pi_k)$. Then, $\pi_k \rightarrow \pi^{\star}$ as $k \rightarrow \infty$.*

4.1.3 Tsallis Actor Critic for Model-Free RL

We extend Tsallis policy iteration to a Tsallis actor-critic (TAC) method, which can be applied to a continuous action space. From our theoretical results, existing SG entropy-based methods can be freely extended to utilize a Tsallis entropy by replacing the SG entropy term. In order to verify the pure effect of the Tsallis entropy, we modified the soft actor critic (SAC) method by employing $\ln_q(\pi(a|s))$ instead of $\ln(\pi(a|s))$ and compare to the SAC method.

Similarly to SAC, our algorithm maintains five networks to model a policy π_ϕ, state value V_ψ, target state value V_{ψ^-}, two state action values Q_{θ_1} and Q_{θ_2}. We also utilize a replay buffer $\mathcal{D}$ which stores every interaction data $(s_t, a_t, r_{t+1}, s_{t+1})$. To update state value network V_ψ, we minimize the following loss,

$$J_\psi = \mathop{\mathbb{E}}_{s_t, a_t \sim \mathcal{B}} \left[(y_t - V_\psi(s_t))^2 / 2 \right] \tag{4.11}$$

where $\mathcal{B} \subset \mathcal{D}$ is a mini-batch and y_t is a target value defined as $y_t = Q_{\min}(s_t, a_t) - \alpha \ln_q(\pi_\phi(a_t|s_t)$, and, $Q_{\min}(s_t, a_t) = \min[Q_{\theta_1}(s_t, a_t), Q_{\theta_2}(s_t, a_t)]$. The technique using the minimum state action value between two approximations of Q^π is known to prevent overestimation problem [40] and makes the learning process numerically stable. After updating ψ, ψ^- is updated by an exponential moving average with a ratio τ. For both θ_1 and θ_2, we minimize the following loss function,

$$J_\theta = \mathop{\mathbb{E}}_{b_t \sim \mathcal{B}} \left[(Q_\theta(s_t, a_t) - r_{t+1} - \gamma V_{\psi^-}(s_{t+1}))^2 / 2 \right], \tag{4.12}$$

where b_t is $(s_t, a_t, s_{t+1}, r_{t+1})$. This loss function is induced by the Tsallis policy evaluation step.

When updating an actor network, we minimize a policy improvement objective defined as

$$J_\phi = \mathop{\mathbb{E}}_{s_t \sim \mathcal{B}} \left[\mathop{\mathbb{E}}_{a \sim \pi_\phi} [\alpha \ln_q(\pi_\phi(a|s_t)) - Q_\theta(s_t, a)] \right]. \tag{4.13}$$

Note that a is sampled from π_ϕ not a replay buffer. Since updating J_ϕ requires to compute a stochastic gradient, we use a reparameterization trick similar to [51] instead of a score function estimation. In our implementation, a policy function is defined as a Gaussian distribution defined by a mean μ_ϕ and variance σ_ϕ^2. Consequently, we can rewrite J_ϕ with a reparameterized action and a stochastic gradient is computed as

$$\nabla_\phi J_\phi = \mathop{\mathbb{E}}_{s_t \sim \mathcal{B}} \left[\mathop{\mathbb{E}}_{\epsilon \sim P_\epsilon} \left[\alpha \nabla_\phi \ln_q(\pi_\phi(a|s_t)) - \nabla_\phi Q_\theta(s_t, a) \right] \right],$$

where $a = \mu_\phi + \sigma_\phi \epsilon$ and ϵ is a unit normal noise. Furthermore, we present TAC with Curricular (TAC^2) that gradually increase the entropic index q based on Theorem 20. While it is optimal to search the proper entropic index given an RL problem, the exhaustive search is often impractical due to prohibitive high sample complexity. The entire TAC and TAC^2 algorithms are summarized in Algorithm 3.

4.1.4 Experiments Setup

Simulation Setup

To verify the characteristics and efficiency of our algorithm, we prepare four simulation tests on continuous control problems using the MuJoCo simulator: HalfCheetah-v2, Ant-v2, Pusher-v2, and Humanoid-v2. For each task, a robot with multiple actuated joints is given where the number of joints is different from each task. Then, a state is defined as sensor measurements of actuators and an action is defined as torques. The goal of each task is to control a robot with multiple actuated joints to move forward as fast as possible. More detailed definition can be found in [37].

In the first simulation, to verify the effect of the entropic index q, we conduct

Algorithm 3 Tsallis Actor Critic (TAC)

```
Input: Total time steps $t_{\max}$, Max episode length $l_{\max}$, Memory size $N$, Entropy coefficient $\alpha$, Entropic index $q$ (or schedule), Moving average ratio $\tau$, Environment $env$
Initialize: $\psi, \psi^{-}, \theta_1, \theta_2, \phi$, $\mathcal{D}$ : Queue with size $N$, $t = 0$, $t_e = 0$
while $t \leq t_{\max}$ do
    $a_t \sim \pi_\phi$ and $\mathbf{r}_{t+1}, s_{t+1}, d_{t+1} \sim env$ where $d_{t+1}$ is a terminal signal.
    $\mathcal{D} \leftarrow \mathcal{D} \cup \{(s_t, a_t, \mathbf{r}_{t+1}, s_{t+1}, d_{t+1})\}$
    $t_e = t_e + 1$, $t = t + 1$
    if $d_{t+1} =$ True or $t_e = l_{\max}$ then
        for $i = 1$ to $t_e$ do
            Randomly sample a mini-batch $\mathcal{B}$ from $\mathcal{D}$
            Minimize $J_\psi, J_{\theta_1}, J_{\theta_2}$, and $J_\phi$ using a stochastic gradient descent
            $\psi^{-} \leftarrow (1 - \tau)\psi^{-} + \tau\psi$
        end for
        Reset $env$, $t_e = 0$
        if Schedule of $q$ exists then
            Update $q_t$
        end if
    end if
end while
```

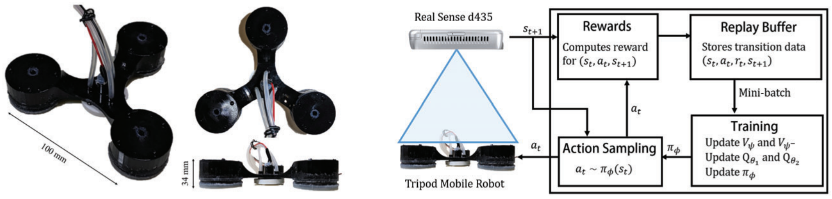
experiments with a wide range of q values from 0.5 to 5.0 and measure the total average returns during the training phase. We only change the entropic index and fix an entropy coefficient α to 0.05 for Humanoid-v2 and 0.2 for other problems. We run entire algorithms with ten different random seeds. Second, to verify the effect of α, we run TAC with different q values (including SAC) for three α values:

$0.2, 0.02$, and 0.002 on the Ant-v2 problem. Third, we test the variant of TAC by linearly scheduling the entropic index. From the results of the first simulations, we observe that there exists a numerically stable region of $1 < q < 2$, which will be explained in Section 4.1.5. We schedule q to linearly increase from 1 to 2 for every 5000 steps and we run TAC with q schedule for three α values: $0.2, 0.02$, and 0.002 on the Ant-v2 problem. Finally, we conduct a compare our algorithm to the existing state-of-the-art on-policy and off-policy actor-critic methods. For on-policy methods, trust region policy optimization (TRPO) [113] and proximal policy optimization (PPO) [116] are compared where a value network is employed for generalized advantage estimation [114]. For off-policy methods, deep deterministic policy gradient (DDPG) [80] and twin delayed DDPG which is called TD3 [40] are compared. We also compare with the soft actor-critic (SAC) method [51] which employs the SG entropy for exploration. Since TAC can be reduced to SAC with $q = 1$ and algorithmic details are the same, we denote TAC with $q = 1$ as SAC. We utilize OpenAI's implementations and extend the SAC algorithm to TAC. To obtain consistent results, we run all algorithms with ten different random seeds. While we compare various existing methods, results of TRPO, PPO, and DDPG are omitted here due to their poor performance and the entire results can be found in the supplementary material. The source code is publicly available[3].

Hardware Platform Setup

To test our algorithm on a soft mobile robot, we use a tripod mobile robot that consisted of three pneumatic soft vibration actuators, a direct current (DC) motor, and an equilateral triangle body plate as shown in Figure 4.1(a). Each ac-

[3] `https://github.com/rllab-snu/tsallis_actor_critic_mujoco`



(a) Tripod Mobile Robot

(b) Training System

Figure 4.1: (a) A soft mobile robot used in experiment. (b) A diagram for training system. The position of robot is measured by using blob detection from a RGB image ofr RealSense d435.

tuator can independently vibrate continuously and robustly regardless of contact with external objects by using the nonlinear stiffness characteristic of hyperelastic material (Eco-flex 30). In addition, the vibration frequency of the actuator can be controlled by the input pressure. In order to control the direction of rotation of the robot, a direct current (DC) motor was installed at the center of the robot combined with a rotating plate. As a result, the mobile robot is capable of making various motions, such as translation and rotation, with a combination of the three vibration modes of the actuator and the rotation of the rotating plate.

Real Robot Experiment Setup

We apply the proposed algorithm to a soft mobile robot and compare the proposed method to SAC with $\alpha = 0.01$ and SAC with automatic entropy adjustment (SAC-AEA) [49] which automatically adjusts α to maintain the entropy to be greater than a predefined threshold δ. In experiment, we heuristically set δ to $-\ln(d)$ as proposed in [49] where d is a dimension of the action space. In [49], since SAC-AEA shows efficient performances for learning quadrupedal locomotion, we try to check whether SAC-AEA can be applied to a soft mobile robot while comparing their performance to the proposed method. We would like to note that TAC^2 only schedule q with fixed α and SAC-AEA only changes α with fixed $q = 1.0$. From this comparison, we can demonstrate which factor is more important to achieve efficient exploration.

In this task, our goal is to train a feedback controller of a soft mobile robot where a controller makes a robot move in a straight line towards a goal position (x_g, y_g) with a heading $\theta_g := \arctan(y_g - y_t, x_g - x_t)$ where x_t, y_t is a current position of the robot. Note that if a robot's heading is aligned to its moving direction, then, $\theta_g = \theta_t$.

The robot has three soft membrane vibration actuators and one motor for controlling the angular momentum of the robot. Hence, an action is defined as a four dimensional vector as $a_t = (p_1, p_2, p_3, \delta\Omega)_t$, where p_i is an input pressure of each vibration actuator and $\delta\Omega$ is the change in the motor speed. Note that, if we directly change the motor signal, it may generate unstable motion and inconsistent movements due to the delay of the motor. Hence, by controlling a difference of the motor signal, we can generate a smooth change of motor speed.

Then, a state of a robot is defined as $s_t := (\Delta\theta_t, d_t, \Omega_t)$, where $\Delta\theta_t := \theta_g - \theta_t$ is a difference between heading and goal direction, $d_t := \sqrt{(x_t - x_g)^2 + (y_t - y_g)^2}$, is the Euclidean distance to the desired position, and Ω_t is the current motor speed.

A reward function $r(s_t)$ assigns a higher score as a control minimizes the gap between robot's current state and desired state: $r(s_t) := -d_t - |\Delta\theta_t| + 2$, which is a decreasing function of d_t and $\Delta\theta_t$ where 2 is added to give a positive reward near the goal position. γ is set to 0.99. The entire training system is illustrated in Figure 4.1(b).

For a fair comparison, we evaluate each algorithm every 500 steps. In evaluation, we control a robot using only the mean value of the trained policy without sampling. We run all algorithms with 2500 steps for five trials.

4.1.5 Experimental Results

Effect of α in Multi Armed Bandit Problem

We compare the effect of α and effect of q values as shown in Figure 4.2. We compute the $\pi_q^\star$ on a simple MAB problem with the reward function shown in Figure 4.2, while changing α and q values. We can see that all policy with different q values converge to greedy policy when $\alpha \rightarrow 0$. However, the tendency of convergence is different depending on q values. First, the supporting set of $\pi_q^\star$ with entropy coefficient α is defined by

$$1 + (q-1)\left(\frac{\mathbf{r}(a)}{\alpha q} - \psi_q\left(\frac{\mathbf{r}}{\alpha q}\right)\right) > 0.$$

Since we only consider $q \geq 1$ and $(q-1)$ is positive, the supporting set becomes bigger as α goes to infinity. Thus, for fixed q, large α makes $\pi_q^\star$ more uniform but, from the supporting set condition, the probability mass are distributed over restrict elements. For example, when $q = 2.0$, the probability mass are only distributed on four elements while α vareis from 0.1 to 2.0. On the contrary, when $q = 1.0$ the probability mass are distributed over the entire action space while α vareis from 0.1 to 2.0. We would like to note that q value controls the supporting set and α controls the distribution over the supporting set.

Example of Bounds for q-Maximum

From theorem 13, we have the bounds for q-maximum as follows,

$$\max_x(f(x)) \leq q\text{-}\max_x\left(f(x)\right) \leq \max_x(f(x)) - \ln_q\left(1/|\mathcal{X}|\right)$$

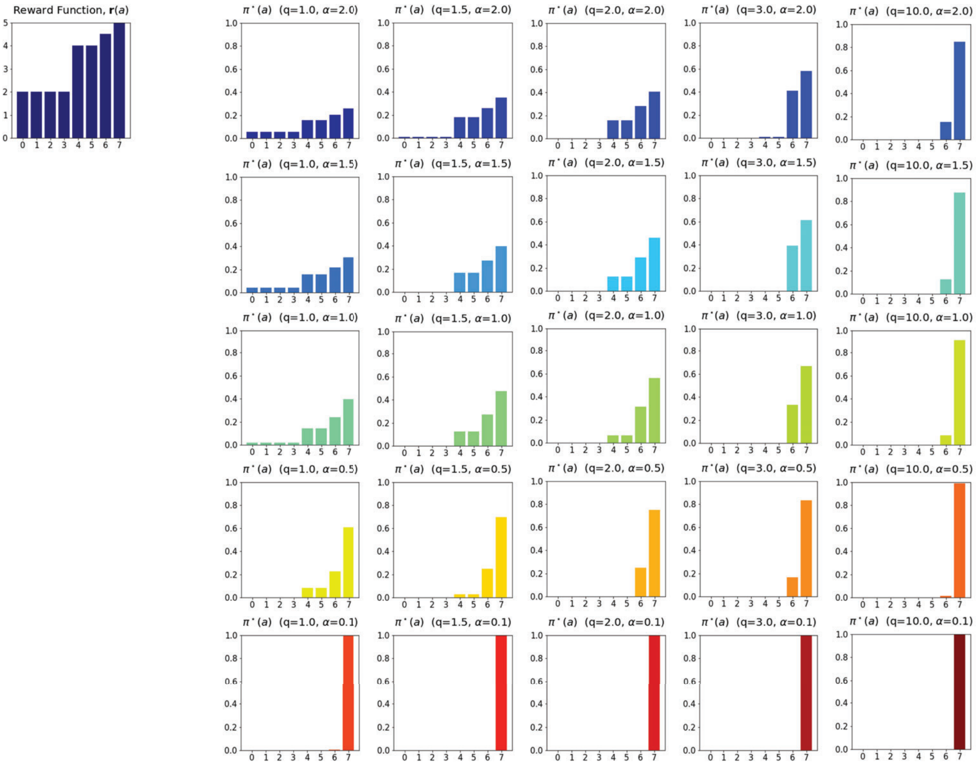


Figure 4.2: Example of $\pi_q^\star(a)$ with various coefficients of entropy α varying from 2.0 to 0.1 and *entropic indices* varying from 1.0 to 10.0, respectively.

In example, we set $\mathcal{X} = \{0, 1\}$ and $f(x)$ is deinfed as $f(0) = 0, f(1) = c$. We see the tendecy of q-maximum when c varies from -2 to 2. Then, $\max_x(f(x))$ becomes $\max([c, 0])$ and we compute the q-$\max([c, 0])$ using numerical solver. Since $\mathcal{X}$ has two elements, the upper bound is $\max([c, 0]) - \ln_q(1/2)$.

Examples of q-maximum with different q values are shown in Figure 4.3. It can be observed that, as q increases, the bounds become tighter. Note that the gap becomes largest when $q = 1$. This gap sometimes leads to overestimation error when we use q-maximum to compute the target value of value networks.

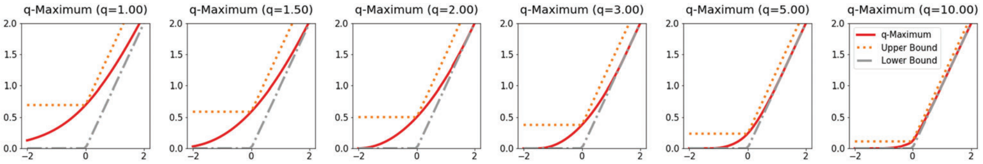
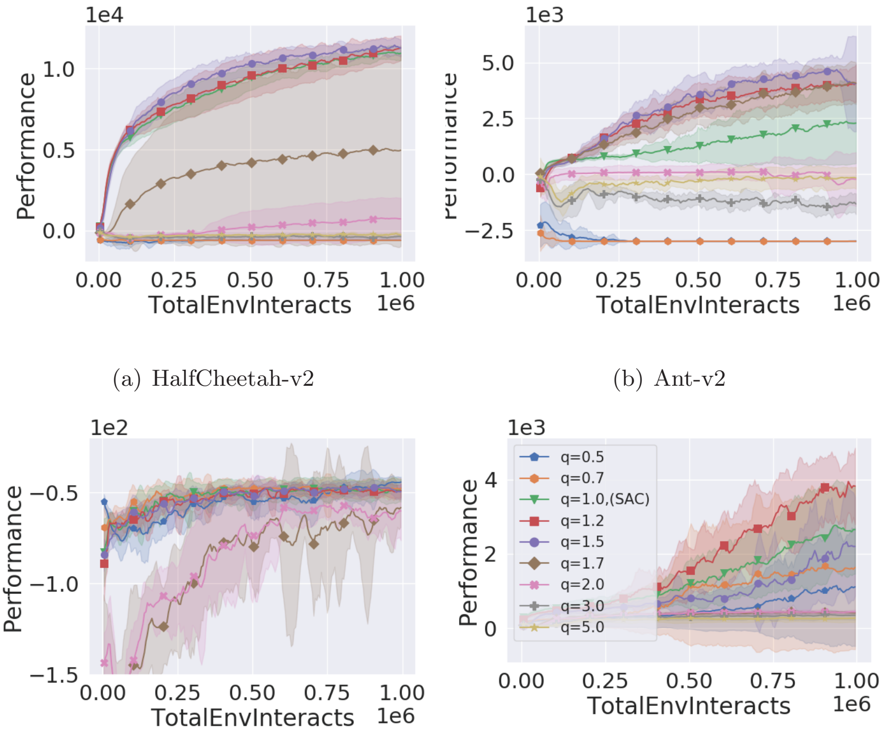


Figure 4.3: Example of q-maximum operator with different q values. The figures show q-max$([c, 0])$ over $c \in [-2, 2]$. The bounds are computed by Theorem 13

Effect of Entropic Index q

The results are shown in Figure 4.4. We realize that the proposed method performs better when $1 \leq q < 2$ than when $0 < q < 1$ and $q \geq 2$, in terms of stable convergence and final total average returns. Using $0 < q < 1$ generally shows poor performance since it hinders exploitation more strongly than the SG entropy. For $1 \leq q < 2$, the Tsallis entropy penalizes less the greediness of a policy compared to the SG entropy (or $q = 1$). From a reparameterization trick, the gradient of the Tsallis entropy becomes $\mathbb{E}_{a\sim\pi_\phi}[\pi_\phi(a|s)^{q-2}\nabla_\phi\pi_\phi(a|s)]$. For $q \geq 2$, the gradient is proportional to $\pi_\phi(a|s)$, thus, if $\pi_\phi(a|s)$ is small, then, the gradient becomes smaller and it leads to early convergence to a locally optimal policy. For $0 < q < 2$, the gradient is proportional to $1/\pi_\phi(a|s)$, thus, if $\pi_\phi(a|s)$ is small, the gradient becomes larger, which encourages exploration of the action with a small probability. For $0 < q < 1$, since $\pi_\phi(a|s)^{q-2}$ is more amplified than when $1 \leq q < 2$, the penalty of greediness is stronger than when $1 \leq q < 2$. Thus, when $0 < q < 1$, it penalizes the exploitation of TAC more and hinders the convergence to an optimal policy. In this regard, we can see TAC with $1 \leq q < 2$ outperforms TAC with $q \geq 2$. Furthermore, in HalfCheetah-v2 and Ant-v2, TAC with $q = 1.5$ shows the best performance and, in Humanoid-v2, TAC with $q = 1.2$ shows the best performance. Furthermore, in Pusher-v2, the final total average returns of



(a) HalfCheetah-v2

(b) Ant-v2

(c) Pusher-v2

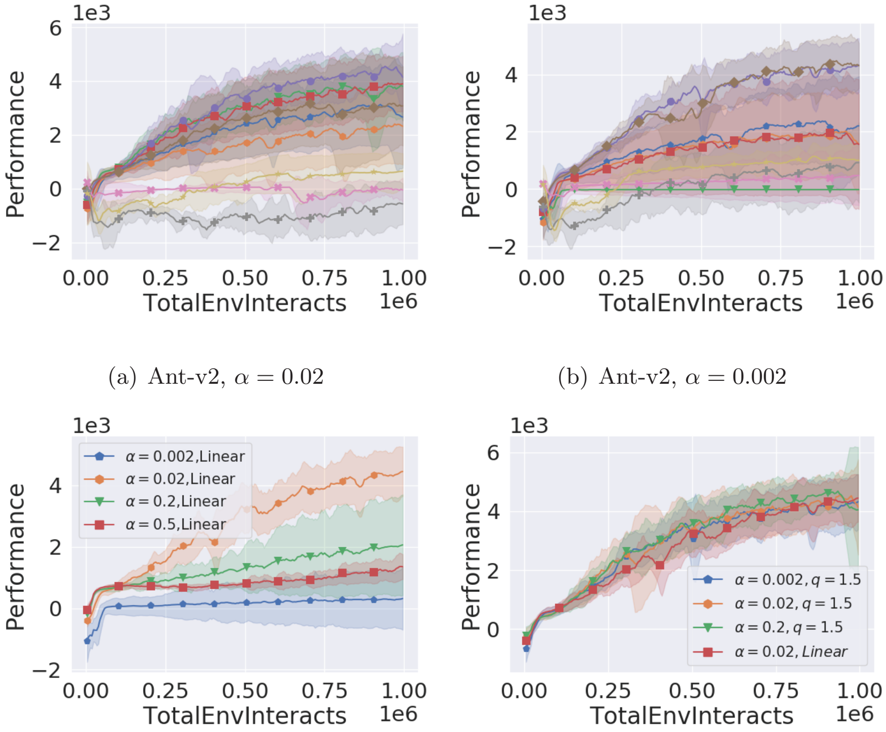
(d) Humanoid-v2

Figure 4.4: Average training returns of TAC with different q values on four MuJoCo tasks. A solid line is the average return over ten trials and the shade area shows one variance.

all settings are similar, but TAC with $q = 1.2$ shows slightly faster convergence. We believe that these results empirically show that there exists an appropriate q value between one and two depending on the environment while $q \geq 2$ has a negative effect on exploration.

Effect of Coefficient α

As shown in Figure 4.4(b),4.5(a) and 4.5(b). For all α values, $q = 1.5$ (purple circle line) always shows the fastest convergence and achieves the best performance



(a) Ant-v2, $\alpha = 0.02$

(b) Ant-v2, $\alpha = 0.002$

(c) Ant-v2, Scheduled q_k

(d) Ant-v2, Comparison

Figure 4.5: (a), (b) Average returns of different $\alpha = \{0.02, 0.002\}$ and different q. (a) and (b) share the legend with Figure 4.4(d). (c) Average returns of scheduling q_k with different α. Linear indicates linear curriculum of q_k. (d) Comparison of all variants of TAC.

among tested q values This result tells us that TAC with the best q value is robust to change α. For $q = 1.2$ (or $q = 1.7$), the average return of TAC with $q = 1.2$ (or $q = 1.7$) is sensitive to α, respectively, where $q = 1.7$ has the best average return at $\alpha = 0.002$, and $q = 1.2$ has the best value at $\alpha = 0.02$. However, TAC with $q = 1.5$ consistently outperforms other entropic indices while α is changed.

Curriculum on Entropic Index q

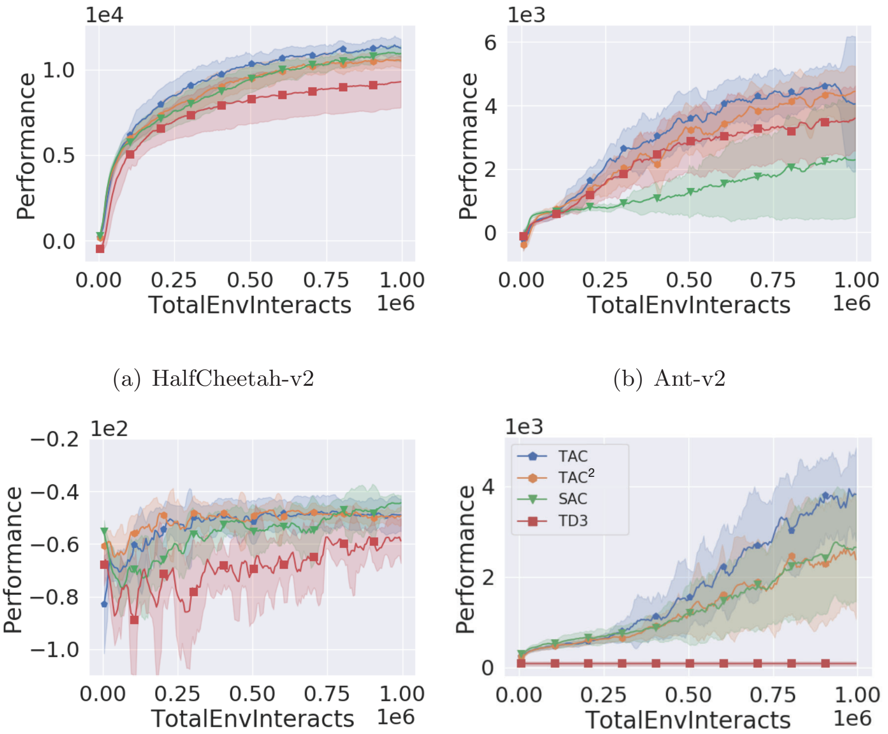
Figure 4.5(c) shows the performance of TAC^2 with different α and Figure 4.5(d) illustrates the comparison to TAC with fixed q. From this observation, it is shown that TAC^2 achieves a similar performance of the best q value without using a brute force search.

Comparative Evaluation

Figure 4.6 shows the total average returns of TAC and other compared methods. We use the best combination of q and α from the previous experiments for TAC with $q \neq 1$ and SAC (TAC with $q = 1$). SAC and TAC use the same architectures for actor and critic networks. TAC and TAC^2 indicates TAC with the fixed best q and linearly scheduled q, respectively. First, TAC with a proper q outperforms all existing methods in all environments. Furthermore, TAC achieves better performance with a smaller number of samples than SAC and TD3 in all problems. Especially, in Ant-v2, TAC improves the performance from SAC by changing $q = 1.5$. Furthermore, in Humanoid-v2 which has the largest action space (17D), TAC with $q = 1.2$ outperforms all the other methods. Finally, TAC^2 consistently shows similar performances to TAC, except Humanoid-v2.

Real Robot Experiment

Figure 4.7 shows the results of compared algorithms including the proposed method. TAC^2 shows the best performance in terms of the convergence speed and the sum of rewards compared to other algorithms. In particular, the policy trained by TAC^2 could reach any goal point with only about 1500 steps ($\approx$30 minutes) of training. Furthermore, TAC with $q = 1.5$ shows the second-best performance.



(a) HalfCheetah-v2

(b) Ant-v2

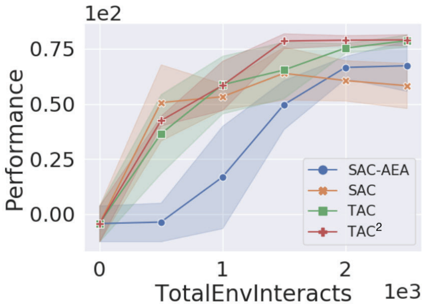
(c) Pusher-v2

(d) Humanoid-v2

Figure 4.6: Comparison to existing actor-critic methods on four MuJoCo tasks. SAC (red square line) is the same as TAC with $q = 1$, TAC and TAC^2 indicates TAC with fixed $q \neq 1$ and scheduled q, respectively.

For SAC and SAC-AEA, while SAC-AEA shows slower convergence than SAC due to the constraint to keep the entropy of the policy above the threshold, it achieves higher performance than SAC at the end of the training. This result demonstrates that maintaining the entropy of the policy helps exploration and leads to better final performance, however, it hampers the exploitation.

While both TAC^2 and SAC-AEA control the exploration-exploitation trade-off by scheduling the level of regularization, the empirical result shows that scheduling q instead of adjusting α shows better performance in terms of both con-



Alg.	Fin. Ret.
TAC	75.9 (5.6)
TAC^2	78.7 (2.4)
SAC	67.3 (10.2)
SAC-AEA	58.3 (9.1)

(a) Evaluation (b) Final Performance

Figure 4.7: Comparison to existing actor-critic methods on training a Tripod mobile robot. (a) Average returns over five trials. (b) Final average performance. The number in parentheses is a standard deviation.

vergence speed and final average return. While adjusting α in SAC-AEA only rescales the magnitude of the gradient of the entropy, scheduling q can change both the scale and direction of the gradient of the entropy, similarly to the results in Section 4.1.5. Specifically, in TAC^2, the regularization effect is gradually reduced as the entropic index q increases while SAC-AEA keeps the level of the Shannon entropy. Hence, scheduling q helps exploitation at the end of the training. Thus, TAC^2 shows not only the highest final average performance but also a much smaller variance than other algorithms, which is a highly preferred feature for training a soft mobile robot. Especially, a low variance of the final performance supports that TAC^2 successfully overcome the unknown stochasticity in the dynamic model of the soft mobile robot. Consequently, we can conclude that TAC^2 efficiently learns a feedback controller of a soft mobile robot and achieves the best performance with the minimum interactions.

4.1.6 Summary

We have proposed a unified framework which allows using a class of different Tsalli entropies in RL problems, which we call Tsallis MDPs, and its application to soft robotics. We first provide the full theoretical analysis about Tsallis MDPs including guarantees of convergence, optimality, and performance error bounds. and have extended it to the Tsallis actor-critic (TAC) method to handle a continuous state-action space. It has been observed that there exists a suitable entropic index for each different RL problem and TAC with the optimal entropic index outperforms existing actor-critic methods. However, since finding an entropic index with the brute force search is a demanding task, we have also present TAC^2 that gradually increases the entropic index and empirically show that it achieves comparable performances with TAC with the optimal entropic index found from an exhaustive search in simulation environments. We have applied TAC^2 on real-world problems of learning a feedback controller for soft mobile robots and demonstrated that TAC^2 shows more efficient exploration tendency than adjusting the regularization coefficient.

4.2 Efficient Exploration for Robotic Grasping

In this section, we apply Shannon-Gibbs entropy-based exploration method to learn to grasp an unseen object. Recent advances of deep learning have enabled to efficiently employ high dimensional observations such as depth image [84], or point clouds [89] in many robotics applications. In particular, a convolutional neural network (CNN) has shown powerful performances in many image-based data-driven methods [99, 72, 100, 42, 48, 148, 149, 62, 110, 27, 18, 145, 89]. For a robotic grasping problem, predicting a grasp pose of a given object from

RGB or depth images using CNNs has been widely investigated and shown high grasp performance where a neural network is utilized to predict a grasp success probability of given depth images of an object and corresponding grasp poses [84] or to generate high quality grasp poses from an image of an object [89].

While a CNN has a large capacity to learn high dimensional data, it often suffers from the over-fitting problem when the number of training data is small which leads to poor prediction results for unseen data. Hence, most existing methods employing CNNs have focused on handling the lack of training data [83, 84, 82]. In [83], training data for the neural network are generated by mesh data and dynamic simulators where a depth image of objects are synthesized and corresponding grasp poses are generated geometrically from the mesh data. Using these data set, [84] successfully trained a neural network to predict a grasp pose given a depth image and empirically shows that the trained network can be applied to real-world grasping. [82] extends [84] to a real-world bin picking problem, which is a sequential grasping problem, by augmenting real-world grasping data.

Using simulated data to train the neural network, however, has the limitation in that there exists the discrepancy between simulations and real-world environment as mentioned in [58, 132, 148, 22, 59, 145]. In particular, a synthesized depth image has a different visual properties from that of real-world. Furthermore, when it comes to dynamics, contact simulations may be inaccurate and not similar to real-world phenomena. To handle this discrepancy, [58, 132, 148, 22, 59, 145] have incorporated domain adaptation and domain randomization techniques which diversify the parameters of simulations to cover various types of dynamic environments when training data are collected. While diversifying dynamic property of simulation can alleviate the lack of data, there still exist significant differences between real-world and simulation. In particular, the network trained in simulations may not be applied to unseen objects whose geometric property such as curvatures is significantly different from the simulated data. To reduce the gap, in general, online learning methods have been widely used for a robot to adapt unexpected situations by autonomously exploring an environment. In robotic grasping problems, to learn to grasp unseen objects, the ability to discover possible grasps is required. To this end, we utilize an online learning framework to the robotic grasping problem.

In this section, we propose a no regret Shannon entropy regularized neural contextual bandit algorithm for learning to grasp unknown objects where the Shannon entropy is employed to encourage explorations. Our strategy is to sample a grasp pose from a stochastic policy which is a conditional distribution of a depth image from a given object. The probability selecting a grasp pose is exponentially weighted by the estimated success probability, or also called grasp quality where exponential weighting is called a softmax distribution which is induced by the Shannon entropy regularization. Since the stochastic policy is employed, the proposed method randomly explores various grasp poses, but the grasp poses which have high estimated qualities are explored more. We also prove that the proposed method converges into optimal policy efficiently fast, which is called no regret property.

In experiments, we verify the efficiency of the proposed method compared to other exploration methods.

4.2.1 Related Work

Owed by recent advances of deep learning, many existing robotic grasp methods have been developed based on data-driven approaches [99, 72, 100, 42, 48, 148, 149, 62, 110, 27, 18, 145, 89]. In general, these methods train two types of network:

grasp quality network and grasp proposal network. The grasp quality network predicts the success probability of given grasp pose and information about an object to be grasped where object information is generally given by a depth image, RGB image, or both. The grasp proposal network generates a grasp pose of end-effector based on given inputs such as RGBD image. Most existing methods focused on how to generate training data for a deep neural network and how to generate simulated data for robust transfer and generalization in real-world. While [83, 82, 84, 99, 72, 100, 42, 48, 148, 149, 62, 110, 27, 18, 145, 89] have been shown powerful results, however, learning based methods have the limitation in that a grasp performance can degenerate for unseen objects which are not included in simulations and training data. To handle this issue, online learning approaches, including ours, for robotic grasping have been investigated where a robot is trained with sequentially generated data during the test phase to adapt unseen objects.

Robotic Grasping with Deep Learning

Conventional methods for robotic grasping are often solved analytically. For example, 3D mesh data of objects is often used to compute grasp pose. [147] directly computes a grasp pose from point cloud data. However, such geometry based methods are not applicable for unknown objects which have no 3D mesh data and are hard to generalize various types of objects.

Due to this drawback, learning based approaches have been developed [83, 82, 84, 89, 78] where grasp poses are often predicted by a deep neural network instead of computing grasp poses from geometric information. These methods often utilize a convolutional neural network (CNN) which shows high performance for image data [83, 82, 84]. The CNN is generally used to model a grasping quality network whose inputs are RGB or depth image and an arbitrary grasp pose and outputs are the success probability of a given grasp. However, to train the CNN for predicting the success probability of a grasp pose, lots of training data are required. In [84], training data are collected in simulation using given 3D mesh data. Thus, the trained grasp model can be generalized into unseen objects while there still exists the gap between simulation and real-world data. In [82], pretrained model of [84] is used to gather real-world training data. Unlikely to [83, 82, 84], [89, 78] predict grasp poses from point cloud data. In [89], point cloud data and corresponding grasp poses are learned with conditional variational auto-encoder where, in testing time, grasp candidates are generated using the decoder. [78] trained the grasp quality network whose inputs are local point clouds where candidates of grasp poses are generated using [129, 47] and their qualities measured by the quality network.

While [83, 82, 84, 89, 78] have demonstrated that the deep neural network trained with large grasping data can generalize predicting grasp poses to unseen objects, using deep learning for the real-world grasping task has the limitation in that training the deep neural network accurately requires a large amount of training data and collecting a large number of grasping data in real-world is a demanding task.

Sim to Real Transfer

To alleviate the lack of real-world grasping data, [58, 132, 148, 22, 59, 145] often utilizes a domain randomization technique generating simulated grasping data while changing various environmental parameters such as light condition, mass, inertia, friction coefficient, or background images. [148, 59] proposed a domain adaptation method which makes a grasp network overcome the discrepancy between real-world data and simulated data. [58, 132] also collected simulated data by randomizing visual effects and dynamic properties and combine them to train the grasp proposal network. In [132], the method randomizing objects has been proposed where simulated objects are randomly generated by combining arbitrary selected meshes. [132] showed that the grasp proposal network trained with only simulated data can effectively grasp real-world objects. In [58], randomization is applied to background images of a simulator where the trained grasp proposal network showed robust performance in real-world grasping test.

Online Learning in Robotics

While existing learning based grasping methods including sim-to-real methods handle the discrepancy. the lack of training data for unseen objects which have never been simulated is still an issue. In particular, since the dynamic contact simulation has an error compared to real-world contacts, the training data collected by simulations may be imperfect in that they do not reflect real-world contacts.

To handle this issue, real-world data augmentation is essential to reduce the gap between simulation and real-world dynamics. Online learning method has been widely used in robot learning problems from learning a dynamic model [70, 17] to finetuning gains of a PID controller [153]. However, it is hardly found to apply an online learning approach to grasping problems.

In this section, we propose an online learning algorithm using a neural network to estimate rewards function and apply the proposed method to a robotic grasping problem. The core idea of the proposed method is to utilize Shannon entropy to generate a stochastic grasp proposal policy. The stochastic policy induced by Shannon entropy regularization encourages diverse grasp poses, but, the grasp pose whose quality is expected to be high will be tried more often. By using this property, we can balance the trade-off between exploration and exploitation. More detail formulation will be explained in the following section.

Bandit Algorithm with Shannon Entropy Regularization

In [13], Shannon entropy regularization is applied to contextual bandit problems. [13] proposed EXP4 which stands for exponential weighting for exploration and exploitation with experts. For each round, given a context s_t, the experts estimate rewards using a linear model and compute the policy as a softmax distribution of the estimated reward of each action. While [13] does not mention that a softmax distribution is induced by Shannon entropy regularization, using the softmax distribution of the estimated rewards can be interpreted as using Shannon entropy regularization. In particular, a softmax distribution in [13] is computed as

$$\pi_t(s_t) := \arg\max_{\pi\in\Pi} \mathbb{E}_{a\sim\pi}\left[\hat{r}_a(s_t;\theta_{t-1})\right] + \alpha S(\pi) \tag{4.14}$$

where $\hat{r}_a(s_t;\theta_{t-1})$ is the reward estimation of action a predicted by a linear estimation [13] and α is a regularization coefficient. In [13], it was proven that EXP4 is no regret which means the cumulative regret of EXP4 grows sub-linearly, thus, it will converge to an optimal policy. While EXP4 shows efficient performances in several problems [13], the linear model assumption of EXP4 makes it difficult to be applied to high dimensional contextual bandit problems such as the problem whose context is given as an image. However, we utilize a convolutional neural network (CNN) to predict rewards given context image in grasping problem. Furthermore, we also prove that, when the CNN is used as a predictor, no regret property of the proposed method also holds.

4.2.2 Shannon Entropy Regularized Neural Contextual Bandit Algorithm

In this section, we propose a Shannon etnropy regularized neural contextual bandit algorithm (SERN) by utilizing an artificial neural network as a reward estimator and exploring various actions using Shannon entropy. The main difference of the proposed method from existing regularized bandit algorithms is that we does not assume unbiased estimation such as a linear model or Gaussian process regression. Furthermore, we analyze the upper bound of the cumulative regret of SERN and show that it is no regret. Thus, the proposed method enables to use a neural network which has the large capacity and shows a powerful performance for high dimensional data while maintaining the no regret property.

For each round, SERN estimates rewards $\hat{r}_a(s_t;\theta)$ for given context s_t where θ is a parameter of a neural network, and computes a policy π_t and samples an action $a_t \sim \pi_t$ where the policy is computed as Equation (4.14). After choosing a_t, random reward $\mathbf{R}_t$ is obtained, then, we collect context, action and reward pair $(s_t, a_t, \mathbf{R}_t)$ and update the parameter θ_{t-1} to θ_t based on collected data using a stochastic gradient descent to minimize the estimation error. Then, the environment gives the next context s_{t+1}.

4.2.3 Theoretical Analysis

In this section, we provide theoretical analysis of our algorithm whose regret grows sub linearly and, thus, it has no regret property. To the best of our knowledge, this is the first analysis for the bandit algorithm to utilize a deep neural network as a reward estimation. Before starting analysis, we introduce some assumptions for the reward function, the neural network and its error bounds.

Algorithm 4 Shannon Entropy Regularized Neural Contextual Bandit Algorithm (SERN)

```
Initialize $\theta_0$ and $\mathcal{D} = \emptyset$
for $t = 1, \cdots, T$ do
    A context $s_t$ is given and agent chooses $a_t \sim \pi_t$
    where $\pi_t := \arg\max_\pi \{\mathbb{E}_{a\sim\pi}[\hat{r}_a(s_t; \theta_{t-1})] + \alpha S(\pi)\}$
    Agent gets a reward $\mathbf{R}_t$ and stores $(s_t, a_t, \mathbf{R}_t)$ into $\mathcal{D}$
    $\theta_{t-1}$ is updated using a stochastic gradient descent method.
end for
```

Assumptions

We first propose some assumptions required to analyze the cumulative regret of SERN.

Assumption 1 (Separable Reward Structure)**.** *Define the reward gap as* $\Delta_a(s) = \max_{a'} r_{a'}(s) - r_a(s)$ *for given* s*. Note that* $\min_a \Delta_a(s) = 0$ *at the best arm* $a^\star = \arg\max_{a'} r_{a'}(s)$*. Let the second minimum reward gap be* $\Delta_2(s) = \min_{a\neq a^\star} \Delta_a(s)$*. Then, we assume that* $\Delta_2(s) > 0$ *for all* s *and define* $\Delta_2 := \min_s \Delta_2(s)$

Assumption 2 (Rewards Estimation)**.** *For each arm* a*, we have the reward estimator* $\hat{r}_a(s; \theta_{n_a})$ *where* n_a *is the number of training data for* $\hat{r}_a$ *collected by pulling* a *and* θ *is the parameter of* $\hat{r}_a$*. We assume that the parameter has the least error, i.e.,* $\theta_{n_a} = \arg\max_\theta \mathbb{E}_{s_{1:n_a}}\left[\sum_{i=1}^{n_a} |r_a(s_i) - \hat{r}_a(s_i; \theta)|\right]$.

Assumption 3 (Error Bound or Sample Complexity)**.** *We assume the upper bound of the error of a reward estimation as follows:*

$$\forall s \in \mathcal{S} \;\; |r_a(s) - \hat{r}_a(s; \theta_{n_a})| < \frac{\beta}{\sqrt{n_a + 1}}$$

where β *is a positive constant depending on a estimation model and a learning* *algorithm. When n_a data is given, the expected error decreases proportionally to the square root of the number of data.*

Assumption 2 and 3 generally hold for a deep neural network. For Assumption 2, we believe that the best parameter for given training data can be achieved by using general optimization techniques for the deep neural network, such as . For Assumption 3, in [15], Barron showed that the error bound of the neural network follows $O(1/\sqrt{n})$ and in [123], Suzuki showed that it is bounded by $O(\log_+(\sqrt{n})/n) \leq O(1/\sqrt{n})$. Thus, our assumptions generally hold.

The proof strategy of no regret property consists of two parts. We first show that our algorithm explores every arms infinitely many. Then, we prove that infinite explorations eventually reduce the estimation error small enough and, then, the best arm can be verified. While the proposed method explore every arms infinitely, the ratio of choosing each arm is proportional to its estimated rewards. Hence, we can achieve the sub-linearly growth of $\mathcal{R}_T$, which is no regret. Note that the detail proofs are omitted here and can be found in Appendix.

Infinite Exploration

Let $N_a(t)$ is a random variable indicating how many times an arm a is selected during t rounds. In this section, we prove that the expectation of $N_a(t)$ diverges as t goes to infinity. Furthermore, since the expectation of $N_a(t)$ diverges, the event that $N_a(t)$ is bounded, or, a is finitely many explored occurs with low probability and, in addition, such event never happens when t goes to infinity.

Theorem 21. *Then, for any arm a, the expected count has the following lower bound, $\mathbb{E}\left[N_a(t)\right] \geq ct$ where $c = \frac{1}{K}\exp(-\frac{1}{\alpha})$.*

Theorem 21 tells us that the lower bounds of $N_a(i)$ linearly grows. Since $\lim_{t\to\infty} ct = \infty$, the expectation of $N_a(i)$ goes to infinity. Thus, the proposed method explores every arms infinitely many. To prove Theorem 21, the lower bound of the probability choosing a for every iteration is required.

Lemma 6. *The policy of SERN has a constant lower bound greater than zero, i.e.,* $[\pi_t]_a \geq c > 0$*, where* $c = \frac{1}{K}\exp(-\frac{1}{\alpha})$.

The proofs can be found in Appendix. By using this lemma, we have the lower bound as follows: $\mathbb{E}\left[N_a(t)\right] = \mathbb{E}\left[\sum_{t=1}^{T} \mathbb{I}(a_t = a)\right] = \sum_{t=1}^{T} \pi_t > ct$. From this fact, it can be observed that, for any arm, the expectation of its counts diverges. In other words, every arms are explored infintiely many times. Using Lemma 6 and Theorem 21, we can derive the upper bound of the tail probability of $N_a(t)$.

Theorem 22. *For any arm* a*, let* $N'_t = N_a(t) - ct$*. Then,* N'_t *is sub-Martingale and, from this fact, the following inequality holds, for any* $\delta > 0$*,*

$$\mathbb{P}(N_a(t) < ct - \delta) \leq \exp\left(-\frac{\delta^2}{8t}\right).$$

The proof can be found in Appendix. This theorem tells us that the probability that random variable $N_a(t)$ is below the expected lower bound has an exponential upper bound with respect to its deviation δ. Using this upper bound, we can control the error term $\beta\sqrt{1/(n+1)}$ of the neural network.

Upper Bounds for Expected Cumulative Regret

Now, we prove the no regret property of SERN. We first derive general upper bound of the cumulative regret and derive more specific bounds by controlling α. Then, finally we show that the proposed method is no regret.

Theorem 23. *For* $\alpha > 0$ *and* $1 > q > 0$*, the expected cumulative regret of SERN* *is bounded as*

$$\begin{aligned}\mathcal{R}_T \leq & \beta \sum_{t=1}^{T} \mathop{\mathbb{E}}_{s_{1:t}, a_{1:t}} \left[\frac{1}{\sqrt{(N_{a^\star}(t-1)+1)}} \right] \\ & + \beta \sum_{t=1}^{T} \mathop{\mathbb{E}}_{s_{1:t}, a_{1:t}} \left[\frac{1}{\sqrt{(N_{a_t}(t-1)+1)}} \right] \\ & + \sum_{t=1}^{T} \mathbb{P}(a^\star \neq \hat{a}^\star_{t-1}) + \alpha \ln(K) T,\end{aligned}$$

where $K = |\mathcal{A}|$, $a^\star = \arg\max_a \mathbb{E}_s \left[r_a(s) \right]$, *and* $\hat{a}^\star_t = \arg\max_a \mathbb{E}_s \left[\hat{r}_a(s; \theta_t) \right]$.

The first term comes from the estimation error of the neural network, the second term comes from the failure probability of the neural network for discriminating the best arm, and the last term indicates the regret induced by Tsallis entropy regularization. Before deriving detail upper bounds, we would like to give some intuition of proof strategies for each term. The first term will be bounded by using Theorem 21 and 22. Furthermore, we prove that the second term has a constant bound using Assumption 1. Note that we assume that there always exist a positive gap Δ between the optimal and sub-optimal arms. Thus, if our estimation error becomes below Δ, then, after that point, we can discriminate the best arm from other sub-optimal arms. Finally, the third term will be bounded by controlling α. The entire bound can be derived as follows.

Theorem 24. *Let* $\alpha = \frac{1}{\ln(T^p)}$. *Then, the expected cumulative regret of SERN is bounded as*

$$\begin{aligned}\mathcal{R}_T \leq & C_0 T^{\frac{3p+1}{2}} + C_1 \left(1 - \exp\left(-d_1 T^{-2p}\right)\right)^{-1} \\ & + C_2 \left(1 - \exp\left(-d_2 T^{-2p}\right)\right)^{-1} + \ln(K) T \left(\ln(T^p)\right)^{-1},\end{aligned}$$

where $C_0 = 2^{\frac{7}{2}} K^{\frac{3}{2}} \beta$, $C_1 = 2\beta K$, $C_2 = 2(K-1)\exp((\beta/\Delta_2)^2 - 1/4)$, $d_1 = 1/(32K^2)$, *and* $d_2 = 1/(8K^2)$.

By using Theorem 24, we can show no regret property as follows.

Theorem 25. *For $1/3 > p > 0$, if the number of rounds, T, goes to infinity, then, time-averaged regret converges to zero:* $\lim_{T\to\infty} \frac{\mathcal{R}_T}{T} = 0$.

Theorem 25 tells us the proposed method eventually find the best arm for given context. Entire proof can be found in Appendix. Here, we provide a proof sketch. From Theorem 24, we have the upper bound of $\mathcal{R}_T$ which consists of four parts. First, since the first term and forth term follow $O\left(T^{\frac{3p+1}{2}}\right)$ and $O\left(T(\ln(T^p))^{-1}\right)$, respectively, they increase sub-linearly and, hence, $\frac{O\left(T^{\frac{3p+1}{2}}\right)}{T}$ and $\frac{O\left(T(\ln(T^p))^{-1}\right)}{T}$ converge to zero. Finally, nontrivial parts are the second and third term. Note that the limit of these terms follows $\lim_{x\to\infty}(x(1-\exp(-ax^{-b})))^{-1}$ with $a > 0$ and $b < 1$. Thus, the proof can be finished by showing that

$$\begin{aligned} \lim_{x\to\infty} \frac{1/x}{(1-\exp(-ax^{-b})))} &= \lim_{x\to\infty} \frac{1/x^2}{-\exp(-ax^{-b})\cdot abx^{-b-1}} \\ &= \lim_{x\to\infty} \frac{x^{b-1}}{-ab\exp(-ax^{-b})} = 0 \end{aligned} \tag{4.15}$$

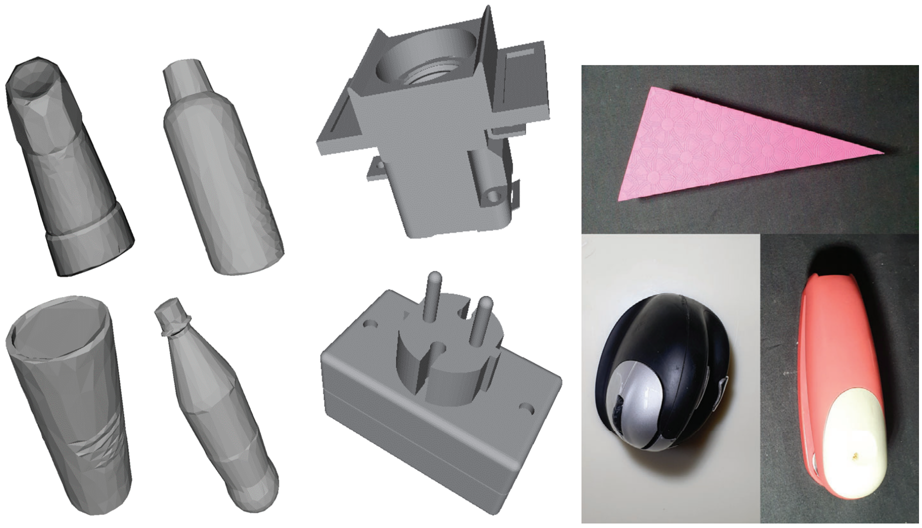
and it implies that the time average of third term converges to zero. The entire proof can be found in Appendix.

4.2.4 Experimental Results

To verify our theorems and effectiveness of the proposed exploration method, we conduct both dynamic simulation and the real-world experiments.

Setup

In the dynamic simulation, we compare three exploration methods including ours. First, a greedy method is compared as a baseline model which simply tries the best grasp whose estimated grasp quality is the maximum. Second, we compare



(a) KIT [63] (b) T-Less [56] (c) Real Objects

Figure 4.8: Objects

a ϵ-greedy method which selects the best grasp pose with probability $1 - \epsilon$ and chooses a uniformly random grasp with probability ϵ. The ϵ-greedy method has been widely used in many existing methods [refs]. For ϵ-greedy method, we set ϵ to be 0.1. Finally, we compare the SERN with $\alpha = 0.05$. ϵ and α are selected by the brute force search.

Furthermore, we employ the grasp quality network and training dataset of Mahler et al.[84] as a pretrained model θ_0 and pretrained data $\mathcal{D}_0$, respectively. We samples $k = 172$ grasping examples from data in $\mathcal{D}_0$. For each round, we generate 64 grasp candidates and corresponding qualities from a given depth image using the pretrained model. We sample one grasp among 64 grasps by applying three sampling methods: greedy, ϵ-greedy, and SERN. By doing so, we can fairly verify effects of the sampling methods since all methods share the pretrained network and only differences are the exploration method. Each round consists of a exploration and evaluation phase. In exploration phase, we collect 20 grasp, image, and result pairs and update the grasp quality network with gathered data. In evaluation, we run the updated network 5 times by selecting the best grasp to verify the actual performance without an effect of exploration.

For the dynamic simulation, a GAZEBO simulator [69] is used with an open dynamics engine [121]. We utilize 3D mesh dataset from KIT [63] and T-Less [56]. As shown in Fig. 4.8(a) and 4.8(b), we select four mesh models from [63] and two mesh models from [56], respectively, which are hardly grasped by the pretrained model [84] in simulations. All algorithms run with five random seeds.

In the the real-world experiment, we compare two methods: ϵ-greedy and the proposed method. We also conduct both exploration and evaluation steps separately. In exploration, we gather 5 grasp examples and, in evaluation, we measure 5 grasp tests. We select three objects: triangle, round stapler, and vertical mouse, which are hardly grasped by a parallel jaw gripper due to its rounded surface and nonparallel shape as shown in Fig. 4.8(c). We use a Baxter robot which has a 7 DoF manipulator to grasp the objects and a RealSense D435 depth camera.

Simulation Results

We measure the improvement of grasp success rate between the first and last performance, and the maximum grasp success rate among success rates of five rounds. The results are shown in Table 4.1.

First, the greedy policy without exploration shows the worst performance in terms of the maximum performance and final performance. From this observation, it is shown that exploration is essential for learning to grasp. Since the greedy policy tries similar grasps when gathering 20 exploratory grasp examples, it cannot gather diverse training data, which causes the over-fitting issue.

On the contrary, the SERN and ϵ-greedy method show better performance than the greedy method. In particular, the proposed method, SERN, outperforms other methods in terms of both maximum grasp success rate and final grasp success rate. In all cases, after training for five rounds, the grasp success rates of SERN are improved compared to initial performances. In particular, the success rate for the CokePlastic increase by 212%.

While ϵ-greedy method outperforms the greedy method in three objects as shown in Table 4.1, it shows poorer performance than the SERN. Since the SERN samples a grasp based on a softmax distribution of estimated grasp qualities, potentially feasible grasp poses are first searched. The ϵ-greedy method, however, explores all grasp poses randomly and it causes inefficiency of exploration in practice. Since we employ the pretrained network, sampling a grasp pose based on the grasp quality estimation from the pretrained network shows better performance while ϵ-greedy method . Thus, the SERN shows the best performance compared to greedy and ϵ-greedy method.

Real-World Results

The results are shown in Table 4.2. In the real-world experiments, the SERN outperforms the ϵ-greedy method. In particular, for the Mouse object, the SERN finds success grasps much faster than ϵ-greedy so that it achieves 60% success rate at the first round and outperforms 80%. These results support the fact that using softmax distribution has benefits over the ϵ-greedy method since it frequently explores the grasp poses that has the high chance of success. Furthermore, the softmax distribution is more suitable than the ϵ-greedy method to employ the pretrained model. From this reason, the SERN generally shows the better performance than the ϵ-greedy method.

Obj.	Alg.	Round 1.	Round 2.	Round 3.	Round 4.	Round 5.	Max.	Imprv.
Marjoram	SERN	52% (±4.90)	72% (±10.20)	88% (±6.32)	75% (±11.66)	**75%** (±6.32)	**88%**	44%
	ϵ-Greedy	72% (±4.90)	55% (±4.90)	61% (±9.80)	76% (±10.20)	70% (±6.32)	76%	−3%
	Greedy	48% (±12.00)	60% (±6.32)	64% (±7.48)	68% (±10.20)	68% (±8.00)	68%	42%
SaltCylinderSmall	SERN	68% (±10.20)	60% (±10.95)	56% (±9.80)	56% (±11.66)	**76%** (±11.66)	**76%**	12%
	ϵ-Greedy	64% (±11.66)	52% (±12.00)	60% (±8.94)	60% (±8.94)	56% (±4.00)	64%	−13%
	Greedy	64% (±11.66)	52% (±10.20)	48% (±4.90)	40% (±6.32)	48% (±10.20)	64%	−25%
BathDetergent	SERN	36% (±13.27)	48% (±13.56)	48% (±8.00)	60% (±6.32)	**64%** (±7.48)	**64%**	78%
	ϵ-Greedy	60% (±16.73)	52% (±13.56)	48% (±12.00)	60% (±18.97)	52% (±16.25)	60%	−13%
	Greedy	36% (±11.66)	40% (±6.32)	36% (±9.80)	60% (±6.32)	20% (±6.32)	60%	−44%
CokePlasticLarge	SERN	16% (±4.00)	23% (±8.94)	29% (±11.66)	40% (±8.00)	**50%** (±14.14)	**50%**	212%
	ϵ-Greedy	32% (±8.00)	46% (±7.48)	21% (±4.00)	28% (±6.32)	46% (±7.48)	46%	44%
	Greedy	12% (±4.90)	15% (±4.90)	33% (±4.90)	28% (±8.94)	26% (±7.48)	33%	117%
T-Less 10	SERN	56% (±7.48)	74% (±10.20)	52% (±14.14)	71% (±12.00)	**81%** (±7.48)	**81%**	45%
	ϵ-Greedy	52% (±4.90)	50% (±9.80)	61% (±12.00)	71% (±8.00)	73% (±13.56)	73%	40%
	Greedy	36% (±7.48)	50% (±7.48)	49% (±7.48)	55% (±9.80)	69% (±16.00)	69%	92%
T-Less 20	SERN	72% (±8.00)	60% (±10.95)	60% (±6.32)	72% (±10.20)	**84%** (±4.00)	**84%**	17%
	ϵ-Greedy	72% (±8.00)	60% (±6.32)	44% (±4.00)	68% (±4.90)	68% (±4.90)	72%	−6%
	Greedy	68% (±10.20)	76% (±4.00)	60% (±6.32)	64% (±9.80)	68% (±13.56)	76%	0%

Table 4.1: Grasp success rate in simulation. The number in the parenthesis indicates a standard deviation. Obj. indicates a name of 3D mesh in KIT and T-Less dataset. Max. is the maximum success rate achieved during five trials. Imprv. is a performance improvement after training compared to the first performance, which is computed as $(r_5 - r_1)/r_1$ where r_i is the ith success rate. The best performances are marked in bold.

The main benefit of SERN compared to ϵ-greedy is the exploration tendency. In ϵ-greedy, the exploration is conducted by an uniform distribution. Thus, ϵ-greedy tries random grasps with ϵ ratio. On the contrast, SERN combine both exploitation and exploration since the greedy action has the largest probability mass and the other actions have the probability mass proportional to its grasp quality.

The examples of grasp candidate are shown in Fig. 4.9. Fig. 4.9(a) and (b) are

Obj.	Alg.	Rnd 1.	Rnd 2.	Rnd 3.	Rnd 4.	Rnd 5.
Triangle	SERN	0%	20%	20%	60%	**80%**
	ϵ-Greedy	0%	20%	0%	40%	40%
Stapler	SERN	0%	0%	40%	80%	**80%**
	ϵ-Greedy	0%	0%	40%	0%	40%
Mouse	SERN	60%	60%	80%	80%	**80%**
	ϵ-Greedy	0%	20%	20%	20%	20%

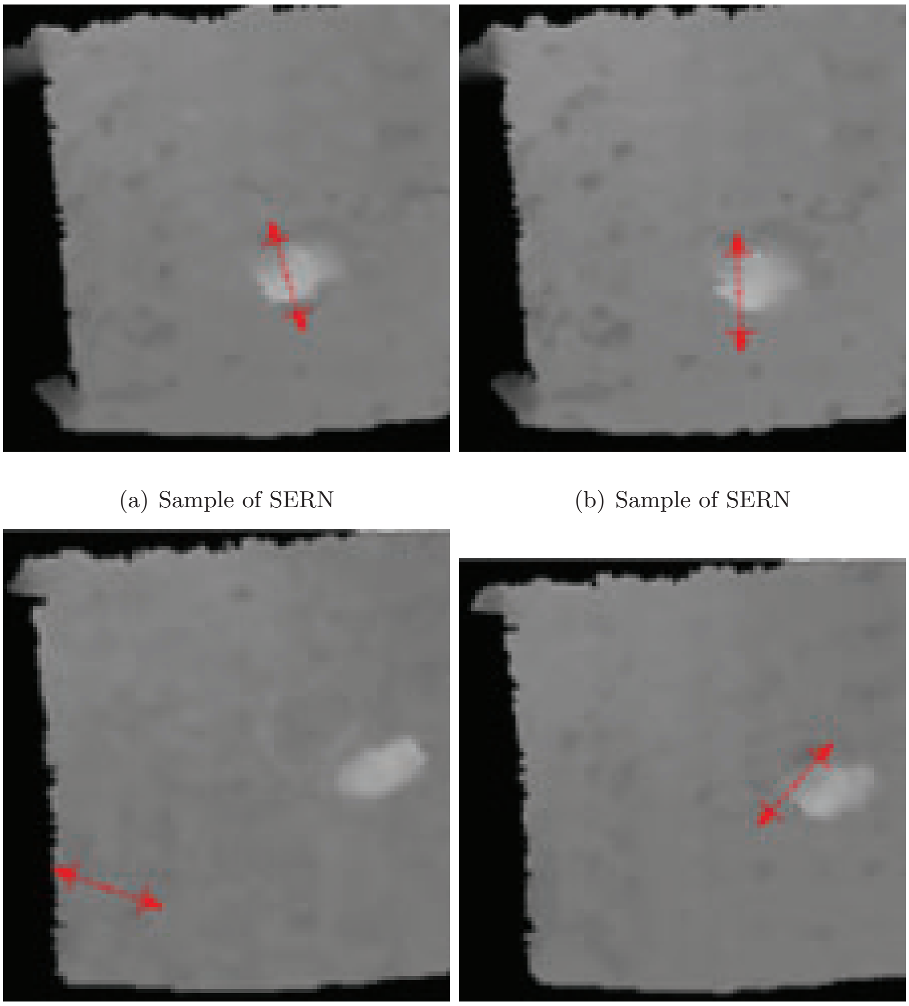
Table 4.2: Grasp success rate in the real-world experiments. Rnd i indicates the ith round. The best performances are marked in bold.

examples of SERN and Fig. 4.9(c) and (d) are examples of ϵ-greedy, respectively. Since ϵ-greedy fully random exploration, unrealistic grasp is selected as shown in Fig. 4.9 (c).

However, SERN tries more promising grasps which have the potential to successfully grasp the object. Specifically, since we employ a pretrained model, the grasp candidate with high estimated quality have the high potential to success to grasp the object as shown in Fig. 4.9(a) and (b). In SERN, $\mathbb{P}(a_t = a)$ is proportional to $\exp\left(\hat{Q}_a\right)$ where $\hat{Q}_a$ is a grasp quality of the grasp a. Thus, we can conclude that exploration with soft max distribution has the benefit in practice.

4.2.5 Summary

In this section, we have proposed a novel Shannon entropy regularized neural contextual bandit online learning (SERN) and have applied SERN to learning to grasp unknown objects. We also proved that SERN has no regret properties and its error converges to zero. We would like to emphasize that we analyzes the effect



(a) Sample of SERN

(b) Sample of SERN

(c) Sample of ϵ-Greedy

(d) Sample of ϵ-Greedy

Figure 4.9: Grasp candidates sampled from SERN and ϵ-Greedy

of using a neural network in the contextual bandit framework. In both simulation and the real-world experiments, we empirically show that SERN outperforms a ϵ-greedy method and improves the grasp performance efficiently.

Chapter 5

Perturbation-Based Exploration

Designing an efficient exploration strategy is important in online learning problems such as designing medical experiment [43] and exploration strategy in reinforcement learning [122, 127, 119]. These problems ask a learning agent the ability to learn an optimal action from trial and error without using prior knowledge of rewards. For each trial, the agent takes an *action* based on prediction, and obtains a *feedback*, often called a *reward*, as a result. If the rewards of all actions are given, the problem is called a *full information* problem. Otherwise, if single reward of a chosen action is given, it is called a *bandit* or *partial information* problem. In both full and partial information settings, the efficiency of an exploration strategy is measured by a *regret* which is the difference between the sum of rewards of optimal decisions and that of the decisions of the exploration strategy.

Under the full information setting, an algorithm called a Follow-the-Perturbed-Leader (FTPL) has been widely and intensively investigated [4, 61, 71]. FTPL stochastically smooths a greedy decision using various types of random perturbation. While FTPL has a benefit of simplicity, it is well known that its regret analysis is complicated since it highly depends on the chosen distribution of the perturbation. Thus, a different analysis technique has been developed for each type of perturbation [61, 71, 36]. Notably, Abernethy et. al. [4] has proposed a general analysis scheme by revealing the relation between the regret bound and the distribution of perturbation. This framework is also applied to the bandit setting in [5]. Correctly, it has been shown that various choices of distributions, including heavy-tailed densities such as Gumbel, Fréchet, and Pareto, can achieve an optimal regret bound in an *adversarial bandit* setting whose sequence of rewards is determined by an adversary. These results have shown that FTPL has achieved the nearly optimal regret bound over different settings. Here, the crucial and natural question arises: *can general analysis scheme be also available in a stochastic bandit setting whose rewards are identically independently distributed for each trial?* One seminal work [64] provides a pioneering answer to the question; it proposes an analysis method of the regret bound for a family of sub-Weibull perturbations and that of all perturbations with bounded support under stochastic bandit problems whose rewards have sub-Gaussian noises. In the following sections, we extend Kim et. al. [64] into two directions. First, we propose more general analysis scheme of regret bounds of perturbation methods including heavy-tailed perturbations such as Fréchet, Pareto, Gamma, Generalized Extreme Value (GEV) distributions. Second, we also develop a general regret analysis scheme to apply perturbation methods to heavy-tailed rewards.

5.1 Perturbed Exploration for sub-Gaussian Rewards

In this section, we investigate a unified framework to obtain the regret bound of the perturbation method with various types of distributions under sub-Gaussian rewards in the stochastic multi-armed bandit. By using this framework, we are able to deal with Pareto, Fréchet, and generalized extreme value distributions which are not covered by the results of [64]. In this framework, we modified the algorithm of FTPL proposed in [64] by controlling the magnitude of perturbation based on the number of times each action has been selected. We call this method as **A**daptively **P**erturbed **E**xploration (APE).

Our analysis scheme reveals the connection between the regret bound and the density of random perturbation provided rewards are sub-Gaussian. It requires general conditions to achieve a sub-linear regret bound of various perturbations: Gaussian, Weibull, Generalized extreme value (GEV) [143], Fréchet, and Pareto distributions. Each regret bound shows its relationship with the parameter of the distribution of each perturbation. Furthermore, these bounds allow us to choose the optimal parameter which can achieve the nearly optimal regret bound. Especially, GEV is a generalized version of Gumbel distribution including Fréchet. While the results for Gaussian and Weibull distribution are already analyzed in [64], these results tell us that our framework includes the results of [64]. We emphasize that our scheme provides a standard guide when designing a perturbation method for a stochastic bandit with sub-Gaussian rewards.

5.1.1 Related Work

In a stochastic multi-armed bandit problem under sub-Gaussian rewards, Kim et. al. [64] have proposed the general analysis for two types of the random perturbation method. The first one is the sub-Weibull perturbation under the anti-concentration assumption and the second one is the bounded perturbation most of whose probability mass is placed at the extreme of the support. Kim et. al. [64] have shown that the sub-Weibull perturbation with anti-concentration condition achieves the problem-independent regret bound of $O(\sqrt{KT\ln(K)})$ with its parameter $q = 2$, which matches the regret bound of Thompson sampling [130] under the Gaussian prior assumption on the reward [7]. Furthermore, Kim et. al. [64] have provided a regret bound of $O(\sqrt{KT\ln(T)})$ for the bounded perturbation method, which is analogous to the upper confidence bound (UCB) under the sub-Gaussian reward assumption [12]. In this section, we focus on extending the range of unbounded perturbations from the sub-Weibull to the heavy-tailed distribution with a polynomial tail. While the general analysis scheme of [64] allow us to analyze the sub-Weibull perturbation for the sub-Gaussian reward, it has the limitation in that the sub-Weibull assumption cannot include the heavy-tailed distributions whose tail probability decays polynomially fast.

Bandit with sub-Gaussian Rewards We consider a stochastic multi-armed bandit problem which is explained in Chapter 2.1.1. A usual condition for noise ϵ_t is *sub-Gaussian*: for $\lambda \in \mathbb{R}_+$

$$\mathbb{E}\left[\exp\left(\lambda\epsilon_t\right)\right] \leq \exp\left(-\lambda^2\sigma^2\right), \tag{5.1}$$

where σ is the parameter of ϵ_t. Suppose the agent employs an average estimation for rewards as follows

$$\hat{r}_{t,a} := \frac{1}{n_{t-1,a}} \sum_{k=1}^{t-1} \mathbf{R}_{k,a}\mathbb{I}\left[a_k = a\right], \tag{5.2}$$

where $n_{t-1,a}$ is the number of times action a has been selected for $t-1$ rounds. Combining (5.1) and reward estimator (5.2), we can derive a Chernoff-type bound:

$$\mathbb{P}\left[\hat{r}_{t,a} - r_a > \delta\right] \leq \exp\left(-\frac{\delta^2 n_{t-1,a}}{2\sigma^2}\right). \tag{5.3}$$

The bound (5.3) shows the ratios of the error reduction of the reward estimator under sub-Gaussian property.

5.1.2 Heavy-Tailed Perturbations

In our algorithm, G denotes the perturbation for exploration. Let the support of G be $[g_-, g_+]$, where $-\infty \leq g_- < g_+ \leq \infty$. We handle the unbounded case $|g_-| + |g_+| = \infty$ that the support is $\mathbb{R}$, or the case that the support is $[g_-, \infty)$. We refer to Kim and Tewari [64] for the bounded support case.

The cumulative density function F for G is given as follows,

$$F(x) := \begin{cases} 0 & x < g_- \\ \mathbb{P}\left[G < x\right] & g_- \leq x \leq g_+ \\ 1 & g_+ \leq x \end{cases}$$

$F(x)$ plays an important role for both implementation and analysis. In implementation perspective, we can sample perturbation G from an inverse function F^{-1} by $G = F^{-1}(U)$ where U is sampled from the uniform distribution on $[0, 1]$. We establish the regret analysis of the proposed perturbation method based on the following assumption:

Assumption 4. *Let $h(x) := \frac{d}{dx} \log(1-F(x))^{-1}$ be a hazard rate. Suppose $F(0) \leq 1/2$, F is log-concave and reward has sub-Gaussian noise* (5.1)*, there exists a constant $C_{F,c,\sigma}$ such that*

$$\int_0^\infty \frac{h(x)\exp\left(-\frac{c^2x^2}{2\sigma^2}\right)}{1-F(x)} dx \leq C_{F,c,\sigma} < \infty. \tag{5.4}$$

If h is bounded i.e. $\|h\|_\infty < \infty$, then, the condition (5.4) *is reduced to the existence of a constant $M_{F,c,\sigma}$ such that*

$$\int_0^\infty \frac{\exp\left(-\frac{c^2x^2}{2\sigma^2}\right)}{1-F(x)} dx \leq M_{F,c,\sigma} < \infty, \tag{5.5}$$

where $C_{F,c,\sigma} \leq \|h\|_\infty M_{F,c,\sigma}$.

Assumption 4 is an advanced version of the conditions introduced in [64], which provides the regret analysis with the following two-sided tail behavior bounds:

$$\exp\left(-\frac{x^q}{2\sigma^q}\right) \leq \mathbb{P}\left(|G| \geq x\right) \leq \exp\left(-\frac{x^p}{2\sigma^p}\right). \tag{5.6}$$

Here the lower bound is called *anti-concentration* condition with parameter q and the upper bound is called *sub-Weibull* condition with parameter p provided $p < q \leq 2$. The anti-concentration property tells us the lower bound of tail probability decays at most exponentially fast with respect to the order $q \leq 2$. Thus, under (5.6), G automatically satisfies (5.4) provided sub-Gaussian rewards are given.

Therefore, we propose a unified framework. For sub-Gaussian rewards, our assumption (5.4) contains *non*-sub-Weibull perturbations which include Pareto, Fréchet, and generalized extreme value (GEV) distributions. Since these densities have polynomial tail behavior, they only satisfy the lower bound (anti-concentration), but the upper bound (sub-Weibull) in (5.6). Consequently, the proposed framework covers both more general perturbations.

Let us explain the probabilistic intuition in Assumption 4. Observe that, due to the Chernoff-type bounds of (5.3), the integral in (5.4) has the lower bound:

$$\int_0^\infty h(x)\frac{\mathbb{P}\left[\epsilon > x\right]}{\mathbb{P}\left[G > x\right]}dx \leq \int_0^\infty \frac{h(x)\exp\left(-\frac{c^2x^2}{2\sigma^2}\right)}{1-F(x)}dx.$$

Thus, the integral of Assumption 4 is bounded below by the integral of the ratio between $\mathbb{P}\left[\epsilon > x\right]$ and $\mathbb{P}\left[G > x\right]$. One can easily observe that the ratio $\mathbb{P}\left[\epsilon > x\right]/\mathbb{P}\left[G > x\right]$ necessarily converges to zero at the tail to validate the integral condition (5.4). Otherwise, the lower bound diverges unless the hazard rate h vanishes at infinity, which does not hold even for simple exponential distribution. Hence, Assumption 4 requires the perturbation to have the tail probability decays slower than that of reward. This requirement can be interpreted as the condition to be essential to overcome noisy reward using the perturbation. If the algorithm misclassifies an optimal action due to the noise ϵ, to surmount the misclassification by exploring other actions, the sampled perturbations should be greater than the sampled noises and it is determined by the ratio between $\mathbb{P}[\epsilon > x]$ and $\mathbb{P}[G > x]$. This interpretation matches the intuition of (5.6) which is already introduced in [64].

5.1.3 Adaptively Perturbed Exploration

In this section, we provides the main theorem of the regret bounds for *Adaptively Perturbed Exploration* (APE) in the stochastic multi-armed bandit setting. The complete proofs of theorems, lemmas, corollaries in this section can be found in the supplementary material.

In APE, similar to the FTPL method, we sample a random noise $G_{t,a}$ at round t from perturbation distribution $F(x)$. Then, the action is taken by the following rule,

$$a_t = \underset{a \in \mathcal{A}}{\operatorname{argmax}} \ \{\hat{r}_{t-1,a} + \beta_{t-1,a} G_{t,a}\} \tag{5.7}$$

where $\beta_{t-1,a}$ is defined as

$$\beta_{t-1,a} := \frac{c}{\sqrt{\max(n_{t-1,a}, 1)}}.$$

Here c is a parameter controlling the magnitude of perturbation. Intuitively speaking, as $n_{t-1,a}$ increases, $\beta_{t-1,a}$ gradually decreases and it reduces exploration of action a. Thus, the level of exploration is adaptive depending on $n_{t-1,a}$ for each action. The entire algorithm is summarized in Algorithm 5.

Algorithm 5 Adaptively Perturbed Exploration (APE)

```
Require: $c, T$, and $F^{-1}(y)$
Initialize $\{\hat{r}_{0,a} = 0\}$
for $t = 1, \cdots, T$ do
    for $\forall a \in \mathcal{A}$ do
        $u \sim \text{Uniform}(0, 1)$
        $\beta_{t-1,a} \leftarrow \frac{c}{\sqrt{\max(n_{t-1,a}, 1)}}$ and $G_{t,a} \leftarrow F^{-1}(u)$
    end for
    $a_t = \arg\max_a \hat{r}_{t-1,a} + \beta_{t-1,a} G_{t,a}$
    Receive $\mathbf{R}_{t,a_t}$
    $\hat{r}_{t,a_t} \leftarrow \frac{n_{t-1,a_t}\hat{r}_{t-1,a_t} + \mathbf{R}_{t,a_t}}{n_{t-1,a_t}+1}$
    $n_{t,a_t} \leftarrow n_{t-1,a_t} + 1$
end for
```

5.1.4 General Regret Bound for Sub-Gaussian Rewards

First, we provide the lower bound of the regret of APE.

Theorem 26. *Under Assumption 4, for $c \in (0, 1)$ and $T \geq \frac{4}{c(1-c)(K-1)}$, the regret of APE satisfies*

$$\mathbb{E}\left[\mathcal{R}_T\right] \geq \Omega\left(\sqrt{KT}F^{-1}\left(1 - K^{-1}\right)\right). \tag{5.8}$$

Theorem 26 gives the lower regret bound of the log-concave F. We emphasize that this theorem holds for sub-Gaussian rewards setting and will be extended to heavy-tailed rewards setting.

Now, we present the relation between the upper bound of regret and the density of perturbation among different noises. For some distributions, the upper bound matches the lower bound up to the constant, which is called a tight bound.

The upper bound of the regret of APE is derived under the sub-Gaussian reward (5.1). Since the exploration is influenced by the random perturbation G, the regret bound of APE is highly related to F.

Theorem 27. *Suppose Assumption 4. For arbitrary $c > 0$,*

$$\mathbb{E}\left[\mathcal{R}_T\right] \leq \sum_{a \neq a^\star} \left[C_{F,c,\sigma} + \frac{F(0)}{1 - F(0)}\right] \frac{18\sigma^2}{\Delta_a} + 2\sum_{k=1}^{T} \Delta_a F\left(-\frac{\Delta_a\sqrt{k}}{6c}\right) \tag{5.9}$$

$$+ \frac{9c^2}{\Delta_a}\left[F^{-1}\left(1 - \frac{c^2}{T\Delta_a^2}\right)\right]^2 + \frac{c^2 + 162\sigma^2}{\Delta_a} + \Delta_a \tag{5.10}$$

If $\|h\|_\infty$ and $M_{F,c,\sigma}$ are finite, we can use $\|h\|_\infty M_{F,c,\sigma}$ instead of $C_{F,c,\sigma}$.

Sketch of Proof. The main difference of this proof from that of [64] is the Lemma 8. We first separate a set $E_{t,a} := \{a_t = a\}$ into three subsets based on the following thresholds: $x_a := r_a + \frac{\Delta_a}{3}$ and $y_a := r_{a^\star} - \frac{\Delta_a}{3}$. Let $\hat{E}_{t,a} := \{\hat{r}_{t,a} \leq x_a\}$, and $\tilde{E}_{t,a} := \{\hat{r}_{t,a} + \beta_{t,a}G_{t+1,a} \leq y_a\}$. Both $\hat{E}_{t,a}$ and $\tilde{E}_{t,a}$ indicate the estimator $\hat{r}_a$ and the perturbed estimation do not deviate from true reward r_a.

Next, we decompose $E_{t,a}$ into three groups,

$$\left(E_{t,a} \cap \hat{E}_{t,a}^{\mathsf{c}}\right) \cup \left(E_{t,a} \cap \hat{E}_{t,a} \cap \tilde{E}_{t,a}\right) \tag{5.11}$$

$$\cup \left(E_{t,a} \cap \hat{E}_{t,a} \cap \tilde{E}_{t,a}^{\mathsf{c}}\right). \tag{5.12}$$

The first term indicates the set of events that the estimation is inaccurate enough. The second term indicates the set of events that sub-optimal action a is selected even though the estimation is accurate enough and the perturbation for the action is not large. These events occur when the estimation for $a^\star$ is incorrect or the perturbation for $a^\star$ is relatively smaller than that of a. The third term indicates the events that the perturbation is too large even though the estimation is correct enough. Following lemmas shows the bounds of each set of events.

Lemma 7. *For any $a \in \mathcal{A}$,*

$$\sum_{t=1}^{T} \mathbb{P}\left[E_{t,a} \cap \hat{E}_{t,a}^{\mathsf{c}}\right] \leq 1 + \frac{18\sigma^2}{\Delta_a^2}. \tag{5.13}$$

The first bound indicates how fast the estimation is concentrated on the true reward where the concentration speed depends on σ of the sub-Gaussian property. Thus, Lemma 7 tells us that the regret caused by the inaccurate estimations is bounded since the bound is exponentially decayed.

Lemma 8. *For any $a \in \mathcal{A}$,*

$$\sum_{t=1}^{T} \mathbb{P}\left[E_{t,a} \cap \hat{E}_{t,a} \cap \tilde{E}_{t,a}\right] \leq 2\sum_{k=1}^{T} F\left(-\frac{\Delta_a\sqrt{k}}{6c}\right) \tag{5.14}$$

$$+\left[C_{F,c,\sigma} + \frac{F(0)}{1-F(0)}\right]\frac{18\sigma^2}{\Delta_a^2} + \frac{144\sigma^2}{\Delta_a^2}. \tag{5.15}$$

If $\|h\|_\infty$ and $M_{F,c,\sigma}$ are finite, we can use $\|h\|_\infty M_{F,c,\sigma}$ instead of $C_{F,c,\sigma}$.

Since this bound depends on the set of events that estimation for $a^\star$ is not correct or the perturbation for $a^\star$ is relatively smaller than that of a, we derive the following bound,

$$\sum_{t=1}^{T}\mathbb{P}\left[E_{t,a} \cap \hat{E}_{t,a} \cap \tilde{E}_{t,a}\right] \leq \sum_{k=1}^{T} \mathbb{E}\left[\frac{p_{\tau_{a^\star,k},a^\star}}{1-p_{\tau_{a^\star,k},a^\star}}\right] \tag{5.16}$$

where $\tau_{a^\star,k}$ is the round that $a^\star$ is selected for the kth times and $p_{\tau_{a^\star,k},a^\star} = F\left((r_{a^\star} - \hat{r}_{\tau_{a^\star,k},a^\star} - \Delta_a/3)\sqrt{k}/c\right)$. Then, similarly to [64], we decompose the events into three cases. The first case is when $\hat{r}_{a^\star}$ is under estimated. The second case is when the perturbation is too small while the estimation error is small enough. The last case is when $\hat{r}_{a^\star}$ is over estimated. Then, the bounds are separately computed.

For the first case, the first term is derived,

$$\left[C_{F,c,\sigma} + \frac{F(0)}{1-F(0)}\right]\frac{18\sigma^2}{\Delta_a^2}.$$

The ratio $\frac{F(x)}{1-F(x)}$ is bounded by the constant in Assumption 4. Furthermore, if the hazard rate is bounded, we can use $\|h\|_\infty M_{F,c,\sigma}$ instead of $C_{F,c,\sigma}$. For the second case, the second term is derived,

$$144\sigma^2/\Delta_a^2,$$

where this bound indicates the regret caused by the estimation error of $\hat{r}_{t,a}$. Finally, the third term is derived by,

$$2\sum_{k=1}^{T} F\left(-\Delta_a\sqrt{k}/6c\right).$$

This part indicates the regret comes from the negative perturbations which makes $a^\star$ not to be selected. Furthermore, we would like to note that if the support of the perturbation does not contain a negative real line, then, the last term becomes zero.

Lemma 9. *For any $a \in \mathcal{A}$,*

$$\sum_{t=1}^{T} \mathbb{P}\left[E_{t,a} \cap \hat{E}_{t,a} \cap \tilde{E}_{t,a}^{\mathsf{c}}\right] \leq \frac{9c^2}{\Delta_a^2}\left[F^{-1}\left(1-\frac{c^2}{T\Delta_a^2}\right)\right]^2 + \frac{c^2}{\Delta_a^2} \tag{5.17}$$

This term relies on how fast the level of perturbation decreases. Specifically, this term is bounded by the time threshold until the probability becomes smaller than T^{-1}. Thus, $\left[F^{-1}\left(1-\frac{c^2}{T\Delta_a^2}\right)\right]^2$ is the number of rounds that is required to make $\mathbb{P}\left[E_{t,a} \cap \hat{E}_{t,a} \cap \tilde{E}_{t,a}^{\mathsf{c}}\right]$ below T^{-1}.

By combining Lemma 7, 8, and 9, the proof is completed. □

5.1.5 Regret Bounds for Specific Perturbations with sub-Gaussian Rewards

Now, from Theorem 27, we derive the regret bounds for specific distributions including Weibull, Gaussian, Pareto, Fréchet, GEV, and logistic distribution. For Pareto distribution, we apply simple modification made in Abernethy et al. [5], which is called modified Pareto distribution.

We first introduce the regret bound of APE with Weibull and Gaussian distributions which disobey $\|h\|_\infty < \infty$.

Corollary 1. *Suppose G follows Weibull distribution with a shape parameter $k < 2$ with $c > 0$ or $k = 2$ with $c > \sqrt{\frac{2\sigma^2}{\lambda^2}}$ and a scale parameter $\lambda > 0$. Then, the problem-dependent regret bound is*

$$\mathbb{E}\left[\mathcal{R}_T\right] \leq C\left(\sum_{a\neq a^\star} \Delta_a + \frac{9c^2}{\Delta_a}\left[\ln\left(\frac{T\Delta_a^2}{c^2}\right)\right]^{\frac{2}{k}}\right). \tag{5.18}$$

The problem-independent regret bound is

$$\mathbb{E}\left[\mathcal{R}_T\right] =\Theta\left(\sqrt{KT}\ln(K)^{1/k}\right). \tag{5.19}$$

The optimal rate is achieved at $k = 2$,

$$\mathbb{E}\left[\mathcal{R}_T\right] = \Theta\left(\sqrt{KT\ln(K)}\right). \tag{5.20}$$

Corollary 2. *Suppose G follows zero-mean Gaussian distribution with a standard deviation parameter $\sigma_g > 0$ with $c > \sqrt{\frac{\sigma^2}{\sigma_g^2}}$. Then, the problem-dependent regret bound is*

$$\mathbb{E}\left[\mathcal{R}_T\right] \leq C\left(\sum_{a\neq a^\star} \Delta_a + \frac{18c^2}{\Delta_a}\left[\ln\left(\frac{T\Delta_a^2}{c^2}\right)\right]\right). \tag{5.21}$$

The problem-independent regret bound is

$$\mathbb{E}\left[\mathcal{R}_T\right] = \Theta\left(\sqrt{KT\ln(K)}\right), \tag{5.22}$$

where it is the optimal rate.

Corollary 1 and 2 show that our analysis scheme recovers the previous results thoroughly for both Weibull and Gaussian perturbations in Kim and Tewari [64]. More specifically, Corollary 1 and 2 match Theorem 3 and Corollary 4 in Kim and Tewari [64], respectively. Additionally, we introduce novel results for Pareto and Fréchet distributions which violate the sub-Weibull property. Since hazard rates of these distribution are bounded, $\|h\|_\infty$ is finite. The remainder is to check whether $M_{F,c,\sigma} < \infty$ according to Assumption 6.

Corollary 3. *Suppose G follows a modified Pareto distribution with a shape parameter $\alpha > 1$ and a scale parameter $\lambda \geq \alpha$. Then, the problem-dependent regret bound is*

$$\mathbb{E}\left[\mathcal{R}_T\right] \leq C\left(\sum_{a \neq a^\star} \Delta_a + \frac{9c^2\lambda^2}{\Delta_a}\left[\left(\frac{T\Delta_a^2}{c^2}\right)^{\frac{1}{\alpha}} - 1\right]^2\right). \tag{5.23}$$

For $\lambda^2 = \alpha$, the problem-independent regret bound is

$$\mathbb{E}\left[\mathcal{R}_T\right] \leq O\left(\sqrt{\alpha^{1+\frac{4}{\alpha}} K^{1+2/\alpha} T}\right). \tag{5.24}$$

The lower bound is

$$\mathbb{E}\left[\mathcal{R}_T\right] \geq \Omega\left(\sqrt{K^{1+2/\alpha} T}\right). \tag{5.25}$$

The optimal rate is obtained by setting $\alpha = \ln(K)$,

$$\mathbb{E}\left[\mathcal{R}_T\right] \leq O\left(\sqrt{KT\ln(K)}\right). \tag{5.26}$$

Corollary 4. *Suppose G follows Fréchet distribution with a shape parameter $1 < \alpha$ and scale parameter λ with $\frac{\sigma^2\alpha}{2c^2} \leq \lambda^2$. Then, the problem-dependent regret bound is*

$$\mathbb{E}\left[\mathcal{R}_T\right] \leq C\left(\sum_{a \neq a^\star} \Delta_a + \frac{9c^2\lambda^2}{\Delta_a}\left[\frac{T\Delta_a^2}{c^2}\right]^{2/\alpha}\right). \tag{5.27}$$

The problem-independent regret bound is

$$\mathbb{E}\left[\mathcal{R}_T\right] \leq O\left(\sqrt{\alpha^{1+\frac{4}{\alpha}} K^{1+\frac{6}{\alpha}} T}\right), \tag{5.28}$$

where $\lambda^2 = \frac{\sigma^2\alpha}{2c^2}$. The lower bound is

$$\mathbb{E}\left[\mathcal{R}_T\right] \geq \Omega\left(\sqrt{K^{1+\frac{2}{\alpha}} T}\right). \tag{5.29}$$

The optimal rate is achieved at $\alpha = \ln(K)$,

$$\mathbb{E}\left[\mathcal{R}_T\right] \leq O\left(\sqrt{KT\ln(K)}\right). \tag{5.30}$$

Interestingly, Corollary 3 tells us that using Pareto distribution also can achieve the same nearly optimal bound. We would like to emphasize that, to the best of our knowledge, this result is the first analysis of both upper and lower regret bound of using Pareto distribution in the stochastic bandit setting. Furthermore, we provide the problem-independent regret bound with respect to arbitrary $\alpha > 1$. However, unfortunately, the lower bound does not match to the upper bound of the Pareto distribution.

In Corollary 4, the bound tells us that APE with Fréchet distribution achieves the near-optimal bound. Since the integral in Assumption 6 is influenced by the parameter α and λ, the range of λ is essential to make $M_{F,c,\sigma}$ independent to α and λ. Note that the Fréchet distribution with the optimal parameter achieves the same optimal regret bound compared to the Gaussian perturbation in [64].

Corollary 3 and 4 also give a meaningful observation that the optimal rate is achieved by setting $\alpha = \ln(K)$ which is the same condition for adversarial bandit setting in [5] and demonstrates the analogous between stochastic and adversarial bandits. However, differently from the adversarial case [3], we should modify the scale parameter λ. This modification is required to make $M_{F,c,\sigma}$ bounded.

Now we show the regret bound for GEV distribution which is scarcely investigated under stochastic bandit setting.

Corollary 5. *Suppose G follows a GEV distribution with a shape parameter of $0 \leq \zeta < 1$. Then, the problem-dependent regret bound is*

$$\mathbb{E}\left[\mathcal{R}_T\right] \leq C \left(\sum_{a \neq a^\star} \Delta_a + \frac{9c^2}{\zeta^2 \Delta_a} \left[\left(\frac{T\Delta_a^2}{c^2} \right)^{\zeta} - 1 \right]^2 \right). \tag{5.31}$$

The problem-independent regret bound is

$$\mathbb{E}\left[\mathcal{R}_T\right] \leq O\left(\sqrt{KT} \ln_\zeta \left(K^2\right)^2 / \ln_\zeta(K) \right), \tag{5.32}$$

where $\ln_\zeta(x) := \frac{x^\zeta - 1}{\zeta}$. The lower bound is

$$\mathbb{E}\left[\mathcal{R}_T\right] \geq \Omega\left(\sqrt{KT} \ln_\zeta \left(K\right) \right). \tag{5.33}$$

The optimal rate is achieved at $\zeta = 0$*,*

$$\mathbb{E}\left[\mathcal{R}_T\right] = \Theta\left(\sqrt{KT}\ln(K)\right). \tag{5.34}$$

The Corollary 5 can be interpreted as the generalization of a Boltzmann-Gumbel exploration (BGE) in [29]. Furthermore, it is interesting that, for all parameter $0 \leq \zeta \leq 1$, using GEV shows near-optimal performance $O(\sqrt{KT})$ up to $\frac{\ln_\zeta(K^2)^2}{\ln_\zeta(K)}$ factor and also guarantees that Gumbel distribution has the best bound among all parameter $0 \leq \zeta \leq 1$.

All parameter settings such as F and $\|h\|_\infty$ are summarized in the supplementary where we provide the hyperparameters which can achieve the near-optimal bound.

5.1.6 Summary

In this chapter, we have analyzed the random perturbation method for a stochastic bandit setting under sub-Gaussian rewards. We have provided the general analysis scheme for the both upper and lower bound of the regret of heavy-tailed perturbations under sub-Gaussian rewards. We believe that the proposed analysis scheme is useful when designing the perturbation distribution for an online learning algorithm. Especially, our analysis scheme have made it available to analyze the heavy-tailed perturbations, such as Pareto, Fréchet, and GEV distribution which was not covered by the previous work [64]. The results of the Pareto and Fréchet perturbations have provided an interesting observation in that they can achieve the same near-optimal regret bound as the sub-Weibull perturbation under sub-Gaussian reward assumption.

5.2 Perturbed Exploration for Heavy-Tailed Rewards

Early researches for stochastic MABs have been investigated under the sub-Gaussian assumption on a reward distribution, which has the exponential-decaying behavior. However, there remains a large class of distributions which are not covered by the sub-Gaussianity and are called heavy-tailed distributions. While there exist several methods for handling such heavy-tailed rewards [26, 136], these methods have two main drawbacks. First, both methods utilize a class of robust reward estimators which require the prior knowledge about the bound on the moments of the rewards distributions, which is hardly available for practical problems. Furthermore, the algorithm proposed in [136] requires the gap information, which is the difference between the maximum and second-largest reward, to balance the exploration and exploitation. These features make the previous algorithms impractical since information about the bound or the gap is not accessible in general. Second, both methods have the sub-optimal gap-independent regret bound. Bubeck et al. [26] derive the lower bound of the regret for an arbitrary algorithm. However, the upper regret bound of the algorithms in [26, 136] does not match the lower regret bound. Thus, there exists a significant gap between the upper and lower bound, which can be reduced potentially. These drawbacks motivate us to design an algorithm which requires less prior knowledge about rewards yet achieves an optimal efficiency.

In this section, we propose a novel p-robust estimator which does not depend on prior information about the bound on the p-th moment $p \in (1, 2]$. Combined with this estimator, we develop a perturbed exploration method for heavy-tailed rewards. A perturbation-based exploration stochastically smooths a greedy policy by adding a random perturbation to the estimated rewards and selecting a greedy action based on the perturbed estimations; hence the distribution of the perturbation determines the trade-off between exploration and exploitation [61, 64]. We first analyze the regret bound of general perturbation method. Notably, we show that, if the tail probability of perturbations decays slower than the error probability of the estimator, then the proposed analysis scheme provides both upper and lower regret bounds. By using this general analysis scheme, we show that the optimal regret bound can be achieved for a broad class of perturbations, including Weibull, generalized extreme value, Gamma, Pareto, and Fréchet distributions. Empirically, the p-robust estimator shows favorable performance compared to the truncated mean and median of mean, which belong to the class of robust estimators [26]. For MAB problems, we also show that the proposed perturbation methods generally outperform robust UCB [26] and DSEE [136], which is consistent with our theoretical results.

The remaining parts of this section will be explained as follows. First, we derive the lower regret bound of robust UCB [26], which has the sub-optimal gap-independent regret bound. Second, we propose novel p-robust estimator which does not rely on prior information about the bound on the p-th moment of rewards and prove that its tail probability decays exponentially. Third, by combining the proposed estimator with the perturbation method, we develop a general regret analysis scheme by revealing the relationship between regret and cumulative density function of the perturbation. Finally, we show that the proposed strategy can achieve the optimal regret bound in terms of the number of rounds T, which is the first algorithm achieving the minimax optimal rate under heavy-tailed rewards.

Stochastic Multi-Armed Bandits with Heavy Tailed Rewards We consider a stochastic multi-armed bandit problem defined as a tuple $(\mathcal{A}, \{r_a\})$ where $\mathcal{A}$ is a set of K actions, and $r_a \in [0, 1]$ is a mean reward for action a. For each round t, the agent chooses an action a_t based on its exploration strategy and, then, get a stochastic reward: $\mathbf{R}_{t,a} := r_a + \epsilon_{t,a}$ where $\epsilon_{t,a}$ is an independent and identically distributed noise with $\mathbb{E}\left[\epsilon_{t,a}\right] = 0$ for all t and a. Note that r_a and $\epsilon_{t,a}$ are called the *mean of reward* and the *noise of reward*, respectively. r_a is generally assumed to be unknown. Then, the goal of the agent is to minimize the cumulative regret over total rounds T, defined as

$$\mathcal{R}_T := \sum_{t=1}^{T} r_{a^\star} - \mathbb{E}_{a_{1:t}}\left[r_{a_t}\right],$$

where $a^\star := \arg\max_{a \in \mathcal{A}} r_a$. The cumulative regret over T represents the performance of an exploration strategy. The smaller $\mathcal{R}_T$, the better exploration performance. To analyze $\mathcal{R}_T$, we consider the heavy-tailed assumption on noises whose p-th moment is bounded by a constant ν_p where $p \in (1, 2]$, i.e., $\mathbb{E}|\mathbb{R}_{t,a}|^p \leq \nu_p$ for all $a \in \mathcal{A}$. Without loss of generality, we regard p as the maximal order of the bounded moment, because, if the p-th moment is finite, then the moment with lower order is also finite automatically.

In this section, we analyze both gap-dependent and gap-independent regret bounds. The gap-dependent bound is the upper regret bound depending on the gap information $\Delta_a := r_{a^\star} - r_a$ for $a \neq a^\star$ and, on the contrary, the gap-independent bound is the upper regret bound independent of the gap.

5.2.1 Related Work

While various researches [117, 81, 93] have investigated heavy-tailed reward setting, they focused on variants of the MAB such as stochastic linear contextual bandit [117], Lipschitz bandit [81], or ϵ contaminated bandit [93]. In this paper, we focus on a conventional MAB problem and provide an optimal algorithm with respect to T. In a conventional MAB setting, few methods have handled heavy-tailed distributions [26, 136, 29, 60]. Bubeck et al. [26] have proposed robust UCB by employing a confidence bound of a class of robust estimators. Note that this class contains the truncated mean and the median of mean for $p \in (1, 2]$ and Catoni's M estimator for $p = 2$. Under these assumptions on rewards and estimators, robust UCB achieves the gap-dependent bound $O\left(\sum_a \ln(T)/\Delta_a^{1/(p-1)} + \Delta_a\right)$ and gap-independent bound $O\left((K\ln(T))^{1-1/p}T^{1/p}\right)$. However, to achieve this regret bound and to define a confidence bound of the robust estimator, prior knowledge of the bound of moments ν_p is required. This condition restricts the practical usefulness of robust UCB since ν_p is not accessible for many MAB problems. Furthermore, while it is proved that the lower regret bound of the MAB with heavy-tailed rewards is $\Omega(K^{1-1/p}T^{1/p})$, the upper regret bound of robust UCB has an additional factor of $\ln(T)^{1-1/p}$. A similar restriction also appears in [136]. Vakili et al. [136] have proposed a deterministic sequencing of exploration and exploitation (DSEE) by exploring every action uniformly with a deterministic sequence. It is shown that DSEE has the gap-dependent bound $O(\ln(T))$, but, its result holds when ν_p and the minimum gap $\min_{a\in\mathcal{A}} \Delta_a$ are known as prior information.

The dependence on ν_p was first removed in [29] for $p = 2$. Cesa-Bianchi et al. [29] have proposed a robust estimator by modifying the Catoni's M estimator and employed the Boltzmann-Gumbel exploration (BGE) with the robust estimator. In BGE, a Gumbel perturbation is used to encourage exploration instead of using a confidence bound of the robust estimator. One interesting observation is that the robust estimator proposed in [29] has a weak tail bound, whose error probability decays slower than that of Catoni's M estimator [28]. However, BGE achieved gap-dependent bound $O\left(\sum_a \ln(T\Delta_a^2)^2/\Delta_a + \Delta_a\right)$ and gap-independent bound $O(\sqrt{KT}\ln(K))$ for $p = 2$. While $\ln(K)$ factor remains, BGE has a better bound than robust UCB in terms of T when $p = 2$. Kagrecha et al. [60] also tried to remove the dependency on ν_p for $p \in (1, 2]$ by proposing a generalized successive rejects (GSR) method. While GSR does not depend on any prior knowledge of the reward distribution, however, GSR only focuses on identifying the optimal arm, also known as pure exploration [25], rather than minimizing the cumulative regret. Hence, GSR lose much reward during the learning process.

5.2.2 Sub-Optimality of Robust Upper Confidence Bounds

In this section, we discuss the sub-optimality of robust UCB [26] by showing the lower bound of robust UCB. The robust UCB employs a class of robust estimators which satisfies the following assumption.

Assumption 5. *Let $\{Y_k\}_{k=1}^{\infty}$ be i.i.d. random variables with the finite p-th moment for $p \in (1, 2]$. Let ν_p be a bound of the p-th moment and y be the mean of Y_k. Assume that, for all $\delta \in (0, 1)$ and n number of observations, there exists an estimator $\hat{Y}_n(\eta, \nu_p, \delta)$ with a parameter η such that*

$$\mathbb{P}\left(\hat{Y}_n > y + \nu_p^{1/p}\left(\frac{\eta \ln(1/\delta)}{n}\right)^{1-1/p}\right) \leq \delta,$$

$$\mathbb{P}\left(y > \hat{Y}_n + \nu_p^{1/p}\left(\frac{\eta \ln(1/\delta)}{n}\right)^{1-1/p}\right) \leq \delta.$$

This assumption naturally provides the confidence bound of the estimator $\hat{Y}_n$. Bubeck et al. [26] provided several examples satisfying this assumption, such as truncated mean, median of mean, and Catoni's M estimator. These estimators essentially require ν_p to define $\hat{Y}_n$. Furthermore, δ should be predefined to bound the tail probability of $\hat{Y}_n$ by δ. By using this confidence bound, at round t, robust UCB selects an action based on the following strategy,

$$a_t := \arg\max_{a \in \mathcal{A}} \left\{\hat{r}_{t-1,a} + \nu_p^{1/p}\left(\eta \ln(t^2)/n_{t-1,a}\right)^{1-1/p}\right\} \tag{5.35}$$

where $\hat{r}_{t-1,a}$ is an estimator which satisfies Assumption 7 with $\delta = t^{-2}$ and $n_{t-1,a}$ denotes the number of times $a \in \mathcal{A}$ have been selected. We first show that there exists a multi-armed bandit problem for which strategy (F.1) has the following lower bound of the expected cumulative regret.

Theorem 28. *There exists a K-armed stochastic bandit problem for which the regret of robust UCB has the following lower bound, for* $T > \max\left(10, \left[\frac{\nu^{\frac{1}{(p-1)}}}{\eta(K-1)}\right]^2\right)$,

$$\mathbb{E}[\mathcal{R}_T] \geq \Omega\left((K\ln(T))^{1-1/p}\, T^{1/p}\right). \tag{5.36}$$

The proof is done by constructing a counterexample which makes robust UCB have the lower bound (5.36) and the entire proof can be found in the supplementary material. Unfortunately, Theorem 28 tells us that the sub-optimal factor $\ln(T)^{1-1/p}$ cannot be removed and robust UCB has the tight regret bound $\Theta\left((K\ln(T))^{1-1/p}\, T^{1/p}\right)$ since the lower bound of (5.36) and upper bound in [26] are matched up to a constant. This sub-optimality is our motivation to design a perturbation-based exploration with a new robust estimator. Now, we discuss how to achieve the optimal regret bound $O\left(T^{1/p}\right)$ by removing the factor $\ln(T)^{1-1/p}$.

5.2.3 Adaptively Perturbed Exploration with A p-Robust Estimator

In this section, we propose a novel robust estimator whose error probability decays exponentially fast when the p-th moment of noises is bounded for $p \in (1, 2]$. Furthermore, we also propose an adaptively perturbed exploration with a p-robust estimator (APE^2). We first define a new influence function $\psi_p(x)$ as:

$$\psi_p(x) := \ln\left(b_p|x|^p + x + 1\right)\mathbb{I}[x \geq 0] - \ln\left(b_p|x|^p - x + 1\right)\mathbb{I}[x < 0] \tag{5.37}$$

where $b_p := \left[2\left((2-p)/(p-1)\right)^{1-2/p} + \left((2-p)/(p-1)\right)^{2-2/p}\right]^{-p/2}$ and $\mathbb{I}$ is an indicator function. Note that $\psi_p(x)$ generalizes the original influence function proposed in [28]. In particular, when $p = 2$, the influence function in [28] is recovered. Using $\psi_p(x)$, a novel robust estimator can be defined as the following theorem.

Theorem 29. *Let $\{Y_k\}_{k=1}^{\infty}$ be i.i.d. random variables sampled from a heavy-tailed distribution with a finite p-th moment, $\nu_p := \mathbb{E}\left|Y_k\right|^p$, for $p \in (1, 2]$. Let $y := \mathbb{E}\left[Y_k\right]$ and define an estimator as*

$$\hat{Y}_n := \frac{c}{n^{1-1/p}} \cdot \sum_{k=1}^{n} \psi_p\left(\frac{Y_k}{cn^{1/p}}\right) \tag{5.38}$$

where $c > 0$ is a constant. Then, for all $\epsilon > 0$,

$$\mathbb{P}\left(\hat{Y}_n > y + \epsilon\right) \leq \exp\left(-\frac{n^{\frac{p-1}{p}}\epsilon}{c} + \frac{b_p\nu_p}{c^p}\right), \tag{5.39}$$

$$\mathbb{P}\left(y > \hat{Y}_n + \epsilon\right) \leq \exp\left(-\frac{n^{\frac{p-1}{p}}\epsilon}{c} + \frac{b_p\nu_p}{c^p}\right). \tag{5.40}$$

The entire proof can be found in the supplementary material. The proof is done by employing the Chernoff-bound and the fact that $-\ln\left(b_p|x|^p - x + 1\right) \leq \psi_p(x) \leq \ln\left(b_p|x|^p + x + 1\right)$ where the definition of b_p makes the inequalities hold. Intuitively speaking, since the upper (or lower, resp.) bound of ψ_p increases (or decreases, resp.) sub-linearly, the effect of large noise is regularized in (5.38). We would like to note that the p-robust estimator is defined without using ν_p and its error probability decays exponentially fast for a fixed ϵ. Compared to Assumption 7, the confidence bound of (5.38) is looser than Assumption 7 for a fixed δ. The inequalities in Theorem 29 can be restated as

$$\mathbb{P}\left(\hat{Y}_n > y + c\frac{\ln\left(\frac{\exp(b_p\nu_p/c^p)}{\delta}\right)}{n^{1-1/p}}\right) \leq \delta$$

and

$$\mathbb{P}\left(y > \hat{Y}_n + c\frac{\ln\left(\frac{\exp(b_p\nu_p/c^p)}{\delta}\right)}{n^{1-1/p}}\right) \leq \delta$$

for all $\delta \in (0, 1)$. Hence, the confidence bound of (5.38) is wider (and looser) than Assumption 7 since $\ln(1/\delta) > \ln(1/\delta)^{1-1/p}$. In addition, the proposed estimator does not depends on ϵ (or δ) while Assumption 7 requires that δ is determined before defining $\hat{Y}_n(\eta, \nu_p, \delta)$.

Interestingly, we can observe that the p-robust estimator of Theorem 29 can recover Cesa's estimator [29] when $p = 2$. Thus, the proposed estimator extends the estimator of [29] to the case of $1 < p \leq 2$. We clarify that the estimator (5.38) extends Cesa's estimator but not Catoni's M estimator. While both estimators employ the influence function $\psi_2(x)$ when $p = 2$, Catoni's M estimator follows the Assumption 7 but not Theorem 29 since it requires prior information about δ and ν_p. Hence, the propose estimator dose not generalizes Catoni's M estimator.

Now, we propose an **A**daptively **P**erturbed **E**xploration method with **a p-**robust **E**stimator (APE^2), which combines the estimator (5.38) with a perturbation method. We also derive a regret analysis scheme for general perturbation methods. In particular, we find an interesting relationship between the cumulative density function (CDF) of the perturbation and its regret bound. Let F be a CDF of perturbation G defined as $F(g) := \mathbb{P}(G < g)$. We consider a random perturbation with unbounded support, such as $(0, \infty)$ or $\mathbb{R}$. Using F and the proposed robust estimator, APE^2 chooses an action for each round t based on the following rule,

$$a_t := \arg\max_{a \in \mathcal{A}} \hat{r}_{t-1,a} + \beta_{t-1,a} G_{t,a}, \quad \beta_{t-1,a} := \frac{c}{(n_{t-1,a})^{1-1/p}}, \tag{5.41}$$

where $n_{t,a}$ is the number of times a has been selected and $G_{t,a}$ is sampled from F. The entire algorithm is summarized in Algorithm 6.

Algorithm 6 Adaptively Perturbed Exploration with a p-robust estimator (APE^2)

```
Require: $c, T$, and $F^{-1}(y)$
Initialize $\{\hat{r}_{0,a} = 0, n_{0,a} = 0\}$, select $a_1, \cdots, a_K$ and receive $\mathbf{R}_{1,a_1}, \cdots, \mathbf{R}_{K,a_K}$ once
for $t = K+1, \cdots, T$ do
    for $\forall a \in \mathcal{A}$ do
        $\beta_{t-1,a} \leftarrow c/\left(n_{t,a}\right)^{1-1/p}$ and $G_{t,a} \leftarrow F^{-1}(u)$ with $u \sim \text{Uniform}(0,1)$
        $\hat{r}_{t-1,a} \leftarrow c/\left(n_{t,a}\right)^{1-1/p} \cdot \sum_{k=1}^{t-1} \mathbb{I}\left[a_k = a\right] \psi_p\left(\mathbf{R}_{k,a}/(c \cdot \left(n_{t,a}\right)^{1/p})\right)$
    end for
    Choose $a_t = \arg\max_{a \in \mathcal{A}}\{\hat{r}_{t-1,a} + \beta_{t-1,a} G_{t,a}\}$ and receive $\mathbf{R}_{t,a_t}$
end for
```

5.2.4 General Regret Bound for Heavy-Tailed Rewards

We propose a general regret analysis scheme which provides the upper bound and lower bound of the regret for APE^2 with a general $F(x)$. We introduce some assumptions on $F(x)$, which are sufficient conditions to bound the cumulative regret.

Assumption 6. *Let $h(x) := \frac{d}{dx}\log(1-F(x))^{-1}$ be a hazard rate. Assume that the CDF $F(x)$ satisfies the following conditions,*

- *F is log-concave, $F(0) \leq 1/2$, and there exists a constant C_F s.t.*

$$\int_0^\infty \frac{h(x)\exp(-x)}{1-F(x)} dx \leq C_F < \infty.$$

- *If h is bounded, i.e., $\sup_{x \in \mathrm{dom}(h)} h(x) < \infty$, then, the condition on C_F is reduced to the existence of a constant M_F such that $\int_0^\infty \frac{\exp(-x)}{1-F(x)} dx \leq M_F < \infty$ where $C_F \leq \sup h \cdot M_F$.*

The condition $F(0) \leq 1/2$ indicates that the half of probability mass must be assigned at positive perturbation $G > 0$ to make the perturbation explore under-estimated actions due to the noises. Similarly, the bounded integral condition is required for overcoming heavy-tailed noises of reward. Note that the error bound of our estimator follows $\mathbb{P}(\hat{Y}_n - y > x) \leq C\exp\left(-n^{1-1/p}x/c\right) \leq C\exp\left(-x\right)$ for $n > c^{p/(p-1)}$ where $C > 0$ is a some constant in Theorem 29. From this observation, the bounded integral condition can be interpreted as

$$\int_0^\infty \frac{h(x)\mathbb{P}(\hat{Y}_n - Y > x)}{\mathbb{P}(G > x)}dx < C\int_0^\infty \frac{h(x)\exp\left(-x\right)}{1-F(x)}dx < \infty \tag{5.42}$$

Hence, if the bounded integral condition holds, then, the integral of the ratio between the error probability and tail probability of the perturbation is also bounded. This condition tells us that the tail probability of perturbation must decrease slower than the estimator's tail probability to overcome the error of the estimator. For example, if the estimator misclassifies an optimal action due to the heavy-tailed noise, to overcome this situation by exploring other actions, the sampled perturbation $G_{t,a}$ must be greater than the sampled noise $\epsilon_{t,a}$. Otherwise, the perturbation method keeps selecting the overestimated sub-optimal action. Finally, the log-concavity is required to derive the lower bound. Based on Assumption 6, we can derive the following regret bounds of the APE^2.

Theorem 30. *Assume that the p-th moment of rewards is bounded by a constant $\nu_p < \infty$, $\hat{r}_{t,a}$ is a p-robust estimator of (5.38) and $F(x)$ satisfies Assumption 6. Then, $\mathbb{E}\left[\mathcal{R}_T\right]$ of APE^2 is bounded as*

$$\mathbb{E}\left[\mathcal{R}_T\right] \leq O\Bigg(\sum_{a\neq a^\star} \frac{C_{p,\nu_p,F}}{\Delta_a^{\frac{1}{p-1}}} + \frac{(6c)^{\frac{p}{p-1}}}{\Delta_a^{\frac{1}{p-1}}}\left[-F^{-1}\left(\frac{c^{\frac{p}{p-1}}}{T\Delta_a^{\frac{p}{p-1}}}\right)\right]_+^{\frac{p}{p-1}} \\ + \frac{(3c)^{\frac{p}{p-1}}}{\Delta_a^{\frac{1}{p-1}}}\left[F^{-1}\left(1-\frac{c^{\frac{p}{p-1}}}{T\Delta_a^{\frac{p}{p-1}}}\right)\right]_+^{\frac{p}{p-1}} + \Delta_a\Bigg)$$

where $[x]_+ := \max(x, 0)$, $C_{p,\nu_p,F} > 0$ is a constant independent of T.

The proof consists of three parts. Similarly to [7, 29], we separate the regret into three partial sums and derive each bound. The first term is caused by an overestimation error of the estimator. The second term is caused due to an underestimation error of the perturbation. When the perturbation has a negative value, the perturbation makes the reward under-estimated and, hence, this event causes a sub-optimal decision. The third term is caused by an overestimation error due to the perturbation. One interesting result is that the regret caused by the estimation error is bounded by $C_{c,p,\nu_p,F}/\Delta_a^{1/(p-1)}$. The error probability of the proposed estimator decreases exponentially fast and this fact makes the regret caused by the estimation error is bounded by a constant, which does not depend on T. The constant $C_{c,p,\nu_p,F}$ is determined by the bounded integral condition. The lower bound of APE^2 is derived by constructing a counterexample as follows.

Theorem 31. *For* $0 < c < \frac{K-1}{K-1+2^{p/(p-1)}}$ *and* $T \geq \frac{c^{1/(p-1)}(K-1)}{2^{p/(p-1)}} \left|F^{-1}\left(1 - \frac{1}{K}\right)\right|^{p/(p-1)}$, *there exists a K-armed stochastic bandit problem where the regret of APE^2 is lower bounded by*

$$\mathbb{E}[\mathcal{R}_T] \geq \Omega\left(K^{1-1/p}T^{1/p}F^{-1}\left(1 - 1/K\right)\right).$$

The proof is done by constructing the worst case bandit problem whose rewards are deterministic. When the rewards are deterministic, no exploration is required, but, APE^2 unnecessarily explores sub-optimal actions due to the perturbation. In other words, the lower bound captures the regret of APE^2 caused by useless exploration. Note that both of the upper and lower bounds are highly related to the inverse CDF F^{-1}. In particular, its tail behavior is a crucial factor of the regret bound when T goes to infinity.

The perturbation-based exploration is first analyzed in [64] under sub-Gaussian reward assumption. Kim and Tewari [64] have provided the regret bound of a family of sub-Weibull perturbations and that of all perturbations with bounded support for sub-Gaussian rewards. Our analysis scheme extends the framework of [64] into two directions. First, we weaken the sub-Gaussian assumption to the heavy-tailed rewards assumption. Second, our analysis scheme includes a wider range of perturbations such as weibull, GEV, Gamma, Pareto, and Fréchet.

5.2.5 Regret Bounds for Specific Perturbations with Heavy-Tailed Rewards

The upper and lower regret bounds of various perturbations such as weibull, GEV, Gamma, Pareto, and Fréchet are introduced in the following corollaries.

Corollary 6. *Suppose G follows a Weibull distribution with a parameter $k \leq 1$ with $\lambda > 1$ with $c > 0$. Then, the problem dependent regret bound is*

$$\mathbb{E}\left[\mathcal{R}_T\right] \leq O\left(\sum_{a \neq a^\star} \frac{C_{c,p,\nu_p,F}}{\Delta_a^{\frac{1}{p-1}}} + \left(\frac{(3c\lambda)^p}{\Delta_a}\right)^{\frac{1}{p-1}} \left[\ln\left(\frac{T\Delta_a^{\frac{p}{p-1}}}{c^{\frac{p}{p-1}}}\right)\right]^{\frac{p}{k(p-1)}} + \Delta_a\right).$$

The problem independent regret bound is, $\mathbb{E}\left[\mathcal{R}_T\right] = \Theta\left(\lambda^{\frac{p}{p-1}} K^{1-\frac{1}{p}} T^{\frac{1}{p}} \ln(K)^{\frac{1}{k}}\right)$.

The minimum rate is achieved at $k = 1$, $\mathbb{E}\left[\mathcal{R}_T\right] = \Theta\left(K^{1-\frac{1}{p}} T^{\frac{1}{p}} \ln(K)\right)$.

Corollary 7. *Suppose G follows a generalized extreme value distribution with a parameter with $0 \leq \zeta < 1$ and $\lambda > 1$. Then, the problem dependent regret bound is*

$$\mathbb{E}\left[\mathcal{R}_T\right] \leq O\left(\sum_{a \neq a^\star} \frac{C_{c,p,\nu_p,F}}{\Delta_a^{\frac{1}{p-1}}} + 2\left(\frac{(6c\lambda)^p}{\Delta_a}\right)^{\frac{1}{p-1}} \ln_\zeta\left(\frac{T\Delta_a^{\frac{p}{p-1}}}{c^{\frac{p}{p-1}}}\right)^{\frac{p}{p-1}} + \Delta_a\right).$$

Let $\ln_\zeta(x) := \frac{x^\zeta - 1}{\zeta}$, *then, the problem independent regret bound is*

$$\Omega\left(K^{1-\frac{1}{p}} T^{\frac{1}{p}} \ln_\zeta(K)\right) \leq \mathbb{E}\left[\mathcal{R}_T\right] \leq O\left(K^{1-\frac{1}{p}} T^{\frac{1}{p}} \frac{\ln_\zeta\left(K^{\frac{2p-1}{p-1}}\right)^{\frac{p}{p-1}}}{\ln_\zeta(K)^{\frac{1}{p-1}}}\right).$$

The minimum rate is achieved at $\zeta = 0$, $\mathbb{E}\left[\mathcal{R}_T\right] = \Theta\left(K^{1-\frac{1}{p}} T^{\frac{1}{p}} \ln(K)\right)$.

Corollary 8. *Suppose G follows a Gamma distribution with a parameter $\alpha \geq 1$ and $\lambda \geq 1$. Then, the problem dependent regret bound is*

$$\mathbb{E}\left[\mathcal{R}_T\right] \leq O\left(\sum_{a \neq a^\star} \frac{C_{c,p,\nu_p,F}}{\Delta_a^{\frac{1}{p-1}}} + \left(\frac{(3\lambda\alpha c)^p}{\Delta_a}\right)^{\frac{1}{p-1}} \ln\left(\frac{\alpha T \Delta_a^{\frac{p}{p-1}}}{c^{\frac{p}{p-1}}}\right)^{\frac{p}{p-1}} + \Delta_a\right). \tag{5.43}$$

The problem independent regret bound is

$$\Omega\left(\lambda K^{1-\frac{1}{p}} T^{\frac{1}{p}} \ln(K)\right) \leq \mathbb{E}\left[\mathcal{R}_T\right] \leq O\left((\lambda\alpha)^{\frac{1}{p-1}} c K^{1-\frac{1}{p}} T^{\frac{1}{p}} \frac{\ln\left(\alpha K^{1+\frac{p}{p-1}}\right)^{\frac{p}{p-1}}}{\ln(K)^{\frac{1}{p-1}}}\right). \tag{5.44}$$

The minimum rate is achieved at $\alpha = 1$, $\mathbb{E}\left[\mathcal{R}_T\right] = \Theta\left(K^{1-\frac{1}{p}} T^{\frac{1}{p}} \ln(K)\right)$.

Corollary 9. *Suppose G follows a Pareto distribution with a parameter $\alpha > \frac{p^2}{p-1}$ and $\lambda \geq \alpha$. Then, the problem dependent regret bound is*

$$\mathbb{E}\left[\mathcal{R}_T\right] \leq O\left(\sum_{a \neq a^\star} \frac{C_{c,p,\nu_p,F}}{\Delta_a^{\frac{1}{p-1}}} + \left(\frac{(3\lambda c)^p}{\Delta_a}\right)^{\frac{1}{p-1}} \left[\frac{T \Delta_a^{\frac{p}{p-1}}}{c^{\frac{p}{p-1}}}\right]^{\frac{p}{\alpha(p-1)}} + \Delta_a\right). \tag{5.45}$$

For $\lambda = \alpha$, the problem independent regret bound is

$$\Omega\left(\alpha K^{1-\frac{1}{p}+\frac{1}{\alpha}} T^{\frac{1}{p}}\right) \leq \mathbb{E}\left[\mathcal{R}_T\right] \leq O\left(\alpha^{1+\frac{p^2}{\alpha(p-1)^2}} K^{1-\frac{1}{p}+\frac{1}{\alpha(p-1)}} T^{\frac{1}{p}}\right). \tag{5.46}$$

For $K > \exp\left(\frac{p^2}{p-1}\right)$, the minimum rate is achieved at $\alpha = \ln(K)$, $\mathbb{E}\left[\mathcal{R}_T\right] = \Theta\left(K^{1-\frac{1}{p}} T^{\frac{1}{p}} \ln(K)\right)$.

Corollary 10. *Suppose G follows a Fréchet distribution with a parameter with $\alpha > \frac{p^2}{p-1}$ and $\lambda \geq \alpha$. Then, the problem dependent regret bound is*

$$\mathbb{E}\left[\mathcal{R}_T\right] \leq O\left(\sum_{a \neq a^\star} \frac{C_{c,p,\nu_p,F}}{\Delta_a^{\frac{1}{p-1}}} + \left(\frac{(3c\lambda)^p}{\Delta_a}\right)^{\frac{1}{p-1}} \left[\frac{T \Delta_a^{\frac{p}{p-1}}}{c^{\frac{p}{p-1}}}\right]^{\frac{p}{\alpha(p-1)}} + \Delta_a\right). \tag{5.47}$$

For $\lambda = \alpha$, the problem independent regret bound is

$$\Omega\left(\alpha K^{1-\frac{1}{p}+\frac{1}{\alpha}} T^{\frac{1}{p}}\right) \leq \mathbb{E}\left[\mathcal{R}_T\right] \leq O\left(\alpha^{1+\frac{p^2}{\alpha(p-1)^2}} K^{1-\frac{1}{p}+\frac{p^2}{\alpha(p-1)^2}} T^{\frac{1}{p}}\right). \quad (5.48)$$

For $K > \exp\left(\frac{p^2}{p-1}\right)$, the minimum rate is achieved at $\alpha = \ln(K)$, $\mathbb{E}\left[\mathcal{R}_T\right] = \Theta\left(K^{1-\frac{1}{p}} T^{\frac{1}{p}} \ln(K)\right)$.

We analyze the regret bounds of various perturbations including Weibull, Gamma, Generalized Extreme Value (GEV), Pareto, and Fréchet distributions. We first compute the gap-dependent regret bound using Theorem 30 and compute the gap-independent bound based on the gap-independent regret bound. We introduce corollaries of upper and lower regret bounds for each perturbation and also provide specific parameter settings to achieve the minimum gap-independent regret bound. From Table 5.1, we can observe that all perturbations we consider have the same gap-independent bound $\Theta\left(K^{1-1/p}T^{1/p}\ln(K)\right)$ while their gap-dependent bounds are different. Hence, the proposed method can achieve the $O(T^{1/p})$ with respect to T under heavy-tailed reward assumption, while the upper bound has the sub-optimal factor of $\ln(K)$ which caused by $F^{-1}(1-1/K)$ of the general lower bound. However, we emphasize that K is finite and T is much bigger than K in many cases, thus, $\ln(K)$ can be ignorable as T increases. For all perturbations, the gap-dependent bounds in Table 5.1 are proportional to two common factors $A_{c,\lambda,a} := ((3c\lambda)^p/\Delta_a)^{\frac{1}{p-1}}$ and $B_{c,a} := (\Delta_a/c)^{p/(p-1)}$ where $\Delta_a = r_{a^\star} - r_a$ and c is a constant in $\beta_{t,a}$. Note that, if Δ_a is sufficiently small or c is sufficiently large, $A_{c,\lambda,a}$ is the dominant term over $B_{c,a}$. We can see that the gap-dependent bounds increase as Δ_a decreases since $A_{c,\lambda,a}$ is inversely proportional to Δ_a. Similarly, as c increases, the bounds also increase. Intuitively speaking, the less Δ_a, the more exploration is needed to distinguish an optimal action from sub-optimal actions and, thus, the upper bound increases. Similarly,

Dist. on G	Prob. Dep. Bnd. $O(\cdot)$	Prob. Indep. Bnd. $O(\cdot)$	Low. Bnd. $\Omega(\cdot)$	Opt. Params.	Opt. Bnd. $\Theta(\cdot)$
Weibull	$\sum_{a\neq a^\star} A_{c,\lambda,a} \left(\ln\left(B_{c,a}T\right)\right)^{\frac{p}{k(p-1)}}$	$C_{K,T}\ln\left(K\right)^{\frac{1}{k}}$	$C_{K,T}\ln\left(K\right)$	$k=1, \lambda \geq 1$	$K^{1-1/p}T^{1/p}\ln\left(K\right)$
Gamma	$\sum_{a\neq a^\star} A_{c,\lambda,a}\alpha^{p/(p-1)}\ln\left(B_{c,a}T\right)^{p/(p-1)}$	$C_{K,T}\frac{\ln\left(\alpha K^{1+p/(p-1)}\right)^{p/(p-1)}}{\ln(K)^{\frac{1}{p-1}}}$	$C_{K,T}\ln\left(K\right)$	$\alpha=1, \lambda\geq 1$	
GEV	$\sum_{a\neq a^\star} A_{c,\lambda,a}\ln_\zeta\left(B_{c,a}T\right)^{p/(p-1)}$	$C_{K,T}\frac{\ln_\zeta\left(K^{\frac{2p-1}{p-1}}\right)^{p/(p-1)}}{\ln_\zeta(K)^{\frac{1}{p-1}}}$	$C_{K,T}\ln_\zeta\left(K\right)$	$\zeta=0, \lambda\geq 1$	
Pareto	$\sum_{a\neq a^\star} A_{c,\lambda,a}\left[B_{c,a}T\right]^{\frac{p}{\alpha(p-1)}}$	$C_{K,T}\alpha^{1+\frac{p^2}{\alpha(p-1)^2}}K^{\frac{1}{\alpha(p-1)}}$	$C_{K,T}\alpha K^{\frac{1}{\alpha}}$	$\alpha=\lambda=\ln(K)$	
Fréchet	$\sum_{a\neq a^\star} A_{c,\lambda,a}\left[B_{c,a}T\right]^{\frac{p}{\alpha(p-1)}}$	$C_{K,T}\alpha^{1+\frac{p^2}{\alpha(p-1)^2}}K^{\frac{1}{\alpha(p-1)}}$	$C_{K,T}\alpha K^{\frac{1}{\alpha}}$	$\alpha=\lambda=\ln(K)$	

Table 5.1: Regret Bounds of Various Perturbations. Dist. means a distribution, Prob. Dep. (or Indep.) Bnd. indicates a gap-dependent (or independent) bound, Low. Bnd. means a lower bound, Opt. Params. indicates optimal parameters to achieve an optimal bound, and Opt. Bnd. indicates the optimal bound. $O(\cdot)$ is an upper bound, $\Omega(\cdot)$ is a lower bound, and $\Theta(\cdot)$ is a tight bound, respectively. For the simplicity of the notation, we define $A_{c,\lambda,a} := \left((3c\lambda)^p/\Delta_a\right)^{\frac{1}{p-1}}$, $B_{c,a} := (\Delta_a/c)^{p/(p-1)}$, and $C_{K,T} := K^{1-1/p}T^{1/p}$.

increasing the parameter c leads to more exploration since the magnitude of $\beta_{t,a}$ increases. Hence, the upper bound increases.

From Table 5.1, we can categorize the perturbations based on the order of the gap-dependent bound with respect to T. The gap-dependent bound of Weibull and Gamma shows the logarithmic dependency on T while that of Pareto and Fréchet has the polynomial dependency on T. The gap-dependent regret bound of GEV shows the polynomial dependency since $\ln_\zeta(T)$ is a polynomial of T, but, for $\zeta = 0$, it has the logarithmic dependency since $\ln_\zeta(T)|_{\zeta=0} = \ln(T)$. Furthermore, both Pareto and Fréchet distributions have the same regret bound since their $F^{-1}(x)$ has the same upper and lower bounds. For gap-independent bounds, all perturbations we consider achieve the optimal rate $O(T^{1/p})$, but, the extra term dependent on K appears. Similarly to the case of the gap-dependent bounds, the sub-optimal factor of Weibull, Gamma, and GEV perturbations is proportional to the polynomial of $\ln(K)$, while that of Pareto and Fréchet is proportional to the polynomial of K.

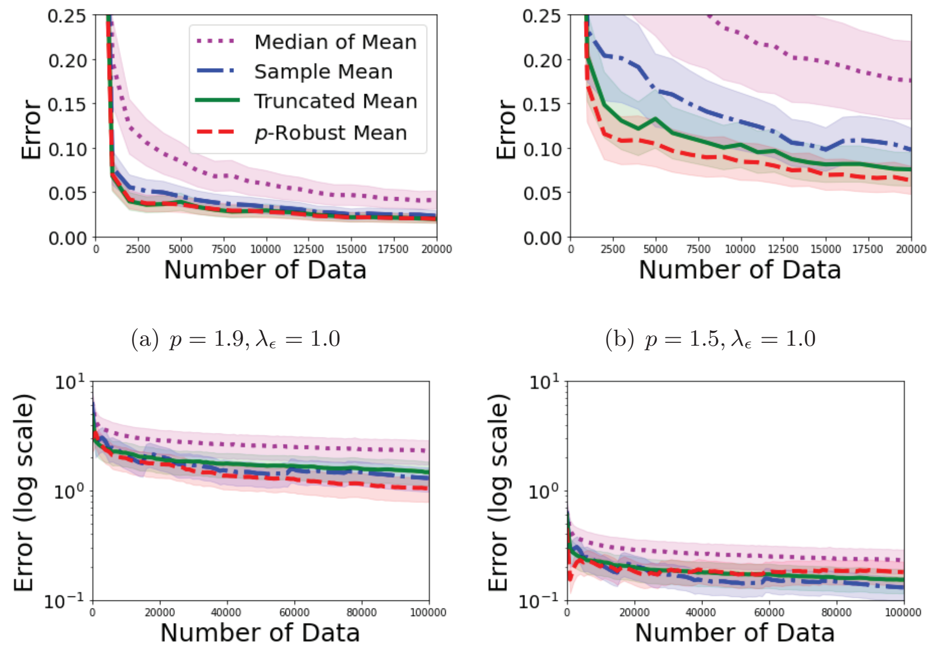
Compared to robust UCB, all perturbation methods have better gap-independent bound, but, the superiority of the gap-dependent bound can vary depending on Δ_a. In particular, the gap-dependent bound of Weibull, Gamma, and GEV ($\zeta = 0$) follows $\ln(\Delta_a^{p/(p-1)}T)^{p/(p-1)}/\Delta_a^{1/(p-1)}$ while that of robust UCB follows $\ln(T)^{p/(p-1)}/\Delta_a^{1/(p-1)}$. Hence, if Δ_a is large, then, $\ln(T)$ dominates $\ln(\Delta_a^{p/(p-1)})$ and it leads that robust UCB can have a smaller regret bound since $\ln(T) < \ln(T)^{p/(p-1)}$. On the contrary, if Δ_a is sufficiently small, Weibull, Gamma, and GEV ($\zeta = 0$) perturbations can have a smaller regret bound than robust UCB since $\ln(\Delta_a^{p/(p-1)})$ is a negative value for $\Delta_a \ll 1$ and reduces the regret bound of the perturbation methods dominantly. This property makes it available that perturbation methods achieve the optimal minimax regret bound with respect to

T while robust UCB has the sup-optimal gap-independent bound.

5.2.6 Experiments

Convergence of Estimator We compare the p-robust estimator with other estimators including truncated mean, median of mean, and sample mean. To make a heavy-tailed noise, we employ a Pareto random variable z_t with parameters α_ϵ and λ_ϵ. Then, a noise is defined as $\epsilon_t := z_t - \mathbb{E}[z_t]$ to make the mean of the noise zero. In simulation, we set a true mean $y = 1$ and $Y_t = y + \epsilon_t$ is observed. We measure the error $|\hat{Y}_t - y|$. Note that, for all $p < \alpha_\epsilon$, the bound on the p-th moment is given as $\nu_p \leq |1 - \mathbb{E}[z_t]|^p + \alpha_\epsilon \lambda_\epsilon^p / (\alpha_\epsilon - p)$. Hence, we set $\alpha_\epsilon = p + 0.05$ to bound the p-th moment. We conduct the simulation for $p = 1.1, 1.5, 1.9$ with $\lambda_\epsilon = 1.0$ and for $p = 1.1$, we run an additional simulation with $\lambda_\epsilon = 0.1$. The entire results are shown in Fig. 5.1.

From Fig. 5.1(a), 5.1(b), 5.1(c), and 5.1(d), we can observe the effect of p. Since the smaller p, the heavier the tail of noise, the error of all estimators increases as p decreases when the same number of data is given. Except for the median of mean, robust estimators show better performance than a sample mean. In particular, for $p = 1.9, 1.5, 1.1$ with $\lambda_\epsilon = 1.0$, the proposed method shows the best performance. For $p = 1.1$ with $\lambda_\epsilon = 0.1$, the proposed method shows a comparable accuracy to the truncated mean even if our method does not employ the information of ν_p. From Fig. 5.1(c) and 5.1(d), we can observe the effect of ν_p for fixed $p = 1.1$. As λ_ϵ decreases, ν_p decreases. When $\lambda_\epsilon = 0.1$, since the truncated mean employs ν_p, the truncated mean shows better performance than the proposed estimator, but, the proposed estimator shows comparable performance even though it does not employ ν_p. We emphasize that these results show the clear benefit of the proposed estimator since our estimator does not employ ν_p, but, generally show



(a) $p = 1.9, \lambda_\epsilon = 1.0$

(b) $p = 1.5, \lambda_\epsilon = 1.0$

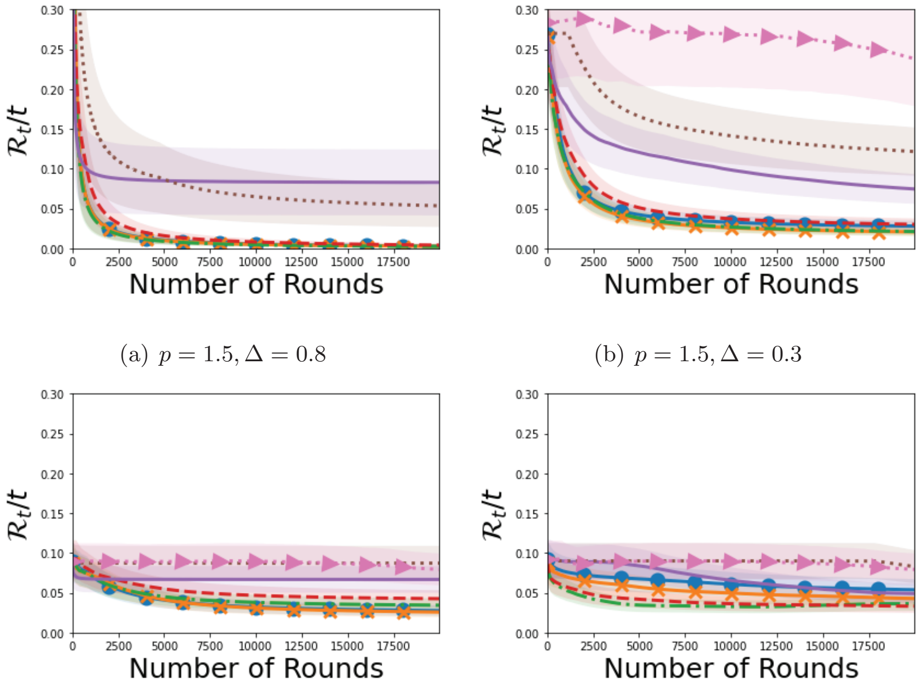
(c) $p = 1.1, \lambda_\epsilon = 1.0$

(d) $p = 1.1, \lambda_\epsilon = 0.1$

Figure 5.1: Error of Robust Estimators with Pareto Noises. p is the maximum order of the bounded moment. λ_ϵ is a scale parameter of the noise. The lower p or the larger λ_ϵ, the heavier the tail of noise. The solid line is an averaged error over 60 runs and a shaded region shows a quarter standard deviation.

faster convergence speed.

Multi-Armed Bandits with Heavy-Tailed Rewards We compare APE^2 with robust UCB [26] and DSEE [136]. Note that an empirical comparison with GSR [60] is omitted here and can be found in the supplementary material since GSR shows poor performance in terms of the cumulative regret as mentioned in Section 6.1.3. For APE^2, we employ the optimal hyperparameter of perturbations shown in Table 5.1. Note that GEV with $\zeta = 0$ is a Gumbel distribution and Gamma with $\alpha = 1$ (or Weibull with $k = 1$) is an Exponential distribution and λ of Gumbel and Exponential is set to be one. Thus, we compare four perturba-



(a) $p = 1.5, \Delta = 0.8$

(b) $p = 1.5, \Delta = 0.3$

(c) $p = 1.5, \Delta = 0.1$

(d) $p = 1.1, \Delta = 0.1$

Figure 5.2: Time-Averaged Cumulative Regret. p is the maximum order of the bounded moment of noises. Δ is the gap between the maximum and second best reward. For $p = 1.5$, $\lambda_\epsilon = 1.0$ and for $p = 1.1$, $\lambda_\epsilon = 0.1$. The solid line is an averaged error over 40 runs and a shaded region shows a quarter standard deviation.

tions: Gumbel, Exponential, Pareto, and Fréchet. For APE^2 and DSEE, the best hyperparameter is found by using a grid search. For robust UCB, since the original robust UCB consistently shows poor performance, we modify the confidence bound by multiplying a scale parameter c and optimize c using a grid search. Furthermore, robust UCB employ the truncated mean estimator since the median of mean shows poor performance for the previous simulation. All hyperparameters can be found in the supplementary material. We synthesize a MAB problem that has a unique optimal action and all other actions are sub-optimal. The optimal mean reward is set to one and $1-\Delta$ is assigned for the sub-optimal actions where $\Delta \in (0,1]$ determines a gap. By controlling Δ, we can measure the effect of the gap. Similarly to the previous simulation, we add a heavy-tailed noise using the Pareto distribution. We prepare six simulations by combining $\Delta = 0.1, 0.3, 0.8$ and $p = 1.5, p = 1.1$. A scale parameter λ_ϵ of noise is set to be 0.1 for $p = 1.1$ and 1.0 for $p = 1.5$, respectively. We measure the time averaged cumulative regret, i.e., $\mathcal{R}_t/t$, for 40 trials.

The selective results are shown in Fig. 5.2 and all results can be found in the supplementary material. First, the perturbation methods generally outperform robust UCB. For $p = 1.5$ and $\Delta = 0.8$, from Fig. 5.2(a), we can observe that all methods converge rapidly at a similar rate. While perturbation methods show better results, performance difference between robust UCB and perturbation methods is marginal. However, when Δ is sufficiently small such as $\Delta = 0.3, 0.1$, Fig. 5.2(b) and 5.2(c) show that perturbation methods significantly outperform robust UCB. In particular, Gumbel and Exponential perturbations generally show better performance than other perturbations. We believe that the results on Δ support the gap-dependent bound of Table 5.1. As mentioned in Section 5.2.5, when Δ decreases, Gumbel and Exponential perturbations show a faster convergence speed than robust UCB. In addition, Fig. 5.2(d) empirically proves the benefit of the perturbation methods. For $p = 1.1$ with $\lambda_\epsilon = 0.1$, Fig. 5.1(d) shows that the proposed estimator converges slightly slower than the truncated mean, however, in the MAB setting, APE^2 convergences significantly faster than robust UCB as shown in Fig. 5.2(d). From this observation, we can conclude that perturbation methods more efficiently explore an optimal action than robust UCB despite of the weakness of the proposed estimator for $p = 1.1$. Unlikely to other methods, DSEE consistently shows poor performance. While APE^2 and robust UCB can stop exploring sub-optimal actions if confidence bound or $\beta_{t,a}$ is sufficiently reduced, DSEE suffers from the lack of adaptability since DSEE is scheduled to choose every action uniformly and infinitely.

5.2.7 Summary

We have proposed novel p-robust estimator which can handle heavy-tailed noise distributions which does not require prior knowledge about the bound on the p-th moment of rewards. By using the proposed estimator, we also proposed an adaptively perturbed exploration with a p-robust estimator (APE^2) and proved that APE^2 has better regret bound than robust UCB. In simulations, we empirically show that the proposed estimator outperforms the existing robust estimators and APE^2 outperforms robust UCB when the gap is small. We have theoretically and empirically demonstrated that APE^2 can overcome rewards that are corrupted by heavy-tailed noises, making APE^2 an appropriate solution for many practical problems, such as online classification, online learning of a recommendation system, and reinforcement learning.

Chapter 6

Inverse Reinforcement Learning with Negative Demonstrations

Reinforcement learning (RL) has been widely used to learn behaviors to perform complex tasks in robotics [68]. RL aims to find the optimal behavior which maximizes the expected sum of rewards during the execution phase. A reward function indicates the one step performance measure about each control at each situation. In order to successfully learn a desirable behavior, the reward function must be elaborately designed to express the given task.

However, in some tasks such as driving a car [1], inverted helicopter flight[2], and socially adaptive path planning [65], it is difficult to design a proper reward function that accurately generates the desired behaviors. It is more natural to learn the desirable behaviors performing such tasks by imitating expert's demonstrations. This problem of extracting the underlying reward function from a sequence of demonstrations is often called inverse reinforcement learning (IRL)[92] or inverse optimal control (IOC) [10]. IRL aims to find the reward function which best explains demonstrations by experts. A key assumption of IRL is that experts follow the optimal policy induced by the underlying reward function, hence, the main idea of solving IRL is to find a reward function that makes experts' behaviors (near) optimal. The reward function learned by IRL is further used to obtain the desired behaviors by solving a usual reinforcement learning problem.

Since demonstrations of experts are often distributed near high reward regions, the resulting reward function learned by IRL cannot approximate low reward regions accurately. This phenomenon was intensively investigated in [109, 108]. The authors argued that the lack of demonstrations of *what to do* in the critical situations will lead to unsatisfactory performance or fatal failure. For example, when learning how to drive, an autonomous vehicle occasionally encounters a risky situation, e.g. heading towards the side of the road. In order to avoid a catastrophic situation, the autonomous vehicle should recover back to the center of the road. However, such recovery behavior rarely appears in demonstrations from a good driver. In [109], Ross and Bargnell tackled this problem via continuous interaction with experts. However, it is not practical to rely on experts frequently.

To handle lack of demonstrations near low reward regions, we incorporate demonstrations about both *what to do* and *what not to do*. As demonstrations about *what not to do* will be often distributed near low reward regions, we can obtain information to avoid catastrophic failures from such demonstrations. Demonstrations of failures had been considered before. In [118], a reward function was modeled as a linear combination of features and a policy was learned by making the learned policy maximally different from the failed demonstrations using maximum entropy IRL [152]. However, since the notion of *failure* only depends on the result of a demonstration, a failed demonstration does not exactly provide the information about *what not to do*, which is further discussed in Section 6.1.1.

6.1 Leveraged Gaussian Processes Inverse Reinforcement Learning

In this section, we propose a novel inverse reinforcement learning algorithm with leveraged Gaussian processes that can incorporate examples of both *what to do* (positive demonstrations) and *what not to do* (negative demonstrations). We model a reward function using a leveraged Gaussian process (LGP) [32], which is capable of modeling a complex nonlinear function. It allows us to accurately estimate the nonlinear structure of a reward function, compared to previous approaches. To mathematically define positive and negative demonstrations, we introduce a novel generative model of a demonstrator, which can sample both positive and negative demonstrations using the same reward function. Our generative model incorporates an additional parameter, called *proficiency*, which can vary continuously from -1 to $+1$, such that a positive (or, respectively, negative) proficiency indicates a positive (or, respectively, negative) demonstration. Hence, in our problem, a demonstration consists of its proficiency value as well as a sequence of state-action pairs. The proposed IRL method finds a reward function such that positive demonstrations have higher values and negative demonstrations have lower values. We extensively validate the performance of the proposed method in terms of accuracy and sample efficiencies. In simulations, the proposed method is more efficient and can approximate the underlying reward function more accurately using a fewer number of demonstrations than existing methods. Moreover, it shows that the use of negative demonstrations is better than simply using only positive demonstrations, illustrating the benefits of using negative

	Pos Demo	Pos and Neg Demo
Margin based model	[1, 105, 92, 106, 107]	[137]
Probabilistic model	[103, 152, 38, 76, 150, 31, 30, 144, 21, 6]	[11, 118], Ours

Table 6.1: Classification of IRL algorithms

examples.

The remainder of this chapter is structured as follows. In Section 6.1.1, related work is discussed. In Section 6.1.2, the Markov decision process (MDP), Gaussian process IRL, and LGP are introduced. In Section 6.1.6, the new expert's model and the proposed learning algorithm are explained. A simulation study is discussed in Section 6.1.7.

6.1.1 Related Work

Recently, a number of IRL methods have been proposed [1, 105, 92, 106, 107, 137, 103, 152, 38, 76, 150, 31, 30, 144, 21, 6, 11, 118]. They can be separated into four different categories based on two criteria. The first criterion is the formulation of problem: margin based or probabilistic model based. The second is the capability of considering both positive and negative demonstrations or not. Many existing algorithms consider only positive demonstrations and a handful of approaches utilize both types of demonstrations. The classification of state-of-the-art IRL algorithms, including ours, is summarized in Table 6.1.

A margin based method maximizes the margin between the value of the expert's policy and all other policies [1, 105, 92, 106, 107]. The margin based algorithms generally assume that the reward function is a linear combination of features. Since many formulations for the margin based methods are quadratic in parameters, quadratic programming (QP) is widely used to solve the problem. In [1], Abbeel and Ng proposed an apprenticeship learning (AL) algorithm, which maximizes the margin between the expert's policy and randomly sampled policies. In [105], Ratliff et al. proposed the maximum margin planning (MMP) algorithm where Bellman-flow constraints are utilized to consider the margin between the experts' policy and all other policies. MMP was mainly motivated from the structured support vector machine (SVM) [128]. In [107], Ratliff et al. extended MMP to allow learning a nonlinear reward function and the method is called learning to search (LEARCH).

A probabilistic model based method first defines a probability distribution of expert's demonstrations and optimize the parameter of the distribution [103, 152, 38, 76, 150, 31, 30, 144]. To define the probability on a trajectory of state-action pairs, many probabilistic IRL algorithms utilize a stochastic policy. The stochastic policy model was first utilized in [103, 152] in order to handle the inconsistency of expert's policy. Ziebart et al. [152] proposed maximum entropy inverse reinforcement learning (MEIRL) using the principle of maximum entropy to handle ambiguity issues of IRL, where the efficient way to compute the gradient of the likelihood of demonstrations is also proposed. Ramachandran et al. [103] proposed Bayesian inverse reinforcement learning (BIRL), where the Bayesian probabilistic model over demonstrations is defined and solved using a Metropolis-Hastings (MH) method. A method for estimating an optimal value function (OptV) is proposed in [38], where OptV is a linearly solvable approximation of a standard Markov decision process. We also note that [76, 150, 31, 30, 144] are variants based on [103, 152]. Gaussian process inverse reinforcement learning (GPIRL) was proposed in [76], where the reward function is represented as a sparse Gaussian process, which can express a nonlinear reward function in a feature space. Robust Bayesian inverse reinforcement learning (RBIRL) was proposed in [150], which is an extension of BIRL to handle noisy demonstrations. RBIRL can automatically identify and remove a noise in demonstrations using the expectation maximization (EM) algorithm. Choi et al. [31] proposed hierarchical Bayesian inverse reinforcement learning (HBIRL), which builds a hierarchy on the graphical model of BIRL. Choi et al. [30] proposed a nonparametric Bayesian feature constructing method for IRL (NPBFIRL) to identify useful composite features for learning a reward function. Wulfmeier et al. [144] proposed maximum entropy deep inverse reinforcement learning (MEDIRL), where the reward function is modeled by a neural network and is learned by maximizing the log likelihood of demonstrations using the method from [152].

Despite successful advances in IRL, the demonstration acquisition is still an open issue. Generating demonstrations by experts can be often an expensive process. To handle this problem, [137, 118, 11] are proposed to utilize inexpert demonstrations. In [137, 11], the authors considered unlabeled demonstrations, for which we do not know whether they are actually generated by an expert or not. [137] proposed semi-supervised apprenticeship learning (SSAL), which is a natural extension of [1], to utilize both labeled and unlabeled demonstrations. Since apprenticeship learning maximizes the margin between the expert's policy and others using a maximum margin method, SSAL treats inexpert demonstrations as negatively labeled data. Also [11] proposed maximum entropy semi-supervised IRL (MESSIRL), which is an extension of MEIRL [152], where demonstrations of multiple qualities are used. In [11], unlabeled demonstrations, which are dissimilar to expert's demonstrations, are filtered using a penalty function and the remaining demonstrations are used to learn the reward function based on similarities to expert's demonstrations.

While SSAL [137] and MESSIRL [11] mainly focused on classifying unlabeled demonstrations, [118] focused more on utilizing failed demonstrations. In [118], Shiarlis et al. proposed inverse reinforcement learning from failure (IRLfF). IRLfF reformulated MEIRL [152] with new constraints that require the learned policy to maximally differ from failed demonstrations. However, failed demonstrations do not exactly provide the information about *what not to do*, since the failed demonstration may contain some level of desirable behavior while its outcome is failure. In this section, we focus on representing negative demonstrations, i.e., *what not to do*, using a novel demonstrator model.

6.1.2 Background

Markov Decision Processes and a Stochastic Policy

A common method to formulate a skill learning problem is a Markov decision process (MDP). An MDP can be characterized by a tuple $\mathbf{M} = \{\mathbf{S}, \mathbf{F}, \mathbf{A}, \mathbf{T}, \gamma, \mathbf{r}\}$, where $\mathbf{S}$ is the state space, $\mathbf{F}$ is the corresponding feature space, $\mathbf{A}$ is the action space, $\mathbf{T}(s'|s,a)$ is the transition probability from $s \in \mathbf{S}$ to $s' \in \mathbf{S}$ by taking an action $a \in \mathbf{A}$, γ is a discount factor, and $\mathbf{r}$ is the reward function. A conventional skill learning problem such as reinforcement learning can be solved by finding the optimal policy which maximizes the expected discounted reward sum. For inverse reinforcement learning (IRL), the problem is expressed as $\mathbf{M}/\mathbf{r}$ with experts' demonstrations $\mathcal{D} = \{\zeta_1, \ldots, \zeta_N\}$, where ζ_i is a sequence of state-action pairs, i.e., $\zeta_i = \{(s_{i,0}, a_{i,0}), \ldots, (s_{i,T}, a_{i,T})\}$. Solving IRL can be interpreted as recovering the underlying reward function $\mathbf{r}$, which best explains the demonstrations of an expert with the assumption that the expert always obeys the optimal policy. In practice, however, demonstrations from experts can be occasionally inconsistent with each other. This motivates the incorporation of uncertainties in the policy function, i.e., a stochastic policy. A stochastic policy model has been widely used in IRL problems [103, 152, 38, 76, 150, 31]. In a stochastic policy model, the probability of choosing action a at state s is exponentially proportional to the state-action value function $Q(s, a)$ as we mentioned in Chapter 2.1.4.

Under this policy, the log likelihood of demonstration ζ under $\mathbf{r}$ can be written as

$$\log P(\zeta|\mathbf{r}) = \sum_{t=0}^{T} [Q(s_t, a_t) - V(s_t)] + C, \tag{6.1}$$

where C is a constant. The reward function $\mathbf{r}$ is often represented as a linear combination of a set of provided features [152].

6.1.3 Gaussian Process Regression

In probability theory, a random process is called a wide-sense stationary process if its mean function $m(\mathbf{x})$ is constant and a covariance function $k(\mathbf{x}\,, \mathbf{x}')$ is a function of a displacement vector $k(\mathbf{x}\,, \mathbf{x}') = k_S(\mathbf{x} - \mathbf{x}')$. If a covariance function is a function of the distance between two inputs, $k(\mathbf{x}\,, \mathbf{x}') = k_I(||\mathbf{x} - \mathbf{x}'||)$, such covariance function is called an *isotropic* kernel function.

Gaussian process regression (GPR) is called stationary GPR if a stationary kernel function is used. The assumption behind using stationary GPR is that the function of our interest has the homogeneous smoothness. However, this may not be always the case, and in [95], the authors proposed non-stationary Gaussian process regression using the following non-stationary kernel function:

$$k(\mathbf{x}i, \mathbf{x}j) = \frac{2^{\frac{p}{2}} |\Sigma_i|^{\frac{1}{4}} |\Sigma_j|^{\frac{1}{4}}}{|\Sigma_i + \Sigma_j|^{\frac{1}{2}}} \exp\left(-(\mathbf{x}i - \mathbf{x}j)^T \bar{\Sigma}^{-1} (\mathbf{x}i - \mathbf{x}j)\right). \tag{6.2}$$

With this non-stationary kernel function, we can manually control the local smoothness by additional parameters: Σ_i and Σ_j. Since a kernel matrix works as a covariance matrix in a Gaussian process, it must satisfy the positive semi-definite condition:

$$\sum_{i=1}^{N}\sum_{j=1}^{N} a_i a_j C(\mathbf{x}i, \mathbf{x}j) \geq 0, \tag{6.3}$$

for every $\mathbf{a} = [a_1 \ \dots \ a_N]^T \in \mathbb{R}^N$ and N.

Unfortunately, directly showing (6.3) for a given kernel function is not trivial. In [95], the authors proved the positive semi-definiteness of (6.2) using the following theorem by Higdon, Swall, and Kern (HSK) [54].

Theorem 32 (HSK Theorem)**.** *A covariance function $C(x_i, x_j)$ defined by*

$$C(\mathbf{x_i}, \mathbf{x_j}) = \int_{\mathbb{R}^d} K_{\mathbf{x_i}}(\mathbf{u}) K_{\mathbf{x_j}}(\mathbf{u}) d\mathbf{u}, \tag{6.4}$$

where $\mathbf{x_i}$, $\mathbf{x_j}$, $\mathbf{u} \in \mathbb{R}^d$, and $K_{\mathbf{x}}(\cdot)$ is a basis kernel function centered at $\mathbf{x}$, is positive semi-definite.

Using Theorem 32, one can avoid the difficulties in verifying the positive semi-definiteness of a kernel function. Deriving the non-stationary kernel function (6.2) can be easily done by setting $K_{\mathbf{x}}(\mathbf{u}) = \mathcal{N}(\mathbf{u}|\mathbf{x}, \Sigma)$. The integration over $\mathbb{R}^d$ can be handled using the properties of Gaussian distributions, in that the product of two Gaussian probability density functions is a pseudo-Gaussian.

However, it is still difficult to prove positive semi-definiteness of an arbitrarily given function as we have to find a basis kernel $K_{\mathbf{x}}(\mathbf{u})$ which forms the given function of interest as shown in (6.4). Another way of showing positive semi-definiteness is using the Bochner's theorem [20].

Theorem 33 (Bochner's Theorem)**.** *Let f be a bounded continuous function on $\mathbb{R}^d$. Then, f is positive semi-definite if and only if it is the (inverse) Fourier transform of a nonnegative and finite Borel measure μ, i.e.,*

$$f(\mathbf{x}) = \int_{\mathbb{R}^d} e^{i\mathbf{w}^T\mathbf{x}} \mu(d\mathbf{w}). \tag{6.5}$$

In particular, Theorem 33 states that if a Fourier transform of a function f is non-negative, then f is positive semi-definite.

Theorem 34. *If a bounded continuous function $f : \mathbb{R}^d \to \mathbb{R}$ is a (inverse) Fourier transform of a non-negative density function of a Borel measure μ, then f is positive semi-definite.*

Proof. Let $g(\mathbf{w}) \geq 0$ be a density of a Borel measure μ, i.e.,

$$\mu(E) = \int_E g(\mathbf{w}) d\mathbf{w}.$$

If we assume g is a Fourier transform of a bounded and continuous function f, f can be written as follows:

$$f(\mathbf{x}) = \int_{\mathbb{R}^d} e^{j\mathbf{w}^T\mathbf{x}} g(\mathbf{w}) d\mathbf{w}. \tag{6.6}$$

From (6.6), one can directly prove positive semi-definiteness of f as follows:

$$\begin{aligned}
\sum_{n=1}^{N}\sum_{m=1}^{N} a_n a_m^* f(\mathbf{x}_n - \mathbf{x}_m) &= \sum_{n=1}^{N}\sum_{m=1}^{N} a_n a_m^* \int_{\mathbb{R}^d} e^{j\mathbf{w}^T(\mathbf{x}_n - \mathbf{x}_m)} g(\mathbf{w}) d\mathbf{w} \\
&= \int_{\mathbb{R}^d} \sum_{n=1}^{N}\sum_{m=1}^{N} a_n a_m^* e^{j\mathbf{w}^T(\mathbf{x}_n - \mathbf{x}_m)} g(\mathbf{w}) d\mathbf{w} \\
&= \int_{\mathbb{R}^d} \left(\sum_{n=1}^{N} a_n e^{j\mathbf{w}^T\mathbf{x}_n}\right)\left(\sum_{m=1}^{N} a_m^* e^{-j\mathbf{w}^T\mathbf{x}_m}\right) g(\mathbf{w}) d\mathbf{w} \\
&= \int_{\mathbb{R}^d} \left\| \sum_{n=1}^{N} a_n e^{j\mathbf{w}^T\mathbf{x}_n} \right\|^2 g(\mathbf{w}) d\mathbf{w} \geq 0.
\end{aligned}$$

□

This theorem will be used in Section 6.1.4 to prove positive semi-definiteness of the proposed kernel function.

6.1.4 Leveraged Gaussian Processes

The non-stationarity of the kernel function (6.2) comes from the kernel's capability to adjust local smoothness by assigning Σ_i for each data sample $\mathbf{x}_i$. In this section, we focus on the leveraged non-stationary kernel function for a non-stationary Gaussian process that can vary the *leverage* of training data.

The motivation behind the development of this kernel function starts with a relatively simple question. *Is it possible to use negative examples in a regression framework?* As the goal of a traditional regression problem is to find a function (or a regressor) that can best *fit* given training data, $D = \{(\mathbf{x}, y)_i, i = 1, ..., N\}$, where $\mathbf{x}_i$ and y_i are input and output, respectively, it usually anchor the regressor to those input and output points.

In our regression formulation, *positive* data work as an attractive force making the regressor as close as possible to such points, and *negative* data work as a repulsive force making the regressor as far as possible from such points. Moreover, it can also vary the *leverage* of each training data from fully negative (-1) to fully positive ($+1$). In particular, when the *leverage* is 0, the corresponding data will have no effect on the regression.

Leveraged Kernel Function

In this section, we propose a leveraged kernel function that can be used in a novel regression framework which can incorporate both positive and negative training data. Furthermore, we prove that the proposed leveraged non-stationary kernel function satisfies the positive semi-definiteness (PSD) condition using kernel composite rules and Theorem 34.

Each training data has its *leverage* parameter which varies from -1 to $+1$, where -1 indicates a fully negative leverage and $+1$ indicates a fully positive leverage. Following is the proposed leveraged kernel function:

$$k(\mathbf{x}i, \mathbf{x}j) = (1 - |\gamma_i - \gamma_j|)\, k_{SE}(\mathbf{x}i,\, \mathbf{x}j), \tag{6.7}$$

where $k_{SE}(\mathbf{x}i, \mathbf{x}j) = \sigma_f^2 \exp\left(-\frac{1}{2}(\frac{||\mathbf{x}i - \mathbf{x}j||}{\sigma_x})^2\right)$, $\mathbf{x}i$ and $\mathbf{x}j$ are input points, γ_i and γ_j are leverage parameters of the i-th and j-th inputs, respectively, and $\theta = \{\sigma_f^2,\, \sigma_x^2\}$ are hyperparameters of the Gaussian process. For test (unseen) inputs, γ will be set to be 1.

Proposition 1. *The proposed leveraged kernel function (6.7) is a positive semi-definite function.*

Proof. Let us begin the proof with the basic properties of generating new kernel functions from existing valid kernel functions. Here, we assume that a valid kernel function satisfies the PSD property. Then the sum and product of two valid kernel functions are also valid [104].

One interesting property of a kernel function is that if $k(\mathbf{x_1}, \mathbf{x_1'})$ and $k(\mathbf{x_2}, \mathbf{x_2'})$ are valid over different spaces $\mathcal{X}_1$ and $\mathcal{X}_2$, then the tensor product $k(\mathbf{x}, \mathbf{x}') = k(\mathbf{x_1}, \mathbf{x_1'})k(\mathbf{x_2}, \mathbf{x_2'})$ is also a valid kernel function defined on the product space $\mathcal{X}_1 \times \mathcal{X}_2$. These properties can easily be verified using the definition of the PSD (see Chapter 4 in [104] for details).

Using these properties, we can decompose the kernel function (6.7) into two parts. The first term is

$$k(\gamma_i, \gamma_j) = (1 - |\gamma_i - \gamma_j|) \tag{6.8}$$

and the second term is

$$k_{SE}(\mathbf{x}i, \mathbf{x}j) = \sigma_f^2 \exp\left(-\frac{1}{2}\left(\frac{||\mathbf{x}i - \mathbf{x}j||}{\sigma_x}\right)^2\right). \tag{6.9}$$

The proposed leveraged kernel function (6.7) is a tensor product of two kernel functions, (6.8) and (6.9). As (6.9) is a well-known squared exponential (SE) kernel which satisfies the PSD, proving (6.8) is a PSD function with respect to every γ_i and γ_j in $[-1, 1]$ will complete the proof. Note that γ indicates the *leverage* and has values between -1 and 1.

Theorem 34 states that showing the non-negativeness of a Fourier transform of a stationary kernel, i.e. $k(x, x') = k(x - x')$, is equivalent to showing the positive semi-definiteness. Thus, showing the Fourier transform of

$$k(t = \gamma_i - \gamma_j) = 1 - |t| \tag{6.10}$$

is non-negative will complete the proof. Since the domain of (6.10) is $[-2, 2]$, without loss of generality, we can extend (6.10) to be a periodic function with period 4 outside the domain, i.e., $k(t + 4) = k(t)$.

As the extended function of (6.10) is a periodic even function, we can express (6.10) as a Fourier series as follows:

$$\begin{aligned} c_n &= \frac{1}{4}\int_{-2}^{2} k(t) \exp\left(\frac{-i\pi n t}{2}\right) dt \\ &= 2\frac{1-(-1)^n}{\pi^2 n^2} \\ &= \begin{cases} 0 & \text{for even } n, \\ 4/(n^2\pi^2) & \text{for odd } n. \end{cases} \end{aligned} \tag{6.11}$$

As the Fourier coefficient (6.11) is non-negative for all integers, by the Theorem 34, the kernel (6.10) is positive semi-definite, which completes the proof. □

The shape of the proposed leveraged kernel function is shown in Figure 6.1. We can see that between positive examples (and negative examples), the kernel function works as an ordinary squared-exponential kernel function. However, between positive and negative data, the correlation decreases as the distance between inputs decreases.

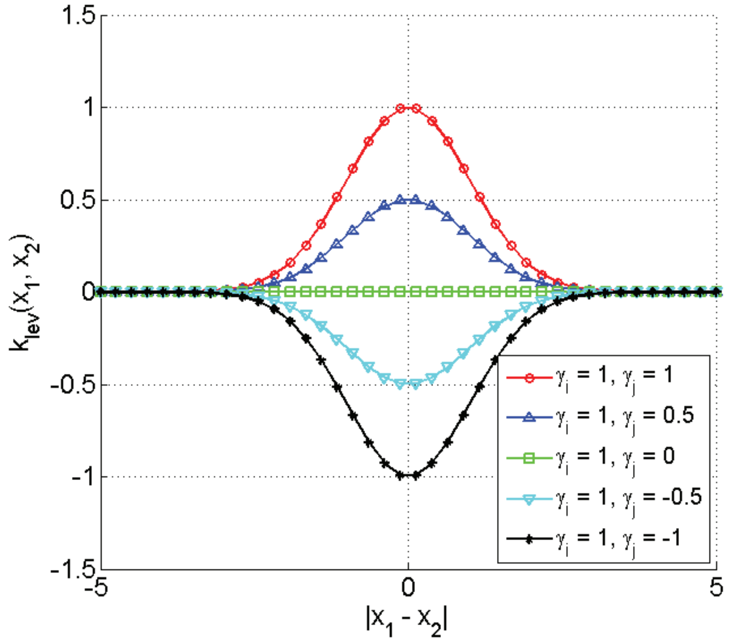
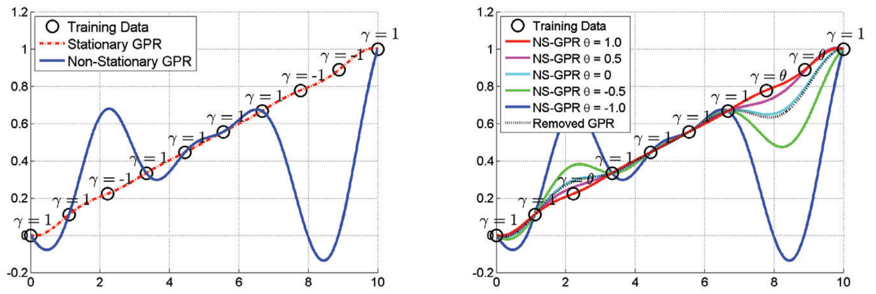


Figure 6.1: The proposed leveraged kernel function with different values of γ_i and γ_j.

As we assume that the *leverage* of unseen data is 1, the positive training data will work as the ordinary training data. However, as being close to the data with a negative *leverage* will lower the correlation, the resulting regressor will be far from such data.

Leveraged Non-Stationary Gaussian Process Regression

The non-stationary Gaussian process regression results using the proposed leveraged kernel function are shown in Figure 6.2(a). The leverages, γ_i, are shown at the top of each training data. As illustrated in Section 6.1.4, the training data with $\gamma = 1$ correspond with the ordinary positive training data and the training data with $\gamma = -1$ work as negative training data. In particular, for those with



(a)

(b)

Figure 6.2: (a) Ordinary Gaussian process regression using a squared exponential (SE) kernel function. (b) Gaussian process regression using the proposed leveraged kernel function.

$\gamma = 0$, such data will have no effect on the resulting regressor.

Figure 6.2(a) shows regression results of stationary GPR using a squared exponential (SE) kernel and non-stationary GPR using the proposed leveraged kernel (6.7). In particular, for non-stationary GPR, third, eighth, and ninth data points work as negative data ($\gamma = -1$) and the rest of the data work as positive data ($\gamma = 1$). As shown in Figure 6.2(a), the non-stationary Gaussian process regression tends to *anchor* to the positive data and *drift away* from the negative data.

Figure 6.2(b) shows how the non-stationary GPR varies with different leverage parameters γ. The removed GPR is the regression result using only the positive training data, i.e., $D = \{(x, y)_i | \, i = 1, 2, 4, 5, 6, 7, 10\}$. We can see that non-stationary GPR with $\theta = 0$ and removed GPR have similar results as the training data with $\gamma = 0$ have no effect to the regressor in the proposed non-stationary GPR.

We would like to note the relevance between proposed leveraged non-stationary Gaussian process and the non-stationary Gaussian process from [95]. In [95], the non-stationarity is introduced via the variance which indicates local smoothness. This value varies with the input space and thus an additional function for modeling varying smoothness is required. Heuristic and domain specific variance functions are often used in this regard [73]. Moreover, modeling these functions requires more computational load as the dimension of the input space gets larger. Similarly, the proposed method has non-stationarity by the additional leverage parameter γ. However, an additional function is not required as we explicitly assign $\gamma \in [-1, 1]$ to each training data and 1 for the test input.

6.1.5 Gaussian Process Inverse Reinforcement Learning

Gaussian process inverse reinforcement learning (GPIRL) was proposed in [76]. GPIRL uses the stochastic policy model and represents the reward function as a Gaussian process, where its structure is determined by its kernel function and hyperparameters θ. In order to apply Gaussian process regression (GPR) to estimate a reward function, training outputs $\mathbf{u} \subset \mathbb{R}$ and corresponding feature inputs $\mathbf{X_u} \subset \mathbf{F}$ are required. But, for IRL, training outputs do not exist since we only observe actions, not the reward outputs. Due to this reason, the true training outputs $\mathbf{u}$ are also estimated during the learning phase. In fact, estimating $\mathbf{u}$ for given $\mathbf{X_u}$ can be interpreted as modeling the reward function using a subset of kernel regressors, which are centered at feature points $\mathbf{X_u}$. The process of choosing a set of inputs $\mathbf{X_u}$ is explained in [76].

The most likely values of $\mathbf{u}$ and θ can be found by maximizing the following likelihood given demonstrations $\mathcal{D} = \{\zeta\}$:

$$\begin{aligned} P(\mathbf{u}, \theta | \mathbf{X_u}, \mathcal{D}) &\propto P(\mathcal{D}, \mathbf{u}, \theta | \mathbf{X_u}) \\ &= \left[\int_{\mathbf{r}} P(\mathcal{D}|\mathbf{r}) P(\mathbf{r}|\mathbf{u}, \mathbf{X_u}, \theta) d\mathbf{r} \right] P(\mathbf{u}|\mathbf{X_u}, \theta) P(\theta|\mathbf{X_u}), \end{aligned} \tag{6.12}$$

where $\mathbf{r}$ is a reward function or reward values of entire feature space $\mathbf{F}$, $P(\mathcal{D}|\mathbf{r})$ is the likelihood of demonstrations which can be computed using (6.1), $P(\mathbf{r}|\mathbf{u}, \mathbf{X_u}, \theta)$ is the GP posterior of the reward function, $P(\mathbf{u}|\mathbf{X_u}, \theta)$ is the prior probability of GP, and $P(\theta|\mathbf{X_u})$ is the predefined prior for hyperparameters. $P(\mathbf{u}|\mathbf{X_u}, \theta)$ is the Gaussian distribution with mean zero and a covariance matrix, whose entries are given by the following squared exponential (SE) kernel function[1]:

$$k_{se}(x_i, x_j; \theta) = \beta \exp\left(-\frac{1}{2}(x_i - x_j)^T \Lambda (x_i - x_j)\right), \tag{6.13}$$

where $\theta = \{\beta, \Lambda\}$, β is the gain of the SE kernel, and Λ is a diagonal matrix of length parameters. $P(\mathbf{r}|\mathbf{u}, \mathbf{X_u}, \theta)$ also has the Gaussian distribution with a predictive mean and covariance matrix given all feature points $\mathbf{F}$. However, the complexity of $P(\mathcal{D}|\mathbf{r})$ makes the integral intractable. In order to handle the integral in (6.12), the deterministic approximation method [101] has been used. Under this approximation scheme, the integral disappears and the reward function $\mathbf{r}(x_*)$ at an unseen input x_* becomes $\mathbf{k}_{*\mathbf{u}}\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{u}$, where $[\mathbf{k}_{*\mathbf{u}}]_i = k_{se}(x_*, \mathbf{X_u}_i)$, $\mathbf{X_u}_i$ is the ith element of $\mathbf{X_u}$, and $[\mathbf{K_{uu}}]_{ij} = k_{se}(\mathbf{X_u}_i, \mathbf{X_u}_j)$. The resulting likelihood can be written as:

$$P(\mathcal{D}, \mathbf{u}, \theta|\mathbf{X_u}) = P(\mathcal{D}|\mathbf{r} = \mathbf{K_{Fu}}\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{u})P(\mathbf{u}|\mathbf{X_u}, \theta)P(\theta|\mathbf{X_u}),$$

where $[\mathbf{K_{Fu}}]_{ij} = k_{se}(\mathbf{F}_i, \mathbf{X_u}_j)$ and $\mathbf{F}_i$ is the ith element of the feature space $\mathbf{F}$. By abuse of notation, $\mathbf{r}$ indicates the reward values of entire feature space or a reward function. Once the likelihood is optimized, the approximated reward can be used to recover the expert's policy on the entire state space.

Leveraged Gaussian Process

In [32], leveraged Gaussian processes (LGP) are proposed to use both positive and negative training samples for Gaussian process regression (GPR). A leveraged

[1] Note that other kernel functions can be used as well.

kernel function makes the prediction result of GPR close to positive samples and drift away from negative samples. Each training sample has its leverage value varying from -1 to $+1$, where -1 indicates a fully negative sample and $+1$ indicates a fully positive sample. A smooth leveraged kernel function proposed in [33] is defined as follows.

$$\begin{aligned} k(x_i, l_i, x_j, l_j; \theta) &= k_{lev}(l_i, l_j) k_{se}(x_i, x_j; \theta) \\ &= \beta \cos\left(\frac{\pi}{2}|l_i - l_j|\right) \exp\left(-\frac{1}{2}(x_i - x_j)^T \Lambda (x_i - x_j)\right), \end{aligned} \tag{6.14}$$

where x_i and x_j are inputs, l_i and l_j are leverage values of the ith and jth inputs, respectively.

A leveraged Gaussian process (LGP) can be used to express multiple correlated Gaussian processes with the same covariance structure by defining cross-covariance function of two Gaussian processes f and g as follows:

$$C_{f,g}(x_i, x_j) = k(x_i, l_i, x_j, l_j; \theta).$$

We note that $\cos\left(\frac{\pi}{2}|l_i - l_j|\right)$ controls the correlation between two Gaussian processes. For learning hyperparameters in LGP regression, derivatives of the leveraged kernel function with respect to hyperparameters are required and they can be computed as follows:

$$\begin{aligned} \frac{\partial k(x_i, x_j, l_i, l_j)}{\partial \beta} &= \frac{k(x_i, x_j, l_i, l_j)}{\beta} \\ \frac{\partial k(x_i, x_j, l_i, l_j)}{\partial \lambda_k} &= -\frac{1}{2}(x_{i,k} - x_{j,k})^2 k(x_i, x_j, l_i, l_j), \end{aligned} \tag{6.15}$$

where λ_k indicates the kth diagonal element of Λ and $x_{i,k}$ is the kth element of x_i.

6.1.6 Inverse Reinforcement Learning with Leveraged Gaussian Processes

Benefits of Negative Demonstrations

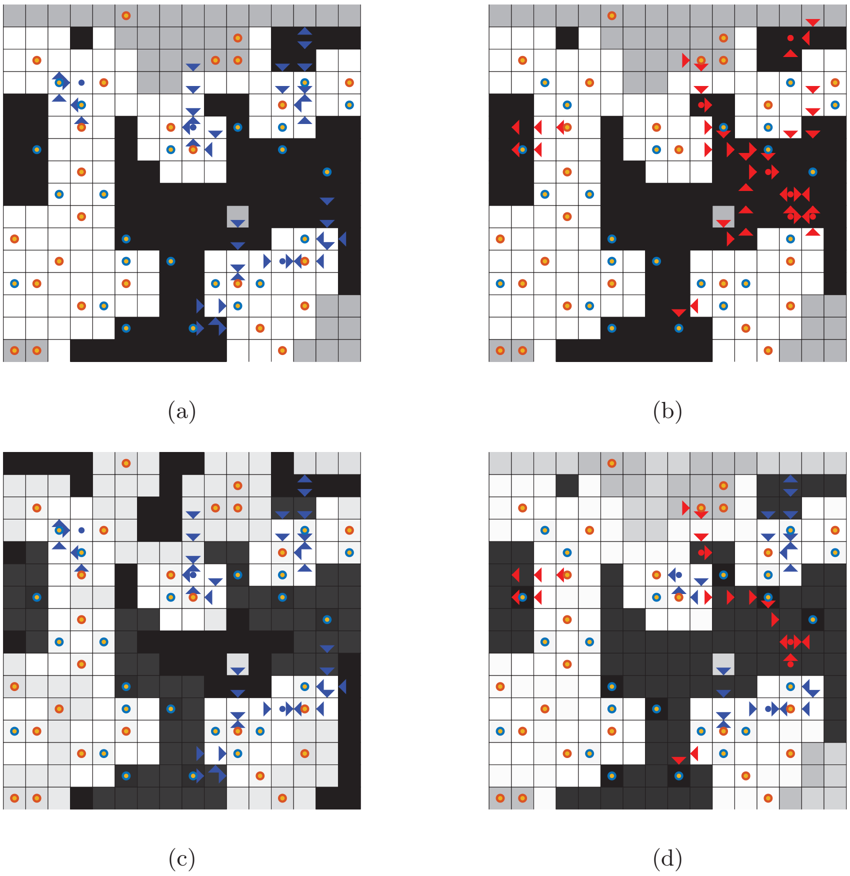
Many existing IRL algorithms focus on using demonstrations of *what to do*. However, as mentioned in [109], the fact that experts rarely encounter fatal situations leads to the lack of information about how to overcome in a fatal situation. To handle this problem, we provide the information about *what not to do* using a negative demonstrator. Demonstrations from experts (positive demonstration) are mostly distributed near the high reward regions. However, we model a negative demonstrator having an inverted reward function compared to an expert. Hence, negative demonstrations from a negative demonstrator is more likely to be generated near low reward regions.

For example, consider the objectworld experiment [76]. Figure 6.3 shows examples of positive and negative demonstrations and results of IRL algorithms. In an $N \times N$ objectworld, colored objects are randomly populated and the state is the location (or cell) in the $N \times N$ grid map. Possible actions are moving towards four adjacent grid cells or staying at the current cell. In Figure 6.3, there are two outer colors (red and blue). The rewards function is defined such that the cell near both red and blue colored objects has a reward of $+1$, the cell near only blue colored objects has a reward of -1, and other cells have a reward of 0. (More details about the objectworld are discussed in Section 6.1.7.) In Figure 6.3(a), most of positive demonstrations move towards high reward regions and positive demonstrations are rarely distributed near low reward regions. On the other hand, in Figure 6.3(b), negative demonstrations are more likely to distributed near low reward regions and negative demonstrations can provide more information about *what not to do.* In Figure 6.3(c) and 6.3(d), reward function reconstruction results of GPIRL [76] and the proposed method are shown, respectively. The result from the proposed method, which uses both positive and negative demonstrations, is more accurate than GPIRL, which uses only positive demonstrations given the same number of demonstrations. We can draw the conclusion that negative demonstrations can provide information about low reward regions and we can estimate the reward function more precisely using both positive and negative demonstrations.

Demonstrator Modeling

Before presenting the problem formulation used In this section, we describe the model of a demonstrator with multiple levels of proficiencies. The main contribution of the proposed model is that it allows the use of negative and positive demonstrations in a single framework. The proficiency of a demonstrator is represented as the leverage parameter in an LGP and we will refer to the leverage parameter as the *proficiency.* The proficiency of an expert is $+1$. On the other hand, a fully negative demonstrator has the proficiency of -1, i.e., she optimizes a reward function which is inverted from the expert's reward function. A demonstrator with the positive or negative proficiency will be referred to as a positive or negative demonstrator, respectively.

A graphical representation of the proposed demonstrator model is shown in Figure 6.4. Each demonstrator has a different version of the reward function but they are related by its proficiency and the original reward function of the expert. Since we model the expert's reward function using an LGP, the prior distribution of expert's reward function is a zero-mean Gaussian process with the covariance function based on a leveraged kernel function.



(a)

(b)

(c)

(d)

Figure 6.3: A 16 × 16 objectworld. Colored circles are objects and the brighter the grid color is, the higher the reward is. Blue arrows are positive demonstrations and red arrows are negative demonstrations. (a) Examples of positive demonstrations. (b) Examples of negative examples. (c) The reward function reconstructed by GPIPL [76] using only positive demonstrations. (d) The reward function reconstructed by the proposed method using both positive and negative demonstrations.

Figure 6.5 illustrates how the proficiency l affects the demonstrations of a demonstrator with a unicycle dynamic model. Trajectories from seven different

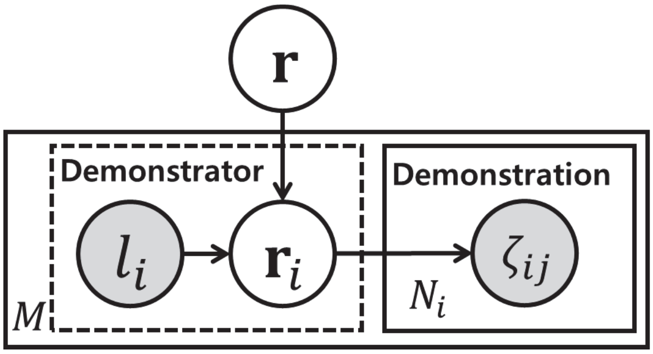


Figure 6.4: A graphical representation of the proposed demonstrator model with multiple proficiencies, where $\mathbf{r}$ is the true reward function (expert's reward) with proficiency +1, M is the number of demonstrators, N_i is the number of demonstrations from the ith demonstrator, l_i is the proficiency of the ith demonstrator, $\mathbf{r}_i$ is the reward function of the ith demonstrator, and ζ_{ij} is the jth demonstration from the ith demonstrator. There are a total of N demonstrations, i.e., $\sum_{i=1}^{M} N_i = N$.

proficiencies, varying from -1 to 1, are depicted with different colors. Higher reward regions are shown in a brighter color. The dark blue trajectories (proficiency of +1) move in brighter regions while light blue trajectories (proficiency between 0.2 and 0.5) move between bright and dark regions. For red trajectories (proficiency of -1) moves in dark regions, as expected.

Problem Formulation

We consider the problem of finding the reward function from given demonstrations and proficiencies and it can be formulated as follows:

$$\underset{\mathbf{u},\theta}{\text{maximize}} \quad \log P(\mathbf{u}, \theta | \mathbf{X}_{\mathbf{u}}, \bar{\mathcal{D}}), \tag{6.16}$$

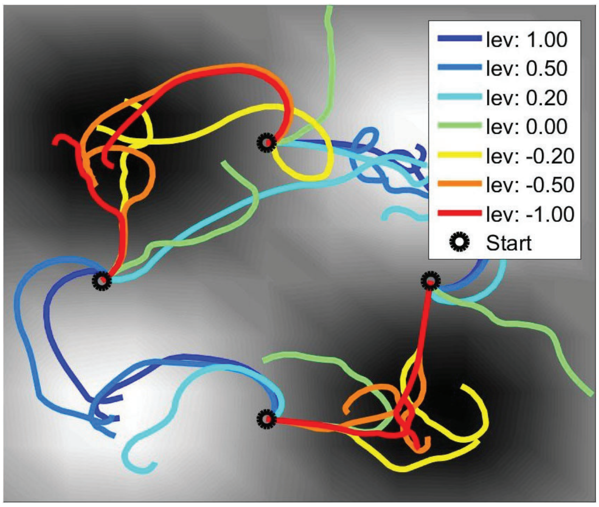


Figure 6.5: Sampled trajectories from demonstrators with different proficiencies (unicycle dynamic model). In the legend, 'lev' indicates the proficiency of a demonstration.

where the reward function is parameterized by $\mathbf{X_u}$ and $\mathbf{u}$ indicating a subset of features and corresponding reward values, respectively, and $\bar{\mathcal{D}} = \{l_i, \{\zeta_{ij}\}_j^{N_i}\}_i^M$

Here, we maximize the probability of reward outputs and hyperparameters given inputs, demonstrations, and proficiencies. We can decompose the objective function into four parts as follows.

$$P(\mathbf{u},\theta|\mathbf{X_u},\bar{\mathcal{D}}) \propto P(\bar{\mathcal{D}},\mathbf{u},\theta|\mathbf{X_u}) = P(\bar{\mathcal{D}}|\mathbf{u},\mathbf{X_u},\theta)P(\mathbf{u}|\mathbf{X_u},\theta)P(\theta|\mathbf{X_u})$$
$$= \prod_{i=1}^{M}\prod_{j=1}^{N_i} P(\zeta_{ij},l_i|\mathbf{u},\mathbf{X_u},\theta)P(\mathbf{u}|\mathbf{X_u},\theta)P(\theta|\mathbf{X_u})$$
$$\propto \prod_{i=1}^{M}\prod_{j=1}^{N_i}\int_{\mathbf{r}_i} P(\zeta_{ij}|\mathbf{r}_i)P(\mathbf{r}_i|l_i,\mathbf{u},\mathbf{X_u},\theta)P(\mathbf{u}|\mathbf{X_u},\theta)P(\theta|\mathbf{X_u}),$$

where l_i and $\mathbf{r}_i$ are the proficiency and reward of the ith demonstrator, respectively, and $P(\mathbf{r}_i|l_i,\mathbf{u},\mathbf{X_u},\theta)$ is the LGP posterior. Since the integral cannot be analytically computed, we utilize the sparse Gaussian process approximation, similar to [76, 101], where a small subset of inputs and its corresponding outputs are used to represent the full set. In particular, we assume that the LGP posterior is deterministic. Then, the integration can be avoided and the resulting probability can be computed as below.

$$P(\bar{\mathcal{D}}, \mathbf{u}, \theta | \mathbf{X_u}) \propto \prod_{i=1}^{M} \prod_{j=1}^{N_i} P(\zeta_{ij} | \mathbf{r}_i = \mathbf{K_{Fu}} \mathbf{K_{uu}^{-1}} \mathbf{u}) P(\mathbf{u} | \mathbf{X_u}, \theta) P(\theta | \mathbf{X_u}),$$

where the kernel matrices $\mathbf{K_{uu}}$ and $\mathbf{K_{Fu}}$ are computed by leveraged kernel function (6.14) using the proficiency l_i and the expert's proficiency $+1$. The equation consists of three parts: the likelihood of demonstrations, the LGP marginal likelihood of outputs $\mathbf{u}$, and the prior on hyperparameters θ.

Finally, (6.16) becomes

$$\max_{u,\theta} \sum_{i=1}^{M} \sum_{j=1}^{N_i} \underbrace{\log P(\zeta_{ij} | \mathbf{r}_i)}_{\text{IRL likelihood}} + \underbrace{\log P(\mathbf{u} | \mathbf{X_u}, \theta)}_{\text{LGP marginal likelihood}} + \underbrace{\log P(\theta | \mathbf{X_u})}_{\text{prior}}, \tag{6.17}$$

where $\log P(\zeta_{ij} | \mathbf{r}_i)$ is given in (6.1). The other two terms can be computed as

$$\begin{aligned} \log P(\mathbf{u} | \mathbf{X_u}, \theta) &= -\frac{1}{2} \mathbf{u}^T \mathbf{K_{uu}}^{-1} \mathbf{u} - \frac{1}{2} \log |\mathbf{K_{uu}}| - \frac{n}{2} \log 2\pi \\ \log P(\theta | \mathbf{X_u}) &= -\frac{1}{2} \mathbf{tr}(\mathbf{K_{uu}}^{-2}) - \sum_{k} \log(\lambda_k + 1), \end{aligned}$$

where λ_k is the kth diagonal entry of Λ. The LGP marginal likelihood and the prior on hyperparameters have an effect of regularization [76]. The IRL likelihood in (6.17) allows us to incorporate multiple proficiency information for learning an expert's reward function via its derivative. The derivative of the likelihood of the ith demonstrator is computed as follows:

$$\sum_{j=1}^{N_i} \frac{\partial \log P(\zeta_{ij} | \mathbf{r}_i)}{\partial \mathbf{r}_i} \frac{\partial \mathbf{r}_i}{\partial \mathbf{u}} = (\hat{\mu}_i - \mu_i)^T \mathbf{K_{Fu}} \mathbf{K_{uu}^{-1}},$$

where $\hat{\mu}_i$ is the empirical state visitation count from N_i demonstrations, μ_i is the expected state visitation computed by an iterative algorithm proposed in

[151] under $\mathbf{r}_i$. If $l_i < 0$ (negative demonstrator), the kernel matrix $\mathbf{K_{Fu}}$ makes the gradient decrease reward values of the states which are highly visited by the ith demonstrator. On the other hand, if $l_i > 0$ (positive demonstrator), the $\mathbf{K_{Fu}}$ makes the gradient increase reward values of the states which are highly visited by the ith demonstrator. Hence, the proposed method finds a reward function such that positive demonstrations result in higher values and negative demonstrations result in lower values. We optimize (6.17) using a gradient ascent method. In order to find the gradient of the objective function, we need to compute the derivatives of (6.17) we have to compute the derivatives of following three parts.

$$\sum_{i=0}^{N} \underbrace{\log P(\zeta_i|\mathbf{r}_i)}_{\text{IRL likelihood}} + \underbrace{\log P(\mathbf{r_e}|\mathbf{X_e}, \theta)}_{\text{LGP marginal likelihood}} + \underbrace{\log P(\theta|\mathbf{X_e})}_{\text{hyperparameter prior}}$$

where the IRL likelihood makes $\mathbf{r}_i$ generate similar behavior compared to demonstrations with proficiency l_i, the LGP marginal likelihood mainly acts as regularization of $\mathbf{u}$, and the hyperparameter prior prevents a singular covariance matrix and induces sparsity in scale parameters.

Since the problem is not convex, a gradient ascent method can suffer from local optima. To handle this problem, the process can be repeated with multiple random restarts and the best solution can be chosen. The derivatives of the other parts of the objective function can be computed by applying the chain rule and using the kernel derivatives in (6.15).

Derivatives

The objective function is partial differentiated by the hyperparameters θ and expert's output $\mathbf{r_e}$ and derivatives of each terms are computed by chain rule or direct differentiation.

IRL Likelihood Derivative

$$
\begin{aligned}
\frac{\partial \log P(\zeta_i|\mathbf{r}_i)}{\partial \mathbf{r}_i}\frac{\partial \mathbf{r}_i}{\partial \mathbf{r_e}} &= (\hat{\mu}_i - \mu_i)\mathbf{K_{re}}\mathbf{K_{ee}}^{-1} \\
\frac{\partial \log P(\zeta_i|\mathbf{r}_i)}{\partial \mathbf{r}_i}\frac{\partial \mathbf{r}_i}{\partial \theta_k} &= (\hat{\mu}_i - \mu_i)\left(\frac{\partial \mathbf{K_{re}}}{\partial \theta_k} - \mathbf{K_{re}}\mathbf{K_{ee}}^{-1}\frac{\partial \mathbf{K_{ee}}}{\partial \theta_k}\right)\mathbf{K_{ee}}^{-1}\mathbf{r_e}
\end{aligned}
$$

where $\hat{\mu}_i$ is empirical visitation of l_i-proficient demonstrator, μ_i is the expected visitation under the reward $\mathbf{r}_i$, $\mathbf{K_{re}}\mathbf{K_{ee}}^{-1}$ is $\frac{\partial \mathbf{r}_i}{\partial \mathbf{r_e}}$ and $\left(\frac{\partial \mathbf{K_{re}}}{\partial \theta_k} - \mathbf{K_{re}}\mathbf{K_{ee}}^{-1}\frac{\partial \mathbf{K_{ee}}}{\partial \theta_k}\right)\mathbf{K_{ee}}^{-1}\mathbf{r_e}$ is $\frac{\partial \mathbf{r}_i}{\partial \theta_k}$. $\hat{\mu}_i$ is estimated by given demonstrations and μ_i is computed by iterative algorithm proposed in [151]. $\frac{\mathbf{K_{ee}}}{\partial \theta_k}$ and $\frac{\mathbf{K_{re}}}{\partial \theta_k}$ are computed by (6.15).

LGP Marginal Likelihood Derivative

The LGP marginal likelihood has two terms which are *data fitting* term and *normalization* term. The derivative with respect to $\mathbf{r_e}$ is simply computed as the marginal likelihood is quadratic form. Hence, the derivatives of marginal likelihood are computed as follows.

$$
\begin{aligned}
\frac{\partial \log P(\mathbf{r_e}|\mathbf{X_e}, \theta)}{\partial \mathbf{r}_e} &= -\mathbf{K_{ee}}^{-1}\mathbf{r_e} \\
\frac{\partial \log P(\mathbf{r_e}|\mathbf{X_e}, \theta)}{\partial \theta_k} &= \frac{1}{2}\mathbf{r_e}^T\mathbf{K_{ee}}^{-1}\frac{\partial \mathbf{K_{ee}}}{\partial \theta_k}\mathbf{K_{ee}}^{-1}\mathbf{r_e} - \frac{1}{2}\mathbf{tr}(\mathbf{K_{ee}}^{-1}\frac{\partial \mathbf{K_{ee}}}{\partial \theta_k})
\end{aligned}
$$

where the derivative with respect to hyperparameters is well explained in [104].

Hyperparameter Prior Derivative

The hyperparameter prior can be differentiated as follows.

$$
\begin{aligned}
\frac{\partial \log P(\theta|\mathbf{X_e})}{\partial \beta} &= \mathbf{tr}(\mathbf{K_{ee}}^{-3}\frac{\partial \mathbf{K_{ee}}}{\partial \beta}) \\
\frac{\partial \log P(\theta|\mathbf{X_e})}{\partial \lambda_k} &= \mathbf{tr}(\mathbf{K_{ee}}^{-3}\frac{\partial \mathbf{K_{ee}}}{\partial \lambda_k}) - \frac{1}{\sum_{k=1}^{d_u}\Lambda_{kk} + 1}
\end{aligned}
$$

where the derivative with respect to scale parameter λ_k has two terms from inverse covariance and sparsity inducing regularization.

Proficiency Estimation

By using our demonstrator model, we can utilize demonstrations with multiple proficiencies. However, in practice, it may be difficult to collect demonstrations with proficiencies. To handle this issue, the semi-supervised framework similar to [118, 11] can be used. Under the semi-supervised framework, we have both labeled demonstrations $\bar{\mathcal{D}} = \{l_i, \{\zeta_{ij}\}_j^{N_i}\}_i^M$ and unlabeled demonstrations $\mathcal{D} = \{\zeta\}$, where labeled (or unlabeled) demonstration means that the proficiency of demonstration is known (or unknown). We estimate the proficiencies of $\mathcal{D}$ using the kernel ridge regression [111] based on the SE kernel function defined over a pair of trajectories in the feature space as follows.

$$k_p(\zeta_i, \zeta_j) = \exp\left(-\frac{1}{2\sigma_p} d(\zeta_i, \zeta_j)^2\right),$$

where $\sigma_p > 0$ is a scale parameter, $d(\zeta_i, \zeta_j) = ||\hat{\mathbf{f}}_i - \hat{\mathbf{f}}_j||_2$, $\hat{\mathbf{f}}_i = \sum_{t=0}^{T} \gamma^t x_{i,t}$, T is the length of ζ_i and $x_{i,t}$ is a feature vector corresponding to $s_{i,t}$ in ζ_i. Then, the proficiency of $\mathcal{D}$ can be estimated as follows:

$$\mathbf{L}_{\mathcal{D}} = \mathbf{K}^{\mathbf{p}}_{\mathcal{D}\bar{\mathcal{D}}} {\mathbf{K}^{\mathbf{p}}_{\bar{\mathcal{D}}\bar{\mathcal{D}}}}^{-1} \mathbf{L}_{\bar{\mathcal{D}}},$$

where $\mathbf{L}_{\mathcal{D}}$ is a vector stacking estimated proficiencies of $\mathcal{D}$, $\mathbf{L}_{\bar{\mathcal{D}}}$ is a vector stacking the proficiencies of $\bar{\mathcal{D}}$, $[\mathbf{K}^{\mathbf{p}}_{\bar{\mathcal{D}}\bar{\mathcal{D}}}]_{ij} = k_p(\zeta_i, \zeta_j)$, ζ_i is the ith demonstration in $\bar{\mathcal{D}}$ and $\mathbf{K}^{\mathbf{p}}_{\mathcal{D}\bar{\mathcal{D}}}$ can be computed by the same manner with $\mathcal{D}$ and $\bar{\mathcal{D}}$.

6.1.7 Simulations and Experiment

In this section, we evaluate the performance of the proposed inverse reinforcement learning algorithm by comparing against existing methods. The proposed method (LIRL) is compared with IRL algorithms using only positive demonstrations: AL [1], MMP [105], BIRL [103], MEIRL [152], LEARCH [107], OptV [38], FIRL [75] GPIRL [76], and NPBFIRL [30]. We also compare with SSAL [137], a relatively recent algorithm, which uses both positive and negative demonstrations. The original SSAL is a semi-supervised learning method which learns the reward function from both labeled and unlabeled demonstrations simultaneously by clustering unlabeled demonstrations. However, in our simulation, we treat SSAL as supervised apprenticeship learning (SAL) which maximizes the margin between positive and negative demonstrations by providing fully labeled demonstrations. We also implement a supervised version of MMP (SMMP) with a new constraint which enforces the resulting value function to be bigger than that of negative demonstrations in a max-margin framework.

To demonstrate the benefit of using negative demonstrations, we have prepared two types of demonstrations: positive and negative. A positive demonstration is sampled from the optimal policy with the proficiency of $+1$. A negative demonstration is sampled from the policy, which optimizes the inverted reward function of the original reward function and its proficiency is -1. While the proposed method can handle multiple proficiency levels, existing methods can only handle binary levels. Hence, for a fair comparison, we have only used demonstrations with either $+1$ or -1 proficiency. The performance of each algorithm is evaluated using the expected value difference (EVD), which is the difference between the optimal value and the value obtained by following the policy learned by an IRL algorithm.

Objectworld

We first validated the performance of IRL methods using the *objectworld* experiment [76], where the state and action space consist of an $N \times N$ grid map and five actions (up, down, left, right, or staying), respectively. Given an action, an agent successfully performs the action with probability of 0.7 or, otherwise, makes a random movement. Inside the grid map, objects with random colors are randomly deployed where each object has an inner and outer colors. Both inner and outer colors are selected from $C \geq 2$ distinct colors. Among C colors, two specific colors c_1 and c_2 are used to compute the reward at each state and other colors work as distracting factors. The reward function is defined as follows:

$$r(s) = \begin{cases} 1, & \text{if } d_1(s) < 3 \wedge d_2(s) < 2 \\ -2, & \text{if } d_1(s) < 3 \wedge d_2(s) \geq 2 \\ 0, & \text{otherwise,} \end{cases}$$

where $d_1(s)$ and $d_2(s)$ are the Euclidean distances from state s to the nearest object whose outer color is c_1 and c_2, respectively.

The state is represented using a binary feature $\phi(s)$ [76], where

$$\phi_i^k(s)_j = \begin{cases} 1, & \text{if } d_i^k(s) \leq j \\ 0, & \text{if } d_i^k(s) > j, \end{cases}$$

for $i = 1, \ldots, C$, $j = 1, \ldots, N$, and $k = 1, 2$ where $d_i^k(s)$ indicates the Euclidean distance from state s to the nearest object whose inner ($k = 1$) or outer ($k = 2$) color is c_i. Hence, by combining inner and outer colors with C colors and N distance thresholds, the dimension of a feature becomes $2CN$. The reason why binary feature is utilized is that some algorithms [75, 30] only work with binary features. In our simulations, we set $N = 32$ and $C = 2$.

We have prepared several sets of demonstrations under three different ratios of the number of negative demonstrations to the number of all demonstrations: 10%, 30% and 50%. Algorithms which can handle both positive and negative demonstrations are provided with three different sets of mixed demonstrations.

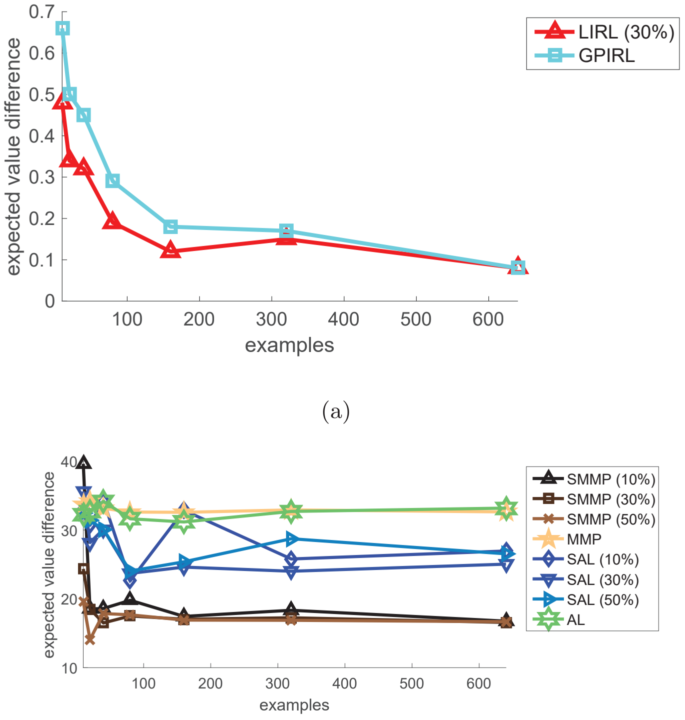
Algorithms using only positive demonstrations are provided with the same number of positive demonstrations.

The average EVDs of different algorithms from eight independent runs are shown in Table 6.2 and Figure 6.6. Since the proposed method (LIRL) and GPIRL constantly outperforms other methods, results from two algorithms are shown in Figure 6.6(a). LIRL shows better performance than GPIRL given the same number of demonstrations. Moreover, the average EVD of LIRL with 160 mixed demonstrations is better than that of GPIRL with 320 positive demonstrations. Figure 6.6(b) shows the benefits of using negative examples when the technique is applied to other methods. SMMP and SAL, which are extensions of MMP and AL with both positive and negative demonstrations, perform better than MMP and AL. This result empirically shows that the use of negative demonstrations can enhance performance of inverse reinforcement learning. The overall results are shown in Table 6.2, where the best performance is marked in bold.

Highway Driving

The same set of IRL algorithms are tested on the *highway driving* task [76], where the state of an agent is the location in a three-lane highway road and the speed at four different levels $(1, 2, 3, 4)$ and the agent can move one lane to the left or right and change vehicle's speed. The action is successfully performed with probability of 0.7 and is failed with probability of 0.3. In the road, several vehicles are randomly deployed with the lowest speed where four different types of vehicle exist by combining two different vehicle types (car and motorcycle) and two different driver types (civilian and police).

The reward function is designed to learn the driving behavior as fast as possible unless the agent's car is located near the police car or motorcycle. If the



(a)

(b)

Figure 6.6: Average expected value differences of different IRL algorithms from the 32 × 32 objectworld experiment with two colors. (a) Results of LIRL with 30% mixed demonstrations and GPIRL. (b) Results of SMMP, SAL, MMP and AL.

distance between the agent and the police vehicle is within two lanes, depending on its speed $(1, 2, 3, 4)$, the agent gets a reward of $(0, -2, -10, -10)$, respectively. If the distance between the agent and the police vehicle is further than two lanes, depending on its speed $(1, 2, 3, 4)$, the agent gets a reward of $(-2, 0, 2, 6)$, respectively.

The state of the agent is represented using a 263 dimensional binary feature. Three dimensions for three lanes and four dimensions for four speed levels are

Algorithms	Sample Size						
	10	20	40	80	160	320	640
LIRL (10%)	**0.42**	**0.3**	0.20	0.24	**0.12**	0.16	0.11
LIRL (30%)	0.48	0.34	0.32	**0.19**	**0.12**	0.15	**0.08**
LIRL (50%)	0.79	0.36	**0.17**	0.43	0.23	**0.08**	0.13
SMMP (10%)	39.62	18.53	18.60	19.79	17.45	18.34	16.76
SMMP (30%)	24.35	18.51	16.50	17.51	17.04	17.25	16.57
SMMP (50%)	19.59	14.09	17.86	17.68	16.91	16.88	16.68
SAL (10%)	33.00	30.69	34.28	22.62	32.86	25.79	26.95
SAL (30%)	35.61	28.17	30.10	23.70	24.61	24.01	25.04
SAL (50%)	32.49	31.40	30.08	24.04	25.38	28.73	26.54
BIRL	14.72	15.37	15.25	13.26	12.52	12.48	12.91
GPIRL	0.66	0.50	0.45	0.29	0.18	0.17	**0.08**
NPBFIRL	10.76	10.43	9.99	9.74	10.19	10.17	10.27
MEIRL	18.33	15.76	15.68	14.29	13.79	13.50	13.70
OptV	40.77	36.73	29.78	22.61	14.78	6.77	2.08
MMP	33.72	34.20	33.15	32.61	32.57	32.92	32.63
AL	32.20	32.69	34.16	31.63	31.17	32.70	33.18
LEARCH	36.59	36.20	35.17	34.27	32.30	32.06	31.66
FIRL	42.07	42.36	41.82	42.30	42.15	42.20	42.31

Table 6.2: Results from the 32×32 objectworld experiment. The average EVDs from eight independent runs are shown. The best performance is marked in bold. The percentage value inside parentheses is the mixing ratio of the number of negative demonstrations to the number of total demonstrations.

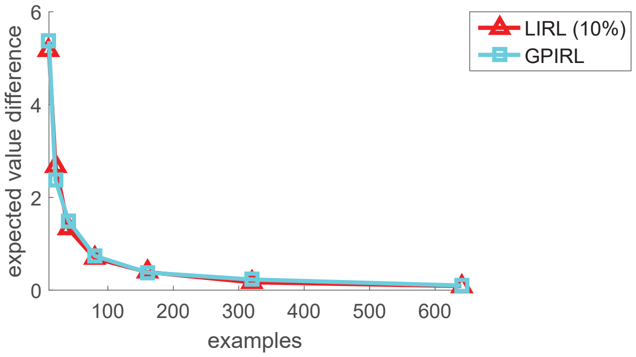


Figure 6.7: The results of 64-car-length highway driving with varying the number of 32 length demonstrations. The EVD of LIRL with 10% mixed demonstrations and GPIRL are shown. LIRL has better performance than GPIRL

added by a combination of four types of vehicles, eight discretization of distance to the nearest vehicle, and eight outward directions $(3 + 4 + 4 \times 8 \times 8 = 263)$.

The average EVDs from eight independent runs are shown in Table 6.3, where the best performance is marked in bold. The proposed method (LIRL) and GPIRL constantly outperform other IRL algorithms under a various number of samples. LIRL shows better performance than GPIRL as the number of demonstrations increases while the performance gap between LIRL and GPIRL is narrower than the objectworld experiment. We have also observed the similar performance improvement in SMMP over MMP.

6.1.8 Summary

In this section, a new inverse reinforcement learning algorithm is proposed. The proposed algorithm uses a leveraged Gaussian process to model a nonlinear reward function and can learn from both positive and negative demonstrations. We have also introduced a novel demonstrator model for modeling demonstrations

Algorithms	Sample Size						
	10	20	40	80	160	320	640
LIRL (10%)	**5.18**	2.67	**1.32**	**0.69**	0.39	**0.17**	**0.09**
LIRL (30%)	7.16	2.48	1.48	0.82	0.44	0.20	**0.09**
LIRL (50%)	13.04	3.35	1.55	0.93	0.45	0.24	0.13
SMMP (10%)	72.40	58.29	51.32	50.18	33.56	33.87	36.11
SMMP (30%)	55.44	34.88	39.52	39.41	34.48	34.34	34.66
SMMP (50%)	58.67	37.14	47.40	40.47	37.65	35.63	34.45
SAL (10%)	56.26	58.15	59.36	59.23	62.81	61.88	63.36
SAL (30%)	55.88	55.77	59.15	59.95	62.01	61.85	62.09
SAL (50%)	54.80	55.71	58.82	58.34	60.53	62.54	62.22
BIRL	15.49	11.92	8.18	5.93	4.33	3.15	2.14
GPIRL	5.37	**2.37**	1.49	0.74	**0.38**	0.23	0.10
NPBFIRL	38.92	30.64	30.71	25.59	22.94	23.40	19.69
MEIRL	5.22	2.58	1.54	2.06	1.53	0.72	0.59
OptV	19.57	14.61	6.17	4.71	1.72	0.68	0.14
MMP	42.34	42.47	45.71	45.60	48.01	49.59	49.10
AL	45.14	47.77	50.43	51.57	55.72	56.10	56.17
LEARCH	46.87	45.67	45.62	44.88	43.88	43.38	44.09
FIRL	20.36	18.55	10.25	3.54	3.06	2.87	0.71

Table 6.3: Results from the highway driving experiment. The best performance is marked in bold.

with different proficiencies. To the best of our knowledge, the proposed method is the first algorithm which can learn a nonlinear reward function using both positive and negative demonstrations. In simulation, the proposed method outperforms existing IRL algorithms. Our experimental results also demonstrate the benefit of using negative demonstrations in inverse reinforcement learning.

Chapter 7

Conclusion

In this dissertation, we have investigated several robot learning methods. Especially, we have focused on developing learning frameworks to reduce the sample complexity of robot learning. In reinforcement learning, we have developed efficient exploration methods based on entropy and perturbation. In imitation learning, we have developed a novel imitation learning framework by incorporating both negative and positive demonstrations.

First, in entropy-based exploration, we have proposed a novel MDP with spare Tsallis entropy regularization which induces a sparse and multi-modal optimal policy distribution. We have also analyzed the full mathematical analysis of the proposed sparse MDPs. Furthermore, we have extended sparse Tsallis entropy to generalized Tsallis entropy. Hence, we have proposed a unified framework which widen a class of different Tsalli entropies in RL problems and we call this framework Tsallis MDPs. We have provided the full theoretical analysis about Tsallis MDPs including guarantees of convergence, optimality, and performance error bounds. We would like to note that Tsallis MDPs include sparse MDPs and soft MDPs [51] as special cases. For Tsallis MDPs, we have extended it to the Tsallis actor-critic (TAC) method to handle a continuous state-action space. It has been observed that there exists a suitable entropic index for each different RL problem and TAC with the optimal entropic index outperforms existing actor-critic methods. However, since finding an entropic index with the brute force search is a demanding task, we have also present TAC^2 that gradually increases the entropic index. We have applied TAC^2 on real-world problems of learning a feedback controller for soft mobile robots and demonstrated that TAC^2 shows more efficient exploration tendency than adjusting the regularization coefficient. Furthermore, we also have applied a Shannon entropy exploration to online learning for grasping unknown objects. we have proposed a novel Shannon entropy regularized neural contextual bandit online learning (SERN). We proved that SERN has no regret properties and its error converges to zero. In both simulation and the real-world experiments, we empirically showed that SERN outperforms a ϵ-greedy method and improves the grasp performance efficiently.

Second, in perturbation-based exploration, we have analyzed the random perturbation method for a stochastic bandit setting under both sub-Gaussian and heavy-tailed rewards. We have provided the general analysis scheme for the both upper and lower bound of the regret of heavy-tailed perturbations under both sub-Gaussian and heavy-tailed rewards. Especially, our analysis scheme have made it available to analyze the heavy-tailed perturbations, such as Pareto, Fréchet, and GEV distribution which was not covered by the previous work [64]. The results of the Pareto and Fréchet perturbations have provided an interesting observation in that they can achieve the same near-optimal regret bound as the sub-Weibull perturbation under sub-Gaussian reward assumption. For heavy-tailed rewards, we have proposed novel p-robust estimator which can handle heavy-tailed noise distributions which does not require prior knowledge about the bound on the

p-th moment of rewards. By using the proposed estimator, we also proposed an adaptively perturbed exploration with a p-robust estimator (APE^2) and proved that APE^2 has better regret bound than robust UCB. We have theoretically and empirically demonstrated that APE^2 can overcome rewards that are corrupted by heavy-tailed noises, making APE^2 an appropriate solution for many practical problems, such as online classification, online learning of a recommendation system, and reinforcement learning.

In imitation learning, we have improved sample efficiency of imitation learning by employing negative demonstrations. We have proposed a new inverse reinforcement learning algorithm which uses a leveraged Gaussian process to model a nonlinear reward function and can learn from both positive and negative demonstrations. We have also introduced a novel demonstrator model for modeling demonstrations with different proficiencies. Furthermore, we empirically showed that the proposed method outperforms existing IRL algorithms in simulations. Our experimental results also demonstrate the benefit of using negative demonstrations in inverse reinforcement learning. Furthermore, we also have applied entropy-based exploration for model free imitation learning. Hence, we have proposed a novel maximum causal Tsallis entropy (MCTE) framework and proved that an optimal solution of MCTE framework is a sparsemax distribution. We have also provided the full mathematical analysis of the proposed framework, including the concavity of the problem, the optimality condition, and the interpretation as robust Bayes. We have also developed the maximum causal Tsallis entropy imitation learning (MCTEIL) algorithm. In experiments, we have verified that the proposed method has advantages over existing methods for learning the multi-modal behavior of an expert since a sparse MDN can search in diverse directions efficiently. From the analysis and experiments, we have shown that the proposed MCTEIL method is an efficient and principled way to learn the multi-modal behavior of an expert.

Appendices

Appendix A

Proofs of Chapter 3.1.

In this section, we provide entire proofs of Chapter 3.1.

A.1 Useful Properties

We first introduce notations and properties. Before introducing a notations, we would like to mention a state-action rewards function $r(s,a)$. $r(s,a,s')$ is generally used as a reward function. Then, state-action reward is defined as

$$r(s,a) := \mathbb{E}[r(s,a,S_{t+1})|S_t = s, A_t = a] = \sum_{s'} r(s,a,s')\mathbb{P}(s'|s,a)$$

where $\mathbb{P}(s'|s,a)$ is a transition probability. In Table A.1, all notations and definitions are summarized. For notational simplicity, we denote the expectation of a discounted sum, $\mathbb{E}[\sum_{t=0}^{\infty} \gamma^t f(s_t,a_t)|\pi,d,T]$, by $\mathbb{E}_\pi[f(s,a)]$, where $f(s,a)$ is a function of a state and an action, such as a rewards function, $r(s,a)$, or an indicator function, $\mathbb{I}_{\{s'=s,a'=a\}}$. We also denote the expectation conditioned on an initial state, $\mathbb{E}[\sum_{t=0}^{\infty} \gamma^t f(s_t,a_t)|\pi, s_0 = s, T]$, by $\mathbb{E}_\pi[f(s,a)|s_0 = s]$. The utility, value, state visitation can be compactly expressed as below in terms of vectors and matrices:

$$J_\pi^{sp} = d^\intercal G_\pi^{-1} r_\pi^{sp}, \qquad V_\pi^{sp} = G_\pi^{-1} r_\pi^{sp}$$
$$J_\pi^{soft} = d^\intercal G_\pi^{-1} r_\pi^{soft}, \qquad V_\pi^{soft} = G_\pi^{-1} r_\pi^{soft}, \;\; \rho_\pi = d^\intercal G_\pi^{-1}$$

where $x^\intercal$ is the transpose of vector x, $G_\pi = (I - \gamma T_\pi)$, sp indicates a sparse MDP problem which is defined as follows:

$$\begin{aligned} \underset{\pi}{\text{maximize}} \quad & \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \middle| \pi, d, T\right] + \alpha W(\pi) \\ \text{subject to} \quad & \forall\, s \;\; \sum_{a'} \pi(a'|s) = 1, \\ & \forall\, s, a \;\; \pi(a'|s) \geq 0, \end{aligned} \tag{A.1}$$

and $soft$ indicates a soft MDP problem which is defined as follows:

$$\begin{aligned} \underset{\pi}{\text{maximize}} \quad & \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \middle| \pi, d, T\right] + \alpha H(\pi) \\ \text{subject to} \quad & \forall\, s \;\; \sum_{a'} \pi(a'|s) = 1, \\ & \forall\, s, a \;\; \pi(a'|s) \geq 0. \end{aligned} \tag{A.2}$$

A.2 Sparse Bellman Optimality Equation

The following theorem explains the optimality condition of the sparse MDP from Karush-Kuhn-Tucker (KKT) conditions.

Proof of Theorem 1. The KKT conditions of (A.1) are as follows:

$$\forall s, a \quad \sum_{a'} \pi(a'|s) - 1 = 0, \;\; -\pi(a|s) \leq 0 \tag{A.3}$$

$$\forall s, a \quad \lambda_{sa} \geq 0 \tag{A.4}$$

$$\forall s, a \quad \lambda_{sa} \pi(a|s) = 0 \tag{A.5}$$

$$\forall s, a \quad \frac{\partial L(\pi, c, \lambda)}{\partial \pi(a|s)} = 0 \tag{A.6}$$

where c and λ are Lagrangian multipliers for the equality and inequality constraints, respectively, and (A.3) is the feasibility of primal variables, (A.4) is the feasibility of dual variables, (A.5) is the complementary slackness and (A.6) is the stationarity condition. The Lagrangian function of (A.1) is written as follows:

$$\begin{aligned}&L(\pi, c, \lambda)\\&= -J_{\pi}^{sp} + \sum_{s} c_s \left(\sum_{a'} \pi(a'|s) - 1 \right) - \sum_{s,a} \lambda_{sa} \pi(a|s)\end{aligned}$$

where the maximization of (A.1) is changed into the minimization problem, i.e., $\min_{\pi} -J_{\pi}^{sp}$. First, the derivative of J_{π}^{sp} can be obtained by using the chain rule.

$$\begin{aligned}&\frac{\partial J_{\pi}}{\partial \pi(a|s)} = d^{\intercal} G_{\pi}^{-1} \frac{\partial r_{\pi}^{sp}}{\partial \pi(a|s)} + \gamma d^{\intercal} G_{\pi}^{-1} \frac{\partial T_{\pi}}{\partial \pi(a|s)} G_{\pi}^{-1} r_{\pi}^{sp}\\&= \rho_{\pi}^{\intercal} \frac{\partial r_{\pi}^{sp}}{\partial \pi(a|s)} + \gamma \rho_{\pi}^{\intercal} \frac{\partial T_{\pi}}{\partial \pi(a|s)} V_{\pi}^{sp}\\&= \rho_{\pi}(s) \left(r(s,a) + \frac{\alpha}{2} - \alpha \pi(a|s) + \gamma \sum_{s'} V_{\pi}^{sp}(s') T(s'|s,a) \right)\\&= \rho_{\pi}(s) \left(Q_{\pi}^{sp}(s,a) + \frac{\alpha}{2} - \alpha \pi(a|s) \right).\end{aligned}$$

Here, the partial derivative of Lagrangian is obtained as follows:

$$\begin{aligned}&\frac{\partial L(\pi, c, \lambda)}{\partial \pi(a|s)}\\&= -\rho_{\pi}(s)(Q_{\pi}^{sp}(s,a) + \frac{\alpha}{2} - \alpha \pi(a|s)) + c_s - \lambda_{sa} = 0.\end{aligned}$$

First, consider a positive $\pi(a|s)$ where the corresponding Lagrangian multiplier λ_{sa} is zero due to the complementary slackness. By summing $\pi(a|s)$ with respect to action a, Lagrangian multiplier c_s can be obtained as follows:

$$\begin{aligned}&0 = -\rho_{\pi}(s)(Q_{\pi}^{sp}(s,a) + \frac{\alpha}{2} - \alpha \pi(a|s)) + c_s\\&\pi(a|s) = \left(-\frac{c_s}{\rho_{\pi}(s)\alpha} + \frac{1}{2} + \frac{Q_{\pi}^{sp}(s,a)}{\alpha} \right)\\&\sum_{\pi(a'|s)>0} \pi(a'|s) = \sum_{\pi(a'|s)>0} \left(-\frac{c_s}{\rho_{\pi}(s)\alpha} + \frac{1}{2} + \frac{Q_{\pi}^{sp}(s,a')}{\alpha} \right) = 1\\&\therefore c_s = \rho_{\pi}(s)\alpha \left[\frac{\sum_{\pi(a'|s)>0} \frac{Q_{\pi}^{sp}(s,a')}{\alpha} - 1}{K} + \frac{1}{2} \right]\end{aligned}$$

where K is the number of positive elements of $\pi(\cdot|s)$. By replacing c_s with this result, the optimal policy distribution is induced as follows.

$$\begin{aligned}\pi(a|s) &= \left(-\frac{c_s}{\rho_\pi(s)\alpha} + \frac{1}{2} + \frac{Q_\pi^{sp}(s,a)}{\alpha}\right) \\ &= \frac{Q_\pi^{sp}(s,a)}{\alpha} - \frac{\sum_{\pi(a'|s)>0} \frac{Q_\pi^{sp}(s,a')}{\alpha} - 1}{K}\end{aligned}$$

As this equation is derived under the assumption that $\pi(a|s)$ is positive. For $\pi(a|s) > 0$, following condition is necessarily fulfilled,

$$\frac{Q_\pi^{sp}(s,a)}{\alpha} > \frac{\sum_{\pi(a'|s)>0} \frac{Q_\pi^{sp}(s,a')}{\alpha} - 1}{K}.$$

We notate this supporting set as $S(s) = \{a|1 + K\frac{Q_\pi^{sp}(s,a)}{\alpha} > \sum_{\pi(a'|s)>0} \frac{Q_\pi^{sp}(s,a')}{\alpha}\}$. $S(s)$ contains the actions which has larger action values than threshold

$$\tau(Q_\pi^{sp}(s,\cdot)) = \frac{\sum_{\pi(a'|s)>0} \frac{Q_\pi^{sp}(s,a')}{\alpha} - 1}{K}.$$

By using these notations, the optimal policy distribution can be rewritten as follows:

$$\pi(a|s) = \max\left(\frac{Q_\pi^{sp}(s,a)}{\alpha} - \tau\left(\frac{Q_\pi^{sp}(s,\cdot)}{\alpha}\right), 0\right).$$

By substituting $\pi(a|s)$ with this result, the following optimality equation of V_π^{sp}

is induced.

$$
\begin{aligned}
&V_\pi^{sp}(s)\\
&=\sum_a \pi(a|s)\left(Q_\pi^{sp}(s,a)+\frac{\alpha}{2}(1-\pi(a|s))\right)\\
&=\sum_a \pi(a|s)\left(Q_\pi^{sp}(s,a)-\frac{\alpha}{2}\pi(a|s)\right)+\frac{\alpha}{2}\sum_a \pi(a|s)\\
&=\sum_{a\in S(s)} \pi(a|s)\\
&\times\left(Q_\pi^{sp}(s,a)-\frac{\alpha}{2}\left(\frac{Q_\pi^{sp}(s,a)}{\alpha}-\tau\left(\frac{Q_\pi^{sp}(s,\cdot)}{\alpha}\right)\right)\right)+\frac{\alpha}{2}\\
&=\sum_{a\in S(s)} \pi(a|s)\frac{\alpha}{2}\left(\frac{Q_\pi^{sp}(s,a)}{\alpha}+\tau\left(\frac{Q_\pi^{sp}(s,\cdot)}{\alpha}\right)\right)+\frac{\alpha}{2}\\
&=\alpha\left[\frac{1}{2}\sum_{a\in S(s)}^{K}\left(\left(\frac{Q_\pi^{sp}(s,a)}{\alpha}\right)^2-\tau\left(\frac{Q_\pi^{sp}(s,\cdot)}{\alpha}\right)^2\right)+\frac{1}{2}\right]
\end{aligned}
$$

To summarize, we obtain the sparse Bellman equation as follows:

$$
\begin{aligned}
Q_\pi^{sp}(s,a)&=r(s,a)+\gamma\sum_{s'}V_\pi^{sp}(s')T(s'|s,a)\\
V_\pi^{sp}(s)&=\alpha\left[\frac{1}{2}\sum_{a\in S(s)}^{K}\left(\left(\frac{Q_\pi^{sp}(s,a)}{\alpha}\right)^2-\tau\left(\frac{Q_\pi^{sp}(s,\cdot)}{\alpha}\right)^2\right)+\frac{1}{2}\right]\\
\pi(a|s)&=\max\left(\frac{Q_\pi^{sp}(s,a)}{\alpha}-\tau\left(\frac{Q_\pi^{sp}(s,\cdot)}{\alpha}\right),0\right).
\end{aligned}
$$

where the final equations can be obtained by change $r(s,a)$ into $r(s,a,s')$. □

A.3 Sparse Tsallis Entropy

In this section, the connection between $W(\pi)$ and Tsallis entropy is explained. The Tsallis entropy is defined as follows:

$$S_{q,k}(p)=\frac{k}{q-1}\left(1-\sum_i p_i^q\right),$$

where p is a probability mass function, q is a parameter called *entropic-index*, and k is a positive real constant.

The following theorem shows that $W(\pi)$ is equivalent to the discounted expected sum of special case of Tsallis entropy when $q = 2$ and $k = \frac{1}{2}$.

Proof of Theorem 2. The proof is simply done by rewriting our regularization as follows:

$$\begin{aligned}
&W(\pi)\\
&= \mathbb{E}\left[\sum_{t=0}^{\infty}\gamma^t\frac{1}{2}(1-\pi(a_t|s_t))\middle|\pi,d,T\right]\\
&= \sum_{s,a}\frac{1}{2}(1-\pi(a|s))\mathbb{E}\left[\sum_{t=0}^{\infty}\gamma^t\mathbb{I}_{\{s_t=s,a_t=a\}}\middle|\pi,d,T\right]\\
&= \sum_{s,a}\frac{1}{2}(1-\pi(a|s))\rho_\pi(s,a)\\
&= \sum_{s}\rho_\pi(s)\sum_a\frac{1}{2}(1-\pi(a|s))\pi(a|s)\\
&= \sum_{s}\rho_\pi(s)\frac{1}{2}(\sum_a\pi(a|s)-\sum_a\pi(a|s)^2)\\
&= \sum_{s}\rho_\pi(s)\frac{1}{2}(1-\sum_a\pi(a|s)^2)\\
&= \sum_{s}S_{2,\frac{1}{2}}(\pi(\cdot|s))\rho_\pi(s) = \mathbb{E}_\pi\left[S_{2,\frac{1}{2}}(\pi(\cdot|s))\right].
\end{aligned}$$

□

A.4 Upper and Lower Bounds for Sparsemax Operation

In this section, we prove the lower and upper bounds of spmax(z) defined as

$$\text{spmax}(z) \triangleq \frac{1}{2}\sum_{i=1}^{K}\left(z_{(i)}^2-\tau(z)^2\right)+\frac{1}{2}. \tag{A.7}$$

The lower bound and upper bound of spmax(z) is as follows,

$$\max(z) \le \alpha\text{spmax}(\frac{z}{\alpha}) \le \max(z)+\alpha\frac{d-1}{2d}. \tag{A.8}$$

Note that the proof of lower bound of (A.8) is provided in [85]. However, we find another interesting way to prove (A.8) by using the Cauchy-Schwartz inequality and the nonnegative property of a quadratic equation.

We first prove $\max(z) \leq \text{spmax}(z)$ and next prove $\text{spmax}(z) \leq \max(z) + \frac{d-1}{2d}$. For simplicity of derivation, we assume that $\alpha = 1$ but the original inequalities can be simply obtained by replacing z with $\frac{z}{\alpha}$.

Lemma 10. *For all* $z \in \mathbb{R}^d$, $\max(z) \leq \text{spmax}(z)$ *holds.*

Proof. We prove that, for all z, $\text{spmax}(z) - z_{(1)} \geq 0$ where $z_{(1)} = \max(z)$ by definition. The proof is done by simply rearranging the terms in (A.7),

$$\begin{aligned}
&\text{spmax}(z) - z_{(1)} \\
&= \frac{1}{2}\sum_{i=1}^{K}\left(z_{(i)}^2 - \tau(z)^2\right) + \frac{1}{2} - z_{(1)} \\
&= \frac{1}{2}\sum_{i=1}^{K} z_{(i)}^2 - \frac{K}{2}\left(\frac{\sum_{i=1}^{K} z_{(i)} - 1}{K}\right)^2 + \frac{1}{2} - z_{(1)} \\
&= \frac{1}{2}\sum_{i=1}^{K} z_{(i)}^2 - \frac{1}{2K}\left(\sum_{i=1}^{K} z_{(i)} - 1\right)^2 + \frac{1}{2} - z_{(1)} \\
&= \frac{K\sum_{i=1}^{K} z_{(i)}^2 - \left(\sum_{i=1}^{K} z_{(i)} - 1\right)^2 - 2Kz_{(1)} + K}{2K} \\
&= \frac{1}{2K}\left(Kz_{(1)}^2 + K\sum_{i=2}^{K} z_{(i)}^2\right. \\
&\left. - \left(z_{(1)} + \sum_{i=2}^{K} z_{(i)} - 1\right)^2 - 2Kz_{(1)} + K\right).
\end{aligned}$$

The quadratic term can be decomposed as follows:

$$\begin{aligned}
&\left(z_{(1)} + \sum_{i=2}^{K} z_{(i)} - 1\right)^2 \\
&= z_{(1)}^2 + \left(\sum_{i=2}^{K} z_{(i)}\right)^2 + 1 + 2z_{(1)}\sum_{i=2}^{K} z_{(i)} - 2z_{(1)} - 2\sum_{i=2}^{K} z_{(i)}.
\end{aligned}$$

By putting this result into the equation and rearranging them, three terms are obtained as follows:

$$
\begin{aligned}
&\operatorname{spmax}(z) - z_{(1)} \\
&= \frac{1}{2K}\Bigg((K-1)z_{(1)}^2 - 2z_{(1)}\left\{\sum_{i=2}^{K} z_{(i)} + K - 1\right\} \\
&+ K\sum_{i=2}^{K} z_{(i)}^2 + 2\sum_{i=2}^{K} z_{(i)} + K - \left(\sum_{i=2}^{K} z_{(i)}\right)^2 \Bigg).
\end{aligned}
$$

Then, $K\sum_{i=2}^{K} z_{(i)}^2 + 2\sum_{i=2}^{K} z_{(i)} + K$ can be replaced with $K\sum_{i=2}^{K}\left(z_{(i)}+1\right)^2 - 2(K-1)\sum_{i=2}^{K} z_{(i)}$ and we also decompose the second term $-2z_{(1)}\left\{\sum_{i=2}^{K} z_{(i)} + K - 1\right\}$ into two parts: $-2z_{(1)}\left\{\sum_{i=2}^{K}(z_{(i)}+1)\right\}$ and $2z_{(1)}$, and rearrange the equation as follows,

$$
\begin{aligned}
&= \frac{1}{2K}\Bigg((K-1)z_{(1)}^2 - 2z_{(1)}\left\{\sum_{i=2}^{K}\left(z_{(i)}+1\right)\right\} \\
&+ K\sum_{i=2}^{K}\left(z_{(i)}+1\right)^2 - 2(K-1)\sum_{i=2}^{K} z_{(i)} - \left(\sum_{i=2}^{K} z_{(i)}\right)^2 \Bigg).
\end{aligned}
$$

Again, we change $-2(K-1)\sum_{i=2}^{K} z_{(i)} - \left(\sum_{i=2}^{K} z_{(i)}\right)^2$ into $-\left(\sum_{i=2}^{K}(z_{(i)}+1)\right)^2 + (K-1)^2$ by adding and subtracting $(K-1)^2$ as follow,

$$
\begin{aligned}
&= \frac{1}{2K}\Bigg((K-1)z_{(1)}^2 - 2z_{(1)}\left\{\sum_{i=2}^{K}\left(z_{(i)}+1\right)\right\} \\
&+ K\sum_{i=2}^{K}\left(z_{(i)}+1\right)^2 - \left(\sum_{i=2}^{K}(z_{(i)}+1)\right)^2 + (K-1)^2 \Bigg).
\end{aligned}
$$

Then, the term $(K-1)z_{(1)}^2 - 2z_{(1)}\left\{\sum_{i=2}^{K}\left(z_{(i)}+1\right)\right\}$ is reformulated as $(K-1)\left(z_{(1)} - \frac{\sum_{i=2}^{K}\left(z_{(i)}+1\right)}{K-1}\right)^2 - (K-1)\left(\frac{\sum_{i=2}^{K}(z_{(i)}+1)}{K-1}\right)^2$. By using this reformulation, we can obtain following equation.

$$
\begin{aligned}
&= \frac{(K-1)}{2K}\left[z_{(1)} - \frac{\sum_{i=2}^{K}\left(z_{(i)}+1\right)}{K-1}\right]^2 + \\
&\frac{1}{2K}\left(- \frac{\left(\sum_{i=2}^{K}(z_{(i)+1})\right)^2}{K-1} + K\sum_{i=2}^{K}\left(z_{(i)}+1\right)^2 - \left(\sum_{i=2}^{K}(z_{(i)}+1)\right)^2 \right. \\
&\left. + (K-1)^2 \right).
\end{aligned}
$$

Finally, we can obtain three terms by rearranging the above equation,

$$
\begin{aligned}
&= \frac{(K-1)}{2K}\left[z_{(1)} - \frac{\sum_{i=2}^{K}\left(z_{(i)}+1\right)}{K-1}\right]^2 \\
&+ \frac{1}{2K}\left(K\sum_{i=2}^{K}\left(z_{(i)}+1\right)^2 - K\frac{\left(\sum_{i=2}^{K}(z_{(i)}+1)\right)^2}{K-1}\right) + \frac{(K-1)^2}{2K} \\
&= \frac{(K-1)}{2K}\left[z_{(1)} - \frac{\sum_{i=2}^{K}\left(z_{(i)}+1\right)}{K-1}\right]^2 \\
&+ \frac{K-1}{2}\left[\sum_{i=2}^{K}\frac{\left(z_{(i)}+1\right)^2}{K-1} - \left(\sum_{i=2}^{K}\frac{\left(z_{(i)}+1\right)}{K-1}\right)^2\right] + \frac{(K-1)^2}{2K}
\end{aligned}
$$

where the first and third terms are quadratic and always nonnegative. The second term is also always nonnegative by the Cauchy-Schwartz inequality. The Cauchy-Schwartz inequality is written as $(\mathbf{p}^\intercal\mathbf{q})^2 \leq ||\mathbf{p}||^2||\mathbf{q}||^2$. Let $z_{2:K} = [z_{(2)}, \cdots, z_{(K)}]^\intercal$, then, by setting $\mathbf{p} = z_{2:K} + \mathbb{I}$ and $\mathbf{q} = \frac{1}{K-1}\mathbb{I}$ where $\mathbb{I}$ is a $K-1$ dimensional vector of ones, it can be shown that the second term is nonnegative. Therefore, $\operatorname{spmax}(z) - z_{(1)}$ is always nonnegative for all z since three remaining terms are always nonnegative, completing the proof. □

Now, we prove the upper bound of sparsemax operation.

Lemma 11. *For all $z \in \mathbb{R}^d$, $\operatorname{spmax}(z) \leq \max(z) + \frac{d-1}{2d}$ holds.*

Proof. First, we decompose the summation of (A.7) into two terms as follows:

$$
\begin{aligned}
\mathrm{spmax}(z) &= \frac{1}{2}\sum_{i=1}^{K}\left(z_{(i)}^2 - \tau(z)^2\right) + \frac{1}{2} \\
&= \frac{1}{2}\sum_{i=1}^{K}\left(z_{(i)} - \tau(z)\right)\left(z_{(i)} + \tau(z)\right) + \frac{1}{2} \\
&\leq \frac{1}{2}\sum_{i=1}^{K} p_i^*(z)\left(z_{(i)} + \tau(z)\right) + \frac{1}{2} \\
&= \frac{1}{2}\sum_{i=1}^{K} p_i^*(z) z_{(i)} + \frac{\tau(z)}{2}\sum_{i=1}^{K} p_i^*(z) + \frac{1}{2} \\
&= \frac{1}{2}\sum_{i=1}^{K} p_i^*(z) z_{(i)} + \frac{\tau(z)}{2} + \frac{1}{2} \\
&= \frac{1}{2}\sum_{i=1}^{K} p_i^*(z) z_{(i)} + \frac{1}{2}\sum_{i=1}^{K}\frac{z_{(i)}}{K} - \frac{1}{2K} + \frac{1}{2}
\end{aligned}
$$

where $p_i^* = \max(z_{(i)} - \tau(z), 0)$ which is the optimal solution of the simplex projection problem and $\sum_{i=1}^{K} p_i^*(z) = 1$ by definition. Now, we use the fact that, for every p on $d-1$ dimensional simplex, $\sum_i^d p_i z_i \leq \max(z)$ for all $z \in \mathbb{R}^d$. By using this property, as $p^*(z)$ and $\frac{1}{K}\mathbb{I}$ are on the probability simplex, following inequality is induced,

$$
\begin{aligned}
&\mathrm{spmax}(z) = \frac{1}{2}\sum_{i=1}^{K} p_i^*(z) z_{(i)} + \frac{1}{2}\sum_{i=1}^{K}\frac{z_{(i)}}{K} - \frac{1}{2K} + \frac{1}{2} \\
&\leq \frac{1}{2}\max(z) + \frac{1}{2}\max(z) - \frac{1}{2K} + \frac{1}{2} \leq \max(z) - \frac{1}{2K} + \frac{1}{2} \\
&\leq \max(z) - \frac{1}{2d} + \frac{1}{2}
\end{aligned}
$$

where $d \geq K$ by definition of K. Therefore, $\mathrm{spmax}(z) \leq \max(z) + \frac{d-1}{2d}$ holds. □

A.5 Comparison to *Log-Sum-Exp*

We explain the error bounds for the *log-sum-exp* operation and compare it to the bounds of the sparsemax operation. The *log-sum-exp* operation has widely known bounds,

$$\max(z) \leq \text{logsumexp}(z) \leq \max(z) + \log(d).$$

We would like to note that *sparsemax* has tighter bounds than *log-sum-exp* as it is always satisfied that, for all $d > 1$, $\frac{d-1}{2d} \leq \log(d)$. Intuitively, the approximation error of *log-sum-exp* increases as the dimension of input space increases. However, the approximation error of *sparsemax* approaches to $\frac{1}{2}$ as the dimension of input space goes infinity. This fact plays a crucial role in comparing performance error bounds of the sparse MDP and soft MDP.

A.6 Convergence and Optimality of Sparse Value Iteration

In this section, the monotonicity, discounting property, contraction of sparse Bellman operation U^{sp} are proved.

Proof of Lemma 1. In [85], the monotonicity of (A.7) is proved. Then, the monotonicity of U^{sp} can be proved using (A.7). Let x and y are given such that $x \leq y$. Then,

$$\frac{r(s,a) + \gamma \sum_{s'} x(s')T(s'|s,a)}{\alpha} \leq \frac{r(s,a) + \gamma \sum_{s'} y(s')T(s'|s,a)}{\alpha}$$

where $T(s'|s,a)$ is a transition probability which is always nonnegative. Since the sparsemax operation is monotone, the following inequality is induced

$$\begin{aligned}&\alpha\text{spmax}\left(\frac{r(s,a) + \gamma \sum_{s'} x(s')T(s'|s,a)}{\alpha}\right)\\ &\leq \alpha\text{spmax}\left(\frac{r(s,a) + \gamma \sum_{s'} y(s')T(s'|s,a)}{\alpha}\right).\end{aligned}$$

Finally, we can obtain

$$\therefore\ \ U^{sp}(x) \leq U^{sp}(y).$$

□

Proof of Lemma 2. In [85], it is shown that for $c \in \mathbb{R}$ and $x \in \mathbb{R}^{|\mathcal{S}|}$, $\text{spmax}(x + c\mathbb{I}) = \text{spmax}(x) + c\mathbb{I}$. Using this property,

$$\begin{aligned}
&U^{sp}(x + c\mathbb{I})(s)\\
&= \alpha\text{spmax}\left(\frac{r(s,a) + \gamma\sum_{s'}(x(s') + c)T(s'|s,a)}{\alpha}\right)\\
&= \alpha\text{spmax}\left(\frac{r(s,a) + \gamma\sum_{s'}x(s')T(s'|s,a) + \gamma c\sum_{s'}T(s'|s,a)}{\alpha}\right)\\
&= \alpha\text{spmax}\left(\frac{r(s,a) + \gamma\sum_{s'}x(s')T(s'|s,a)}{\alpha} + \frac{\gamma c}{\alpha}\right)\\
&= \alpha\text{spmax}\left(\frac{r(s,a) + \gamma\sum_{s'}x(s')T(s'|s,a)}{\alpha}\right) + \gamma c\\
&\therefore\ \ U^{sp}(x + c\mathbb{I}) = U^{sp}(x) + \gamma c\mathbb{I}.
\end{aligned}$$

□

Proof of Lemma 3. First, we prove that U^{sp} is a γ-contraction mapping with respect to d_{max}. Without loss of generality, the proof is discussed for a general function $\phi : \mathbb{R}^{|\mathcal{S}|} \rightarrow \mathbb{R}^{|\mathcal{S}|}$ with discounting and monotone properties.

Let $d_{max}(x, y) = M$. Then, $y - M\mathbb{I} \leq x \leq y + M\mathbb{I}$ is satisfied. By monotone and discounting properties, the following inequality between mappings $\phi(x)$ and $\phi(y)$ is established.

$$\phi(y) - \gamma M\mathbb{I} \leq \phi(x) \leq \phi(y) + \gamma M\mathbb{I},$$

where γ is a discounting factor of ϕ. From this inequality, $d_{max}(\phi(x), \phi(y)) \leq \gamma M = \gamma d_{max}(x, y)$ and $\gamma \in (0, 1)$. Therefore, ϕ is a γ-contraction mapping. In our case, U^{sp} is a γ-contraction mapping.

As $\mathbb{R}^{|\mathcal{S}|}$ and $d_{max}(x, y)$ are a non-empty complete metric space, by Banach fixed-point theorem, a γ-contraction mapping U^{sp} has a unique fixed point. □

Using Lemma 1, Lemma 2, and Lemma 3, we can prove the convergence and optimality of sparse value iteration.

Proof of Theorem 3. Sparse value iteration converges into a fixed point of U^{sp} by the contraction property. Let x_* be a fixed point of U^{sp} and, by definition of U^{sp}, x_* is the point that satisfies the sparse Bellman equation, i.e. $x_* = U^{sp}(x_*)$. Hence, by Theorem 1, x_* satisfies necessity conditions of the optimal solution. By the Banach fixed point theorem, x_* is a unique point which satisfies necessity conditions of optimal solution. In particular, $x_* = U^{sp}(x_*)$ is precisely equivalent to the sparse Bellman equation. In other words, there is no other point that satisfies the sparse Bellman equation. Therefore, x_* is the optimal value of a sparse MDP. □

A.7 Performance Error Bounds for Sparse Value Iteration

In this section, we prove the performance error bounds for sparse value iteration and soft value iteration. We first show that the optimal vlaues of a sparse MDP and a soft MDP are greater than that of the original MDP.

Proof of Lemma 4. We first prove the inequality of the sparse Bellman operation

$$U^n(x) \leq (U^{sp})^n(x), \;\; x_* \leq x_*^{sp}.$$

This inequality can be proven by the mathematical induction. When $n = 1$, the inequality is proven as follows:

$$\max_{a'} \left(r(s,a') + \gamma \textstyle\sum_{s'} x(s')T(s'|s,a')\right)$$
$$\leq \operatorname{spmax}\left(r(s,\cdot) + \gamma \textstyle\sum_{s'} x(s')T(s'|s,\cdot)\right)$$
$$(\because \max(z) \leq \operatorname{spmax}(z)).$$

Therefore,

$$U(x) \leq U^{sp}(x).$$

For some positive integer k, let us assume that $U^k(x) \leq (U^{sp})^k(x)$ holds for every $x \in \mathbb{R}^{|\mathcal{S}|}$. Then, when $n = k+1$,

$$\begin{aligned} U^{k+1}(x) &= U^k(U(x)) \\ &\leq (U^{sp})^k(U(x)) \;\; (\because U^k(x) \leq (U^{sp})^k(x)) \\ &\leq (U^{sp})^k(U^{sp}(x)) \;\; (\because U(x) \leq U^{sp}(x)) \\ &= (U^{sp})^{k+1}(x). \end{aligned}$$

Therefore, by mathematical induction, it is satisfied $U^n(x) \leq (U^{sp})^n(x)$ for every positive integer n. Then, the inequality of the fixed points of U and U^{sp} can be obtained by $n \to \infty$,

$$x_* \leq x_*^{sp}$$

where $*$ indicates the fixed point. The above arguments also hold when U^{sp} and *sparsemax* are replaced with U^{soft} and *log-sum-exp* operation, respectively. □

Before showing the performance error bounds, the upper bounds of $W(\pi)$ and $H(\pi)$ are proved first.

Proof of Lemma 5. For $W(\pi)$,

$$\begin{aligned} &W(\pi) = \sum_s \rho_\pi(s) \sum_a \frac{1}{2}(1 - \pi(a|s))\pi(a|s) \\ &\leq \sum_s \rho_\pi(s) \frac{|\mathcal{A}| - 1}{2|\mathcal{A}|} \;\; (\because \sum_a \frac{1}{2}(1 - \pi(a|s))\pi(a|s) \leq \frac{|\mathcal{A}| - 1}{2|\mathcal{A}|}) \\ &= \frac{1}{1-\gamma} \frac{|\mathcal{A}| - 1}{2|\mathcal{A}|} \;\; (\because \sum_s \rho_\pi(s) = \frac{1}{1-\gamma}). \end{aligned}$$

The inequality that $\sum_a \frac{1}{2}(1-\pi(a|s))\pi(a|s) \leq \frac{|\mathcal{A}|-1}{2|\mathcal{A}|}$ can be obtained by finding the point where the derivative of $\frac{1}{2}(1-x)x$ is zero. Similarly, for $H(\pi)$,

$$\begin{aligned}
H(\pi) &= \mathbb{E}\left[\sum_{t=0}^{\infty}\gamma^t - \log(\pi(a_t|s_t))\middle|\pi, d, T\right] \\
&= \sum_{s,a} - \log(\pi(a|s))\mathbb{E}\left[\sum_{t=0}^{\infty}\gamma^t \mathbb{I}_{\{s_t=s,a_t=a\}}\middle|\pi, d, T\right] \\
&= \sum_{s,a} - \log(\pi(a|s))\rho_\pi(s,a) \\
&= \sum_{s}\rho_\pi(s)\sum_a - \log(\pi(a|s))\pi(a|s) \\
&\leq \sum_{s}\rho_\pi(s)\log(|\mathcal{A}|) \ \ (\because \sum_a - \log(\pi(a|s))\pi(a|s) \leq \log(|\mathcal{A}|)) \\
&= \frac{1}{1-\gamma}\log(|\mathcal{A}|) \ \ (\because \sum_s \rho_\pi(s) = \frac{1}{1-\gamma}).
\end{aligned}$$

The inequality that $\sum_a - \log(\pi(a|s))\pi(a|s) \leq \log(|\mathcal{A}|)$ also can be obtained by finding the point where the derivative of $-x\log(x)$ is zero. □

Using Lemma 4 and Lemma 5, the error bounds of sparse and soft value iterations can be proved.

Proof of Theorem 4. Let π_* be the optimal policy of the original MDP, where the problem is defined as $\max_\pi \mathbb{E}_\pi[\mathbf{r}(s,a)]$.

$$\mathbb{E}_{\pi_*^{sp}}[\mathbf{r}(s,a)] \leq \max_\pi \mathbb{E}_\pi[\mathbf{r}(s,a)] = \mathbb{E}_{\pi_*}[\mathbf{r}(s,a)].$$

The rightside inequality is by the definition of optimality. Before proving the leftside inequality, we first derive the following inequality from Lemma 4:

$$V_* \leq V_*^{sp}, \tag{A.9}$$

where $*$ indicates an optimal value. Since the fixed points of U and U^{sp} are the optimal solutions of the original MDP and sparse MDP, respectively, (A.9) can be derived from Lemma 4. The leftside inequality is proved using (A.9) as follows:

$$
\begin{aligned}
&\mathbb{E}_{\pi_*}(\mathbf{r}(s,a)) = d^\intercal V_* \\
&\leq d^\intercal V_*^{sp} = J_*^{sp} = \mathbb{E}_{\pi_*^{sp}}(\mathbf{r}(s,a)) + \alpha W(\pi_*^{sp}) \\
&\leq \mathbb{E}_{\pi_*^{sp}}(\mathbf{r}(s,a)) + \frac{\alpha}{1-\gamma}\frac{|\mathcal{A}|-1}{2|\mathcal{A}|} \ \ (\because \text{ Lemma 5}).
\end{aligned}
$$

□

Proof of Theorem 5. Let π_* be the optimal policy of the original MDP which is defined as $\max_\pi \mathbb{E}_\pi(\mathbf{r}(s,a))$. The rightside inequality is by the definition of optimality.

$$
\mathbb{E}_{\pi_*^{soft}}[\mathbf{r}(s,a)] \leq \max_\pi \mathbb{E}_\pi[\mathbf{r}(s,a)] = \mathbb{E}_{\pi_*}[\mathbf{r}(s,a)].
$$

Before proving the leftside inequality, we first derive following inequality from Lemma 4:

$$
V_* \leq V_*^{soft} \tag{A.10}
$$

where $*$ indicates an optimal solution. Then, the proof of the leftside inequality is done by using (A.10) as follows:

$$
\begin{aligned}
&\mathbb{E}_{\pi_*}(\mathbf{r}(s,a)) = d^\intercal V_* \\
&\leq d^\intercal V_*^{soft} = J_*^{soft} = \mathbb{E}_{\pi_*^{soft}}(\mathbf{r}(s,a)) + \alpha H(\pi_*^{soft}) \\
&\leq \mathbb{E}_{\pi_*^{soft}}(\mathbf{r}(s,a)) + \frac{\alpha}{1-\gamma}\log(|\mathcal{A}|) \ \ (\because \text{ Lemma 5}).
\end{aligned}
$$

□

Table A.1: Notations and Properties

Terms	sparse MDP	soft MDP
Utility	$J_\pi^{sp} \triangleq \mathbb{E}_\pi \left[\mathbf{r}(s',a') + \frac{\alpha}{2}(1-\pi(a'\|s'))\right] \quad = \sum_s d(s) V_\pi^{sp}(s) = \sum_s \mathbf{r}_\pi^{sp}(s)\rho_\pi(s)$	$J_\pi^{soft} \triangleq \mathbb{E}_\pi \left[\mathbf{r}(s',a') - \alpha \log(\pi(a'\|s'))\right] \quad = \sum_s d(s) V_\pi^{soft}(s) = \sum_s \mathbf{r}_\pi^{soft}(s)\rho_\pi(s)$
Value	$V_\pi^{sp}(s) \triangleq \mathbb{E}_\pi \left[\mathbf{r}(s',a') + \frac{\alpha}{2}(1-\pi(a'\|s'))\middle\|s_0 = s\right] = \mathbf{r}_\pi^{sp}(s) + \gamma \sum_{s'} V_\pi^{sp}(s') T_\pi(s'\|s)$	$V_\pi^{soft}(s) \triangleq \mathbb{E}_\pi \left[\mathbf{r}(s',a') - \alpha \log(\pi(a'\|s'))\|s_0 = s\right] = \mathbf{r}_\pi^{soft}(s) + \gamma \sum_{s'} V_\pi^{soft}(s') T_\pi(s'\|s)$
Action value	$Q_\pi^{sp}(s,a) \triangleq \qquad \mathbf{r}(s,a) + \gamma \sum_{s'} V_\pi^{sp}(s') T(s'\|s,a)$	$Q_\pi^{soft}(s,a) \triangleq \qquad \mathbf{r}(s,a) + \gamma \sum_{s'} V_\pi^{soft}(s') T(s'\|s,a)$
Expected State Reward	$\mathbf{r}_\pi^{sp}(s) \triangleq \qquad \sum_{a'} \left(\mathbf{r}(s,a') + \frac{\alpha}{2}(1-\pi(a'\|s))\right)\pi(a'\|s)$	$\mathbf{r}_\pi^{soft}(s) \triangleq \qquad \sum_{a'} \left(\mathbf{r}(s,a') - \alpha \log(\pi(a'\|s))\right)\pi(a'\|s)$
Policy Regularization	$W(\pi) \triangleq \mathbb{E}_\pi \left[\frac{1}{2}(1-\pi(a\|s))\right] \qquad = \sum_{s,a} \frac{1}{2}(1-\pi(a\|s))\pi(a\|s)\rho(s)$	$H(\pi) = \mathbb{E}_\pi[-\pi(a\|s)\log(\pi(a\|s))] \qquad = \sum_{s,a} -\pi(a\|s)\log(\pi(a\|s))\rho(s)$
Max Approximation	$\mathrm{spmax}(z) \triangleq \frac{1}{2}\sum_{i=1}^{K}\left(z_{(i)}^2 - \tau(z)^2\right) + \frac{1}{2}$	$\mathrm{logsumexp}\,(z) \triangleq \log \sum_i \exp(z_i)$
Value Iteration Operator	$U^{sp}(x)(s) = \alpha \mathrm{spmax}\left(\frac{r(s,\cdot)+\gamma\sum_{s'} x(s')T(s'\|s,\cdot)}{\alpha}\right)$	$U^{soft}(x)(s) = \alpha \mathrm{logsumexp}\left(\frac{r(s,\cdot)+\gamma\sum_{s'} x(s')T(s'\|s,\cdot)}{\alpha}\right)$
State Visitation	$\rho_\pi(s) \triangleq \mathbb{E}_\pi\left[\mathbb{I}_{\{s'=s\}}\right] = d(s) + \gamma \sum_{s',a'} T(s\|s',a')\rho_\pi(s',a')$	
State Action Visitation	$\rho_\pi(s,a) \triangleq \mathbb{E}_\pi\left[\mathbb{I}_{\{s'=s,a'=a\}}\right] = \pi(a\|s)d(s) + \gamma \sum_{s',a'} \pi(a\|s)T(s\|s',a')\rho_\pi(s',a')$	
Transition Probability given π	$T_\pi(s'\|s) \triangleq \sum_a T(s'\|s,a)\pi(a\|s)$	

Appendix B

Proofs of Chapter 3.2.

We consider the maximum causal Tsallis entropy problem defined as follows:

$$\begin{aligned} \underset{\pi}{\text{maximize}} \quad & \alpha W(\pi) \\ \text{subject to} \quad & \mathbb{E}_{\pi}\left[\phi(s,a)\right] = \mathbb{E}_{\pi_E}\left[\phi(s,a)\right], \\ & \forall\, s,a \;\; \sum_{a'} \pi(a'|s) = 1, \;\; \pi(a|s) \geq 0. \end{aligned} \tag{B.1}$$

Note that the constraints for Π are explicitly added. For the remainder of this supplementary material, we will explicitly write all constraints for Π and $\mathbf{M}$.

B.1 Change of Variables

Proof of Theorem 7. The proof is simply done by checking two equalities. First,

$$\begin{aligned} W(\pi) &= \frac{1}{2}\mathbb{E}_{\pi}\left[1-\pi(a|s)\right] = \frac{1}{2}\sum_{s,a}\rho_{\pi}(s,a)\left(1-\pi(a|s)\right) \\ &= \frac{1}{2}\sum_{s,a}\rho_{\pi}(s,a)\left(1-\frac{\rho_{\pi}(s,a)}{\sum_{a'}\rho_{\pi}(s,a')}\right) \end{aligned}$$

and, second,

$$\begin{aligned}\bar{W}(\rho) = \frac{1}{2}\sum_{s,a}\rho(s,a)\left(1 - \frac{\rho(s,a)}{\sum_{a'}\rho(s,a')}\right) = \frac{1}{2}\sum_{s,a}\rho_{\pi_\rho}(s,a)\left(1 - \pi_\rho(a|s)\right)\\ = W(\pi_\rho).\end{aligned}$$

□

Base on Theorem 6 and Theorem 12, we convert the problem (B.1) into

$$\begin{aligned}\underset{\rho}{\text{maximize}}\quad & \alpha\bar{W}(\rho)\\ \text{subject to}\quad & \sum_{s,a}\rho(s,a)\phi(s,a) = \sum_{s,a}\rho_E(s,a)\phi(s,a),\\ & \forall\ s,a,\ \ \rho(s,a)\geq 0,\ \sum_{a}\rho(s,a) = d(s) + \gamma\sum_{s',a'}T(s|s',a')\rho(s',a')\end{aligned}\tag{B.2}$$

where $\bar{W}(\rho) = W(\frac{\rho}{\sum_a \rho})$, the second constraints are Bellman flow constraints for $\mathbf{M}$, and ρ_E is the state action visitation corresponding to π_E.

B.2 Concavity of Maximum Causal Tsallis Entropy

The following theorem shows that the objective function $\bar{W}(\rho)$ of the problem (B.2) is a concave function.

Proof of Theorem 8. Proof of concavity of $\bar{W}(\rho)$ is equivalent to show that following inequality is satisfied for all state s and action a pairs:

$$\begin{aligned}&(\lambda_1\rho_1(s,a) + \lambda_2\rho_2(s,a))\left(1 - \frac{\lambda_1\rho_1(s,a) + \lambda_2\rho_2(s,a)}{\lambda_1\sum_{a'}\rho_1(s,a') + \lambda_2\sum_{a'}\rho_2(s,a')}\right)\\ &\geq \lambda_1\rho_1(s,a)\left(1 - \frac{\rho_1(s,a)}{\sum_{a'}\rho_1(s,a')}\right) + \lambda_2\rho_2(s,a)\left(1 - \frac{\rho_2(s,a)}{\sum_{a'}\rho_2(s,a')}\right)\end{aligned}$$

where $\lambda_1 \geq 0$, $\lambda_2 \geq 0$, and $\lambda_1 + \lambda_2 = 1$. For notational simplicity, $\rho_i(s,a)$ and $\sum_{a'}\rho_i(s,a')$ are replaced with a_i and b_i, respectively. Then, the right-hand side is

$$\begin{aligned}
\sum_{i=1,2} \lambda_i a_i \left(1 - \frac{a_i}{b_i}\right) &= \sum_{i=1,2} \lambda_i a_i \left(1 - \frac{\lambda_i a_i}{\lambda_i b_i}\right) \\
&= \left(\sum_{j=1,2} \lambda_j b_j\right) \sum_{i=1,2} \left[\frac{\lambda_i b_i}{\left(\sum_{j=1,2} \lambda_j b_j\right)} \frac{\lambda_i a_i}{\lambda_i b_i} \left(1 - \frac{\lambda_i a_i}{\lambda_i b_i}\right)\right].
\end{aligned}$$

Let $F(x) = x(1-x)$, which is a concave function. Then the above equation can be expressed as follows,

$$\sum_{i=1,2} \lambda_i a_i \left(1 - \frac{a_i}{b_i}\right) = \left(\sum_{j=1,2} \lambda_j b_j\right) \sum_{i=1,2} \left[\frac{\lambda_i b_i}{\left(\sum_{j=1,2} \lambda_j b_j\right)} F\left(\frac{\lambda_i a_i}{\lambda_i b_i}\right)\right].$$

By using the property of concave function $F(x)$[1], we obtain the following inequality:

$$\begin{aligned}
&\left(\sum_{j=1,2} \lambda_j b_j\right) \sum_{i=1,2} \left[\frac{\lambda_i b_i}{\left(\sum_{j=1,2} \lambda_j b_j\right)} F\left(\frac{\lambda_i a_i}{\lambda_i b_i}\right)\right] \\
&\leq \left(\sum_{j=1,2} \lambda_j b_j\right) F\left(\sum_{i=1,2} \left[\frac{\lambda_i b_i}{\left(\sum_{j=1,2} \lambda_j b_j\right)} \frac{\lambda_i a_i}{\lambda_i b_i}\right]\right) = \left(\sum_{j=1,2} \lambda_j b_j\right) F\left(\frac{\sum_{i=1,2} \lambda_i a_i}{\sum_{j=1,2} \lambda_j b_j}\right) \\
&= \left(\sum_{j=1,2} \lambda_j b_j\right) \frac{\sum_{i=1,2} \lambda_i a_i}{\sum_{j=1,2} \lambda_j b_j} \left(1 - \frac{\sum_{i=1,2} \lambda_i a_i}{\sum_{j=1,2} \lambda_j b_j}\right) = \sum_{i=1,2} \lambda_i a_i \left(1 - \frac{\sum_{i=1,2} \lambda_i a_i}{\sum_{j=1,2} \lambda_j b_j}\right).
\end{aligned}$$

Finally, we have the following inequality for every state and action pair,

$$\begin{aligned}
&(\lambda_1 \rho_1(s,a) + \lambda_2 \rho_2(s,a)) \left(1 - \frac{\lambda_1 \rho_1(s,a) + \lambda_2 \rho_2(s,a)}{\lambda_1 \sum_{a'} \rho_1(s,a') + \lambda_2 \sum_{a'} \rho_2(s,a')}\right) \\
&\geq \lambda_1 \rho_1(s,a) \left(1 - \frac{\rho_1(s,a)}{\sum_{a'} \rho_1(s,a')}\right) + \lambda_2 \rho_2(s,a) \left(1 - \frac{\rho_2(s,a)}{\sum_{a'} \rho_2(s,a')}\right),
\end{aligned}$$

and, by summing up with respect to s, a, we get

$$\bar{W}(\lambda_1 \rho_1 + \lambda_2 \rho_2) \geq \lambda_1 \bar{W}(\rho_1) + \lambda_2 \bar{W}(\rho_2).$$

Therefore, $\bar{W}(\rho)$ is a concave function. □

[1] $\sum_i \mu_i F(x_i) \leq F(\sum_i \mu_i x_i)$, for some $(x_i, \ldots, x_n)$ and $(\mu_i, \ldots, \mu_n)$ such that $\mu_i \geq 0$ and $\sum_i \mu_i = 1$.

Theorem 8 tells us that the problem (B.2) is a concave problem and, hence, strong duality holds. The dual problem can be found as follows:

$$\begin{aligned} &\max_{\theta,c,\lambda}\ \min_{\rho} \quad L_W(\theta, c, \lambda, \rho) \\ &\text{subject to} \quad \forall\, s, a, \ \ \lambda(s, a) \geq 0 \end{aligned} \tag{B.3}$$

where $L_W(\theta, c, \lambda, \rho) = -\alpha \bar{W}(\rho) - \sum_{s,a} \rho(s, a)\theta^{\intercal}\phi(s, a) + \sum_{s,a} \rho_E(s, a)\theta^{\intercal}\phi(s, a) - \sum_{s,a} \lambda_{sa}\rho(s, a) + \sum_s c_s \left(\sum_a \rho(s, a) - d(s) - \gamma \sum_{s',a'} T(s|s', a')\rho(s', a')\right)$ and θ, c, and λ are Lagrangian multipliers. Since strong duality holds, the optimal solutions of primal and dual variables necessarily and sufficiently satisfy the KKT conditions.

B.3 Optimality Condition of Maximum Causal Tsallis Entropy

The following theorem explains the optimality condition of the maximum causal Tsallis entropy problem and also tells us that the optimal policy distribution has a sparse and multi-modal distribution.

Proof of Theorem 9. These conditions are derived from the stationary condition of KKT, where the derivative of L_W is equal to zero,

$$\frac{\partial L_W}{\partial \rho(s, a)} = 0.$$

We first compute the derivative of $\bar{W}$ as follows:

$$\frac{\partial \bar{W}}{\partial \rho(s, a)} = \frac{1}{2} - \frac{\rho(s, a)}{\sum_{a'} \rho(s, a')} + \frac{1}{2} \sum_{a'} \left(\frac{\rho(s, a')}{\sum_{a'} \rho(s, a')} \right)^2 .$$

We also check the derivative of Bellman flow constraints as follows:

$$\begin{aligned}&\frac{\partial \sum_s c_s \left(\sum_{a'} \rho(s,a') - d(s) - \gamma \sum_{s',a'} T(s|s',a')\rho(s',a')\right)}{\partial \rho(s'',a'')} \\ &\qquad\qquad = c_{s''} - \gamma \sum_s c_s T(s|s'',a'').\end{aligned}$$

Hence, the stationary condition can be obtained as

$$\begin{aligned}\frac{\partial L_W}{\partial \rho(s,a)} =& \alpha \left[-\frac{1}{2} + \frac{\rho(s,a)}{\sum_{a'} \rho(s,a')} - \frac{1}{2} \sum_{a'} \left(\frac{\rho(s,a')}{\sum_{a'} \rho(s,a')} \right)^2 \right] - \theta^\intercal \phi(s,a) \\ &+ c_s - \gamma \sum_{s'} c_{s'} T(s'|s,a) - \lambda_{sa} = 0.\end{aligned} \tag{B.4}$$

First, let us consider a positive $a \in S(s) = \{a|\rho(s,a) > 0\}$. From the complementary slackness, the corresponding λ_{sa} is zero. By replacing $\frac{\rho(s,a)}{\sum_a' \rho(s,a')}$ with $\pi_\rho(a|s)$ and using the definition of q_{sa}, the following equation is obtained from the stationary condition (B.4).

$$\pi(a|s) - \frac{q_{sa}}{\alpha} = \frac{1}{2} + \frac{1}{2} \sum_{a'} \left(\pi(a'|s)\right)^2 - \frac{c_s}{\alpha}. \tag{B.5}$$

It can be observed that the right hand side of the equation only depends on the state s and is constant for the action a. In this regards, by summing up with respect to the action with positive $\rho(s,a) > 0$, c_s is obtained as follows:

$$\begin{aligned}1 - \sum_{a \in S(s)} \frac{q_{sa}}{\alpha} &= K \left(\frac{1}{2} + \frac{1}{2} \sum_{a'} \left(\pi(a'|s)\right)^2 - \frac{c_s}{\alpha} \right) \\ \frac{c_s}{\alpha} &= \frac{1}{2} + \frac{1}{2} \sum_{a'} \left(\pi(a'|s)\right)^2 + \frac{\sum_{a \in S(s)} \frac{q_{sa}}{\alpha} - 1}{K},\end{aligned}$$

where K is the number of actions with positive $\rho(s,a) > 0$. By plug in $\frac{c_s}{\alpha}$ into (B.5), we obtain a policy as follows:

$$\pi(a|s) = \frac{q_{sa}}{\alpha} - \left(\frac{\sum_{a \in S(s)} \frac{q_{sa}}{\alpha} - 1}{K} \right)$$

Now, we define $\tau(\frac{q_s}{\alpha}) \triangleq \frac{\sum_{a \in S(s)} \frac{q_{sa}}{\alpha} - 1}{K}$, and, interestingly, τ is the same as the threshold of a sparsemax distribution [85]. Then, we can obtain the optimality condition for the policy distribution $\pi(a|s)$ as follows:

$$\forall s, a \;\; \pi(a|s) = \max\left(\frac{q_{sa}}{\alpha} - \tau(s), 0\right).$$

where $\tau(s)$ indicates $\tau(\frac{q_s}{\alpha})$.

The Lagrangian multiplier c_s can be found from π as follows:

$$\begin{aligned}
\frac{c_s}{\alpha} &= \frac{1}{2} + \frac{1}{2}\sum_{a'} \left(\pi(a'|s)\right)^2 + \tau(s) \\
&= \frac{1}{2} + \frac{1}{2}\sum_{a' \in S(s)} \left(\frac{q_{sa'}}{\alpha} - \tau(s)\right)^2 + \tau(s) \\
&= \frac{1}{2} + \frac{1}{2}\sum_{a' \in S(s)} \left(\frac{q_{sa'}}{\alpha}\right)^2 - \sum_{a' \in S(s)} \frac{q_{sa'}}{\alpha}\tau(s) + \frac{K}{2}\tau(s)^2 + \tau(s) \\
&= \frac{1}{2} + \frac{1}{2}\sum_{a' \in S(s)} \left(\frac{q_{sa'}}{\alpha}\right)^2 - K\frac{\sum_{a' \in S(s)} \frac{q_{sa'}}{\alpha} - 1}{K}\tau(s) + \frac{K}{2}\tau(s)^2 \\
&= \frac{1}{2} + \frac{1}{2}\sum_{a' \in S(s)} \left(\frac{q_{sa'}}{\alpha}\right)^2 - \frac{K}{2}\tau(s)^2 \\
c_s &= \alpha\left[\frac{1}{2}\sum_{a \in S(s)} \left(\left(\frac{q_{sa}}{\alpha}\right)^2 - \tau\left(\frac{q_s}{\alpha}\right)^2\right) + \frac{1}{2}\right].
\end{aligned}$$

To summarize, we obtain the optimality condition of (B.2) as follows:

$$\begin{aligned}
&q_{sa} \triangleq \theta^\intercal \phi(s,a) + \gamma \sum_{s'} c_{s'} T(s'|s,a), \\
&c_s = \alpha\left[\frac{1}{2}\sum_{a \in S(s)} \left(\left(\frac{q_{sa}}{\alpha}\right)^2 - \tau\left(\frac{q_{s\cdot}}{\alpha}\right)^2\right) + \frac{1}{2}\right], \\
&\pi(a|s) = \max\left(\frac{q_{sa}}{\alpha} - \tau\left(\frac{q_{s\cdot}}{\alpha}\right), 0\right).
\end{aligned}$$

□

B.4 Interpretation as Robust Bayes

In this section, the connection between MCTE estimation and a minimax game between a decision maker and the nature is explained. We prove that the proposed MCTE problem is equivalent to a minimax game with the Brier score.

Proof of Theorem 10. The objective function can be reformulated as

$$\mathbb{E}_{\tilde{\pi}}\left[\sum_{a'}\frac{1}{2}\left(\mathbb{I}_{\{a'=a\}}-\pi(a'|s)\right)^2\right]=\mathbb{E}_{\tilde{\pi}}\left[B(s,a)\right]=\sum_{s,a}\rho_{\tilde{\pi}}(s,a)B(s,a)$$
$$=\frac{1}{2}\sum_{s,a}\rho_{\tilde{\pi}}(s,a)\left(1-2\pi(a|s)+\sum_{a'}\pi(a'|s)^2\right),$$

Hence, the objective function is quadratic with respect to $\pi(a|s)$ and is linear with respect to $\rho_{\tilde{\pi}}(s,a)$. By using the one-to-one correspondence between $\tilde{\pi}$ and $\rho_{\tilde{\pi}}$, we change the variable of inner maximization into the state action visitation. After changing the optimization variable, by using the minimax theorem [86], the minimization and maximization of the original problem are interchangeable as follows:

$$\min_{\pi\in\Pi}\max_{\rho_{\tilde{\pi}}\in\mathbf{M}}\mathbb{E}_{\tilde{\pi}}\left[\sum_{a'}\frac{1}{2}\left(\mathbb{I}_{\{a'=a\}}-\pi(a|s)\right)^2\right]$$
$$=\max_{\rho_{\tilde{\pi}}\in\mathbf{M}}\min_{\pi\in\Pi}\mathbb{E}_{\tilde{\pi}}\left[\sum_{a'}\frac{1}{2}\left(\mathbb{I}_{\{a'=a\}}-\pi(a|s)\right)^2\right]$$

where sum-to-one, positivity, and Bellman flow constraints are omitted here. After converting the problem, an optimal solution of the inner minimization with respect to π is easily computed as $\pi=\tilde{\pi}$ using $\nabla_{\pi(a''|s'')}\mathbb{E}_{\tilde{\pi}}\left[B(s,a)\right]=0$. After applying $\pi=\tilde{\pi}$ and recovering the variables from $\rho_{\tilde{\pi}}$ to $\tilde{\pi}$, the problem (3.13) is converted into

$$\max_{\tilde{\pi}\in\Pi}\frac{1}{2}\sum_{s}\rho_{\tilde{\pi}}(s)\left(1-\sum_{a}\tilde{\pi}(a|s)^2\right)=\max_{\tilde{\pi}\in\Pi}W(\tilde{\pi}),$$

where $\rho_{\tilde{\pi}}(s) = \sum_a \rho_{\tilde{\pi}}(s,a)$. Hence, the problem (3.13) is equivalent to the maximum causal Tsallis entropy problem. □

In summary, the policy found in the maximum causal Tsallis entropy problem can be interpreted as the optimal decision maker considering the worst nature in sense of the Brier score.

B.5 Generative Adversarial Setting with Maximum Causal Tsallis Entropy

In this section, we convert the maximum causal Tsallis entropy problem (B.3) into the generative adversarial setting by adding a reward regularization defined as follows:

$$\begin{aligned} \max_{\theta} \min_{\pi} \quad & -\alpha W(\pi) - \mathbb{E}_{\pi}\left[\theta^{\intercal}\phi(s,a)\right] + \mathbb{E}_{\pi_E}\left[\theta^{\intercal}\phi(s,a)\right] - \psi(\theta) \\ \text{subject to} \quad & \forall\, s,a \;\; \sum_{a'} \pi(a'|s) = 1, \;\; \pi(a|s) \geq 0 \end{aligned} \tag{B.6}$$

The proof consists of two parts. We first show that the maximization and minimization of the problem (B.6) are interchangable, which means that the solution of the maxi-min problem is equivalent to that of the mini-max problem.

Proof of Theorem 11. We first change the variable from π to ρ as follows:

$$\begin{aligned} \max_{\theta} \min_{\rho} \quad & -\alpha \bar{W}(\rho) - \theta^{\intercal} \sum_{s,a} \rho(s,a)\phi(s,a) - \theta^{\intercal} \sum_{s,a} \rho_E(s,a)\phi(s,a) - \psi(\theta) \\ \text{subject to} \quad & \forall s,a, \; \sum_{s,a} \rho(s,a)\phi(s,a) = \sum_{s,a} \rho_E(s,a)\phi(s,a), \\ & \rho(s,a) \geq 0, \;\; \sum_{a} \rho(s,a) = d(s) + \gamma \sum_{s',a'} T(s|s',a')\rho(s',a'), \end{aligned} \tag{B.7}$$

where ρ_E is ρ_{π_E}. Let

$$\bar{L}(\rho,\theta) \triangleq -\alpha \bar{W}(\rho) - \psi(\theta) - \theta^{\intercal} \sum_{s,a} \rho(s,a)\phi(s,a) + \theta^{\intercal} \sum_{s,a} \rho_E(s,a)\phi(s,a). \tag{B.8}$$

From Theorem 8, $\bar{W}(\rho)$ is a concave function with respect to ρ for a fixed θ. Hence, $\bar{L}(\rho,\theta)$ is also a concave function with respect to ρ for a fixed θ. From the convexity of ψ, $\bar{L}(\rho,\theta)$ is a convex function with respect to θ for a fixed ρ. Furthermore, the domain of ρ is compact and convex and the domain of θ is convex. Based on this property of $\bar{L}(\rho,\theta)$, we can use minimax duality [86]:

$$\max_{\theta}\min_{\rho}\ \bar{L}(\rho,\theta)=\min_{\rho}\max_{\theta}\ \bar{L}(\rho,\theta).$$

Hence, the maximization and minimization are interchangable. By using this fact, we have:

$$\begin{aligned}
&\max_{\theta}\min_{\rho}\ \bar{L}(\rho,\theta)=\min_{\rho}\max_{\theta}\ \bar{L}(\rho,\theta)\\
&=\min_{\rho}\ -\alpha\bar{W}(\rho)+\max_{\theta}\left(-\psi(\theta)+\theta^{\intercal}\sum_{s,a}\left(\rho(s,a)-\rho_E(s,a)\right)\phi(s,a)\right)\\
&=\min_{\rho}\ -\alpha\bar{W}(\rho)+\psi^{*}\left(\sum_{s,a}\left(\rho(s,a)-\rho_E(s,a)\right)\phi(s,a)\right)\\
&=\min_{\pi}\ \psi^{*}\left(\mathbb{E}_{\pi}\left[\phi(s,a)\right]-\mathbb{E}_{\pi_E}\left[\phi(s,a)\right]\right)-\alpha W(\pi)
\end{aligned}$$

□

B.6 Tsallis Entropy of a Mixture of Gaussians

The Tsallis entropy of a mixture of Gaussian distribution has an analytic.

Proof of Theorem 12. The causal Tsallis entropy of a mixture of Gaussian distribution can be obtained as follows:

$$
\begin{aligned}
W(\pi) &= \frac{1}{2}\sum_{s}\rho_{\pi}(s)\left(1-\int_{\mathcal{A}}\pi(a|s)^2\mathbf{d}a\right)\\
&= \frac{1}{2}\sum_{s}\rho_{\pi}(s)\left(1-\int_{\mathcal{A}}\left(\sum_{i}^{K}w_i(s)\mathcal{N}\left(a;\mu_i(s),\Sigma_i(s)\right)\right)^2\mathbf{d}a\right)\\
&= \frac{1}{2}\sum_{s}\rho_{\pi}(s)\\
&\times\left(1-\sum_{i}^{K}\sum_{j}^{K}w_i(s)w_j(s)\int_{\mathcal{A}}\mathcal{N}\left(a;\mu_i(s),\Sigma_i(s)\right)\mathcal{N}\left(a;\mu_j(s),\Sigma_j(s)\right)\mathbf{d}a\right)\\
&= \frac{1}{2}\sum_{s}\rho_{\pi}(s)\left(1-\sum_{i}^{K}\sum_{j}^{K}w_i(s)w_j(s)\mathcal{N}\left(\mu_i(s);\mu_j(s),\Sigma_i(s)+\Sigma_j(s)\right)\right)
\end{aligned}
\tag{B.9}
$$

□

B.7 Causal Entropy Approximation

In our implementation of maximum causal Tsallis entropy imitation learning (MCTEIL), we approximate $W(\pi)$ using sampled trajectories as follows:

$$
W(\pi) = \mathbb{E}_{\pi}\left[\frac{1}{2}\left(1-\pi(a|s)\right)\right] \approx \frac{1}{N}\sum_{i=0}^{N}\sum_{t=0}^{T_i}\frac{\gamma^t}{2}\left(1-\int_{\mathcal{A}}\pi(a|s_{i,t})^2\mathbf{d}a\right), \tag{B.10}
$$

where $\{(s_{i,t},a_{i,t})_{t=0}^{T_i}\}_{i=0}^{N}$ are N trajectories and T_i is the length of the ith trajectory. Since the integral part of (B.10) is analytically computed by Theorem 12, there is no additional computational cost. We have also tested the following approximation:

$$
W(\pi) = \mathbb{E}_{\pi}\left[\frac{1}{2}\left(1-\pi(a|s)\right)\right] \approx \frac{1}{N}\sum_{i=0}^{N}\sum_{t=0}^{T_i}\frac{\gamma^t}{2}\left(1-\pi(a_{i,t}|s_{i,t})\right).
$$

However, this approximation has performed poorly compared to (B.10).

For soft GAIL, $H(\pi)$ is approximated as the sum of discounted likelihoods

$$
H(\pi) = \mathbb{E}_{\pi}\left[-\log\left(\pi(a|s)\right)\right] \approx \frac{1}{N}\sum_{i=0}^{N}\sum_{t=0}^{T_i}-\gamma^t\log\left(\pi(a_{i,t}|s_{i,t})\right).
$$

Note that the same approximation (B.10) of $W(\pi)$ is not available for $H(\pi)$ since $-\int_{\mathcal{A}} \pi(a|s) \log (\pi(a|s))\, \mathbf{d}a$ is intractable when we model $\pi(a|s)$ as a mixture of Gaussians.

Appendix C

Proofs of Chapter 4.1.

We show that the Tsallis entropy is a concave function over the distribution P and has the maximum at an uniform distribution. Note that this is an well known fact, but, we restate it to make the manuscript self-contained.

Proposition 2. *Assume that $\mathcal{X}$ is a finite space. Let P is a probability distribution over $\mathcal{X}$. If $q > 0$, then, $S_q(P)$ is concave with respect to P.*

Proof. Let us consider the function $f(x) = -x \ln_q(x)$ defined over $(x > 0)$. Second derivative of $d^2f(x)/dx^2$ is computed as

$$\frac{d^2 f(x)}{dx^2} = -qx^{q-2} < 0 \ \ (x > 0, q > 0).$$

Thus, $f(x)$ is a concave function. Now, using this fact, we show that the following inequality holds. For $\lambda_1, \lambda_2 \geq 0$ such that $\lambda_1 + \lambda_2 = 1$, and probabilities P_1 and P_2,

$$\begin{aligned} S_q(\lambda_1 P_1 + \lambda_2 P_2) &= \sum_x -(\lambda_1 P_1(x) + \lambda_2 P_2(x)) \ln_q(\lambda_1 P_1(x) + \lambda_2 P_2(x)) \\ &< \sum_x -\lambda_1 P_1(x) \ln_q(P_1(x)) - \lambda_2 P_2(x) \ln_q(P_2(x)) \\ &= \lambda_1 S_q(P_1) + \lambda_2 S_q(P_2). \end{aligned}$$

Consequently, $S_q(P)$ is concave with respect to P. □

Proposition 3. *Assume that $\mathcal{X}$ is finite space. Then, $S_q(P)$ is maximized when P is a uniform distribution, i.e., $P = 1/|\mathcal{X}|$ where $|\mathcal{X}|$ is the number of elements in $\mathcal{X}$.*

Proof. We would like to employ the KKT condition on the following optimization problem.

$$\max_{P\in\Delta} S_q(P) \tag{C.1}$$

where $\Delta = \{P|P(x) \geq 0, \sum_x P(x) = 1\}$ is a probability simplex. Since $\mathcal{X}$ is finite, the optimization variables are probability mass defined over each element. The KKT condition of C.1 is

$$\begin{aligned}
&\forall x \in \mathcal{X}, \left.\frac{\partial\left(S_q(\pi) - \sum_x \lambda^\star(x)P(x) - \mu^\star\left(1 - \sum_x P(x)\right)\right)}{\partial P(x)}\right|_{P(x)=P^\star(x)} \\
&\qquad = -\ln_q(P^\star(x)) - (P^\star(x))^{q-1} - \lambda^\star(x) + \mu^\star \\
&\qquad = -q\ln_q(P^\star(x)) - 1 - \lambda^\star(x) + \mu^\star = 0 \\
&\forall x \in \mathcal{X}, 0 = 1 - \sum_x P^\star(x), P^\star(x) \geq 0 \\
&\forall x \in \mathcal{X}, \lambda^\star(x) \leq 0 \\
&\forall x \in \mathcal{X}, \lambda^\star(x)P^\star(x) = 0
\end{aligned}$$

where $\lambda^\star$ and $\mu^\star$ are the Lagrangian multipliers for constraints in Δ. First, let us consider $P^\star(x) > 0$. Then, $\lambda^\star(x) = 0$ from the last condition (complementary slackness). The first condition implies

$$P^\star(x) = \exp_q\left(\frac{\mu^\star - 1}{q}\right).$$

Hence, $P^\star(x)$ has constant probability mass which means $P^\star(x) = 1/|S|$ where $S = \{x|P^\star(x) > 0\}$. The optimal value is $S_q(P^\star) = -\ln_q(1/|S|)$. Since $-\ln_q(x)$ is a monotonically decreasing function, $|S|$ should be the largest number as possible as it can be. Hence, $S = \mathcal{X}$ and $P^{\star}(x) = 1/|\mathcal{X}|$. □

C.1 q-Maximum: Bounded Approximation of Maximum

Now, we prove the property of q-maximum which is defined by

Proof of Theorem 13. First, consider the lower bound. Let Δ be a probability simplex. Then,

$$\begin{aligned} q\text{-}\max_{x}(f(x)) &= \max_{P\in\Delta}\left[\mathop{\mathbb{E}}_{X\sim P}[f(X)] + S_q(P)\right] \leq \max_{P\in\Delta}\mathop{\mathbb{E}}_{X\sim P}[f(X)] + \max_{P\in\Delta} S_q(P) \\ &= \max_{x}(f(x)) - \ln_q\left(\frac{1}{|\mathcal{X}|}\right) \end{aligned} \tag{C.2}$$

where $S_q(P)$ has the maximum at an uniform distribution.

The upper bound can be proven using the similar technique. Let P' be the distribution whose probability is concentrated at a maximum element, which means if $x = \arg\max_{x'} f(x')$, then, $P'(x) = 1$ and, otherwise, $P'(x) = 0$. If there are multiple maximum at $f(x)$, then, one of them can be arbitrarily chosen. Then, the Tsallis entropy of P' becomes zero since all probability mass is concentrated at a single instance, i.e., $S_q(P') = 0$. Then, the upper bound can be computed as follows:

$$\begin{aligned} q\text{-}\max_{x}(f(x)) &= \max_{P\in\Delta}\left[\mathop{\mathbb{E}}_{X\sim P}[f(X)] + S_q(P)\right] \\ &\geq \mathop{\mathbb{E}}_{X\sim P'}[f(X)] + S_q(P') = f\left(\arg\max_{x'} f(x')\right) = \max_{x} f(x). \end{aligned} \tag{C.3}$$

□

We now analyze the solution of q-maximum operator.

Proposition 4. *The optimal solution of q-maximum operator is*

$$\pi_q^\star(a) = \exp_q\left(\frac{\mathbf{r}(a)}{q} - \psi_q\left(\frac{\mathbf{r}}{q}\right)\right), \tag{C.4}$$

where the q-potential function ψ_q is a functional defined on $\{\mathcal{A}, \mathbf{r}\}$. ψ_q is determined uniquely for given $\{\mathcal{A}, \mathbf{r}\}$ by the following normalization condition:

$$\sum_a \pi_q^\star(a) = \sum_a \exp_q\left(\frac{\mathbf{r}(a)}{q} - \psi_q\left(\frac{\mathbf{r}}{q}\right)\right) = 1. \tag{C.5}$$

Furthermore, using $\pi_q^\star$, the optimal value can be written as

$$\mathop{\mathbb{E}}_{a\sim\pi^\star}[R] + S_q(\pi^\star) = (q-1)\sum_a \frac{\mathbf{r}(a)}{q}\exp_q\left(\frac{\mathbf{r}(a)}{q} - \psi_q\left(\frac{\mathbf{r}}{q}\right)\right) + \psi_q\left(\frac{\mathbf{r}}{q}\right)\cdot \tag{C.6}$$

Proof. It is easy to check ψ_q exists uniquely for given $\{\mathcal{A}, \mathbf{r}\}$. Indeed, because $\exp_q \in [0, \infty)$ is a continuous monotonic function, for any $\{\mathcal{A}, \mathbf{r}\}$, $\sum_a \exp_q\left(\frac{\mathbf{r}}{q} - \xi\right)$ converge to 0 and ∞ if ξ goes to $+\infty$ and $-\infty$, respectively. Therefore by the intermediate value theorem, there exists a unique constant $\xi^* \in \mathbb{R}$ such that $\sum_a \exp_q\left(\frac{\mathbf{r}(a)}{q} - \xi^*\right) = 1$. Hence it is sufficient to take $\psi_q(\mathbf{r}/q) = \xi^*$.

To show the remains, we mainly employ the convex optimization technique. Since $S_q(\pi)$ is concave and the expectation and constraints of Δ are linear w. r. t. π, the problem is concave. Thus, strong duality holds and we can use KKT conditions to obtain an optimal solution.

Δ has two constraints: sum-to-one and nonnegativity. Let μ be a dual variable for $1 - \sum_a \pi(a) = 0$ and $\lambda(a)$ be a dual variable for $\pi(a) \geq 0$. Then, KKT conditions are as follows:

$$\begin{aligned}
&\forall i \;\; 1 - \sum_a \pi_q^\star(a) = 0,\; \pi_q^\star(a) \geq 0 \\
&\forall i \;\; \lambda^\star(a) \leq 0 \\
&\forall i \;\; \lambda^\star(a) p_i^\star = 0 \\
&\forall i \;\; \mathbf{r}(a) - \mu^\star - \ln_q(\pi_q^\star(a)) - (\pi_q^\star(a))^{q-1} + \lambda^\star(a) = 0
\end{aligned} \tag{C.7}$$

where * indicates an optimal solution. We focus on the last condition. The last condition is converted into

$$\begin{aligned}
0 &= \mathbf{r}(a) - \mu^\star - \ln_q(\pi_q^\star(a)) - (\pi_q^\star(a))^{q-1} + \lambda^\star(a) \\
0 &= \mathbf{r}(a) - \mu^\star - \ln_q(\pi_q^\star(a)) - (q-1)\frac{\pi_q^\star(a)^{q-1} - 1}{q-1} - 1 + \lambda^\star(a) \\
0 &= \mathbf{r}(a) - \mu^\star - q\ln_q(\pi_q^\star(a)) - 1 + \lambda^\star(a)
\end{aligned} \tag{C.8}$$

First, let's consider positive measure $\pi_q^\star(a) > 0$ ($\lambda^\star(a) = 0$). Then, from equation (C.8),

$$\exp_q\left(\frac{\mathbf{r}(a)}{q} - \frac{\mu^\star + 1}{q}\right) = \pi_q^\star(a) \tag{C.9}$$

and $\mu^\star$ can be found by solving the following equation:

$$\sum_a \exp_q\left(\frac{\mathbf{r}(a)}{q} - \frac{\mu^\star + 1}{q}\right) = 1. \tag{C.10}$$

Since the equation (C.10) is exactly same as a normalization equation (C.5), $\mu^\star$ can be found using a q-potential function ψ_q:

$$\mu^\star = q\psi_q\left(\frac{\mathbf{r}}{q}\right) - 1 \tag{C.11}$$

Then,

$$\pi_q^\star(a) = \exp_q\left(\frac{\mathbf{r}(a)}{q} - \psi_q\left(\frac{\mathbf{r}}{q}\right)\right). \tag{C.12}$$

The optimal value of primal problem is

$$\begin{aligned}
\mathop{\mathbb{E}}_{a\sim\pi_q^\star}[R] + S_q(\pi_q^\star) &= \sum_a \mathbf{r}(a)\pi_q^\star(a) - \sum_a \left[\frac{\mathbf{r}(a)}{q} - \psi_q\left(\frac{\mathbf{r}}{q}\right)\right]\pi_q^\star(a) \\
&= (q-1)\sum_a \frac{\mathbf{r}(a)}{q}\exp_q\left(\frac{\mathbf{r}(a)}{q} - \psi_q\left(\frac{\mathbf{r}}{q}\right)\right) + \psi_q\left(\frac{\mathbf{r}}{q}\right).
\end{aligned} \tag{C.13}$$

Finally, let's check the supporting set. For $\pi_q^\star(a) > 0$, the following condition should be satisfied:

$$1 + (q-1)\left(\frac{\mathbf{r}(a)}{q} - \psi_q\left(\frac{\mathbf{r}}{q}\right)\right) > 0, \tag{C.14}$$

where this condition comes from the definition of $\exp_q(x)$. □

C.2 Tsallis Bellman Optimality Equation

Markov Decision Processes with Tsallis entropy maximization is formulated as follows.

$$\underset{\pi\in\Pi}{\text{maximize}} \quad \underset{\tau\sim P,\pi}{\mathbb{E}}\left[\sum_{t}^{\infty}\gamma^{t}\mathbf{R}_{t}\right]+\alpha S_{q}^{\infty}(\pi) \tag{C.15}$$

In this section, we analyze the optimality condition of a Tsallis MDP.

Before starting proof, we first remind two propositions and prove one lemma. They are mainly employed to convert the optimization variable from π to the state action visitation ρ.

Proposition 5. *Let a state visitation be* $\rho_{\pi}(s)=\mathbb{E}_{\tau\sim P,\pi}\left[\sum_{t=0}^{\infty}\gamma^{t}\mathbb{I}(s_{t}=s)\right]$ *and state action visitation be* $\rho_{\pi}(s,a)=\mathbb{E}_{\tau\sim P,\pi}\left[\sum_{t=0}^{\infty}\gamma^{t}\mathbb{I}(s_{t}=s,a_{t}=a)\right]$ *. Following relationships hold.*

$$\rho_{\pi}(s)=\sum_{a}\rho_{\pi}(s,a),\quad \rho_{\pi}(s,a)=\rho_{\pi}(s)\pi(a|s) \tag{C.16}$$

$$\sum_{a}\rho_{\pi}(s,a)=d(s)+\sum_{s',a'}P(s|s',a')\rho_{\pi}(s',a'),\quad \rho_{\pi}(s,a) \tag{C.17}$$

where Equation (C.17) is called Bellman Flow constraints.

Proof. Proof can be found in [98, 126] □

Proposition 5 tells us, for fixed policy π, ρ_{π} satisfies Bellman Flow constraints. Then, the next remark show the opposite direction where if some function ρ satisfies Bellman Flow constraints, then there exist an unique policy which induces ρ.

Proposition 6 (Theorem 2 of [126])**.** *Let* $\mathbf{M}$ *be a set of state-action visitation measures, i.e.,*

$$\mathbf{M}\triangleq\{\rho|\forall s,\ a,\ \rho(s,a)\geq 0,\ \sum_{a}\rho(s,a)=d(s)+\sum_{s',a'}P(s|s',a')\rho(s',a')\}.$$

If $\rho \in \mathbf{M}$, then it is a state-action visitation measure for $\pi_\rho(a|s) \triangleq \frac{\rho(s,a)}{\sum_{a'} \rho(s,a')}$, and π_ρ is the unique policy whose state-action visitation measure is ρ.

Proof. Proof can be found in [98, 126]. □

Now, proposition 5 and 6 tell us that a policy and state action visitation have the one-to-one correspondence. In the following lemmas, we convert the optimization variable from π to ρ based on one-to-one correspondence.

Lemma 12. *Let*

$$\bar{S}_q^\infty(\rho) = -\sum_{s,a} \rho(s,a) \ln_q \left(\frac{\rho(s,a)}{\sum_{a'} \rho(s,a')} \right).$$

Then, for any stationary policy $\pi \in \Pi$ and any state-action visitation measure $\rho \in \mathbf{M}$, $S_q^\infty(\pi) = \bar{S}_q^\infty(\rho_\pi)$ and $\bar{S}_q^\infty(\rho) = S_q^\infty(\pi_\rho)$ hold.

Proof. First, show that $S_q^\infty(\pi) = \bar{S}_q^\infty(\rho_\pi)$.

$$\begin{aligned} S_q^\infty(\pi) &= \mathop{\mathbb{E}}_{\tau \sim P,\pi} \left[\sum_{t=0}^{\infty} \gamma^t S_q(\pi(\cdot|s_t)) \right] \\ &= \sum_s S_q(\pi(\cdot|s)) \cdot \mathbb{E}_{\tau \sim P,\pi} \left[\sum_{t=0}^{\infty} \gamma^t \mathbb{I}(s_t = s) \right] \\ &= \sum_s S_q(\pi(\cdot|s)) \rho_\pi(s) = \sum_{s,a} -\ln_q(\pi(a|s)) \pi(a|s) \rho_\pi(s) \\ &= \sum_{s,a} -\ln_q \left(\frac{\rho_\pi(s,a)}{\sum_{a'} \rho_\pi(s,a')} \right) \rho_\pi(s,a) = \bar{S}_q^\infty(\rho_\pi) \end{aligned} \tag{C.18}$$

Next, show that $\bar{S}_q^\infty(\rho) = S_q^\infty(\pi_\rho)$.

$$\begin{aligned} \bar{S}_q^\infty(\rho) &= \sum_{s,a} -\ln_q \left(\frac{\rho(s,a)}{\sum_{a'} \rho(s,a')} \right) \rho(s,a) \\ &= \sum_{s,a} -\ln_q(\pi_\rho(a|s)) \pi_\rho(a|s) \rho(s) = S_q^\infty(\pi_\rho) \end{aligned} \tag{C.19}$$

□

Corollary 11. *The problem (C.20) is equivalent to a Tsallis MDP, which means if $\rho^{\star}$ is an optimal solution of (C.20), then, $\pi_{\rho^{\star}}$ is an optimal solution of a Tsallis MDP and vice versa.*

Proof. Let $\rho^{\star}$ be an optimal solution of (C.20). Assume that there exist another policyt π' such that $J(\pi') + S_q^{\infty}(\pi') > J(\pi_{\rho^{\star}}) + S_q^{\infty}(\pi_{\rho^{\star}})$ where $J(\pi) = \mathbb{E}_{\tau\sim\pi,P}\left[\sum_{t=0}^{\infty}\gamma^t\mathbf{R}_t\right]$. Then, $\sum_{s,a}\rho_{\pi'}(s,a)\mathbf{r}(s,a)+\bar{S}_q^{\infty}(\rho_{\pi'}) > \sum_{s,a}\rho_{\star}(s,a)\mathbf{r}(s,a)+\bar{S}_q^{\infty}(\rho_{\star})$. It contradicts to the fact that $\rho^{\star}$ is the optimal solution of (C.20). Thus, for all π, $J(\pi) + S_q^{\infty}(\pi) \le J(\pi_{\rho^{\star}}) + S_q^{\infty}(\pi_{\rho^{\star}})$ which means $\pi_{\rho^{\star}}$ is the optimal policy. The opposite direction also can be proven in the same way. □

Lemma 12 shows that $\bar{S}_q^{\infty}(\rho)$ and $S_q^{\infty}(\pi)$ has the same function value. Thus, we can freely change the optimization variable from π to ρ since the optimal point does not change due to the Corollary 11.

C.3 Variable Change

Based on Proposition 6 and Lemma 12, we convert a Tsallis MDP problem to

$$\begin{aligned}\underset{\rho}{\text{maximize}} \quad & \sum_{s,a}\rho(s,a)\sum_{s'}\mathbf{r}(s,a,s')P(s'|s,a) - \sum_{s,a}\rho(s,a)\ln_q\left(\frac{\rho(s,a)}{\sum_{a'}\rho(s,a')}\right) \\ \text{subject to} \quad & \forall s,\ a,\ \rho(s,a)\ge 0,\ \sum_{a}\rho(s,a) = d(s) + \sum_{s',a'}P(s|s',a')\rho(s',a').\end{aligned} \tag{C.20}$$

Now, the optimization variables in the problem (C.20) is a state action visitation. In the following lemmas, we show that the problem (C.20) is concave with respect to a state action visitation.

Lemma 13. *$\bar{S}_q^{\infty}(\rho)$ is concave function with respect to $\rho \in \mathbf{M}$*

Proof. Let us consider the function $f(x) = -x\ln_q(x)$ defined over $(x > 0)$. Second derivative of $d^2 f(x)/dx^2$ is computed as

$$\frac{d^2 f(x)}{dx^2} = -qx^{q-2} < 0 \;\; (x > 0).$$

Since its second derivative is always negative on its domain, $f(x)$ is a concave function. From this fact, we can show that $\bar{S}_q^{\infty}(\rho)$ is concave. Proving the concavity is equivalent to show that for any $0 < \lambda_1, \lambda_2 < 1$ such that $\lambda_1 + \lambda_2 = 1$, and for $\rho_1, \rho_2 \in \mathbf{M}$ the following inequality holds

$$\bar{S}_q^{\infty}(\lambda_1\rho_1 + \lambda_2\rho_2) > \lambda_1 \bar{S}_q^{\infty}(\rho_1) + \lambda_2 \bar{S}_q^{\infty}(\rho_2)$$

For notional simplicity, let $\tilde{\rho}$ be $\lambda_1\rho_1 + \lambda_2\rho_2$ and define $\mu_1 = \frac{\lambda_1 \sum_{a'} \rho_1(s,a')}{\sum_{a'} \tilde{\rho}(s,a')}$ and $\mu_2 = \frac{\lambda_2 \sum_{a'} \rho_2(s,a')}{\sum_{a'} \tilde{\rho}(s,a')}$. Note that from the definition, $\mu_1 + \mu_2 = 1$. It can be shown as follow:

$$\begin{aligned}
\bar{S}_q^{\infty}(\lambda_1\rho_1 + \lambda_2\rho_2) &= -\sum_{s,a} \tilde{\rho}(s,a) \ln_q \left(\frac{\lambda_1\rho_1(s,a) + \lambda_2\rho_2(s,a)}{\sum_{a'} \tilde{\rho}(s,a')} \right) \\
&= -\sum_{s,a} \tilde{\rho}(s,a) \ln_q \left(\frac{\mu_1\rho_1(s,a)}{\sum_{a'} \rho_1(s,a')} + \frac{\mu_2\rho_2(s,a)}{\sum_{a'} \rho_2(s,a')} \right) \\
&= -\sum_{s,a} \left(\sum_{a'} \tilde{\rho}(s,a') \frac{\lambda_1\rho_1(s,a) + \lambda_2\rho_2(s,a)}{\sum_{a'} \tilde{\rho}(s,a')} \right) \ln_q \left(\frac{\mu_1\rho_1(s,a)}{\sum_{a'} \rho_1(s,a')} + \frac{\mu_2\rho_2(s,a)}{\sum_{a'} \rho_2(s,a')} \right) \\
&= -\sum_{s,a}\sum_{a'} \tilde{\rho}(s,a') \left(\frac{\mu_1\rho_1(s,a)}{\sum_{a'} \rho_1(s,a')} + \frac{\mu_2\rho_2(s,a)}{\sum_{a'} \tilde{\rho}(s,a')} \right) \ln_q \left(\frac{\mu_1\rho_1(s,a)}{\sum_{a'} \rho_1(s,a')} + \frac{\mu_2\rho_2(s,a)}{\sum_{a'} \rho_2(s,a')} \right)
\end{aligned} \tag{C.21}$$

Then, for all s, a,

$$\begin{aligned}
&-\left(\frac{\mu_1\rho_1(s,a)}{\sum_{a'} \rho_1(s,a')} + \frac{\mu_2\rho_2(s,a)}{\sum_{a'} \tilde{\rho}(s,a')} \right) \ln_q \left(\frac{\mu_1\rho_1(s,a)}{\sum_{a'} \rho_1(s,a')} + \frac{\mu_2\rho_2(s,a)}{\sum_{a'} \rho_2(s,a')} \right) \\
&> -\mu_1 \frac{\rho_1(s,a)}{\sum_{a'} \rho_1(s,a')} \ln_q \left(\frac{\rho_1(s,a)}{\sum_{a'} \rho_1(s,a')} \right) - \mu_2 \frac{\rho_2(s,a)}{\sum_{a'} \rho_2(s,a')} \ln_q \left(\frac{\rho_2(s,a)}{\sum_{a'} \rho_2(s,a')} \right)
\end{aligned}$$

Equation (C.21) becomes

$$\begin{aligned}
&\bar{S}_q^{\infty}(\lambda_1\rho_1 + \lambda_2\rho_2) \\
&= -\sum_{s,a}\sum_{a'} \tilde{\rho}(s,a') \left(\frac{\mu_1\rho_1(s,a)}{\sum_{a'} \rho_1(s,a')} + \frac{\mu_2\rho_2(s,a)}{\sum_{a'} \tilde{\rho}(s,a')} \right) \ln_q \left(\frac{\mu_1\rho_1(s,a)}{\sum_{a'} \rho_1(s,a')} + \frac{\mu_2\rho_2(s,a)}{\sum_{a'} \rho_2(s,a')} \right) \\
&> -\sum_{s,a}\sum_{a'} \tilde{\rho}(s,a')\mu_1 \frac{\rho_1(s,a)}{\sum_{a'} \rho_1(s,a')} \ln_q \left(\frac{\rho_1(s,a)}{\sum_{a'} \rho_1(s,a')} \right) \\
&\quad -\sum_{s,a}\sum_{a'} \tilde{\rho}(s,a')\mu_2 \frac{\rho_2(s,a)}{\sum_{a'} \rho_2(s,a')} \ln_q \left(\frac{\rho_2(s,a)}{\sum_{a'} \rho_2(s,a')} \right)
\end{aligned}$$

Since $\sum_{a'} \tilde{\rho}(s,a')\mu_1 \frac{\rho_1(s,a)}{\sum_{a'}\rho_1(s,a')} = \sum_{a'} \tilde{\rho}(s,a') \frac{\lambda_1 \sum_{a'}\rho_1(s,a')}{\sum_{a'}\tilde{\rho}(s,a')} \frac{\rho_1(s,a)}{\sum_{a'}\rho_1(s,a')} = \lambda_1\rho_1(s,a)$, finally, we get

$$
\begin{aligned}
&\bar{S}_q^{\infty}(\lambda_1\rho_1 + \lambda_2\rho_2)\\
&\quad > -\sum_{s,a}\sum_{a'} \tilde{\rho}(s,a')\mu_1 \frac{\rho_1(s,a)}{\sum_{a'}\rho_1(s,a')} \ln_q\left(\frac{\rho_1(s,a)}{\sum_{a'}\rho_1(s,a')}\right)\\
&\qquad -\sum_{s,a}\sum_{a'} \tilde{\rho}(s,a')\mu_2 \frac{\rho_2(s,a)}{\sum_{a'}\rho_2(s,a')} \ln_q\left(\frac{\rho_2(s,a)}{\sum_{a'}\rho_2(s,a')}\right)\\
&\quad = -\sum_{s,a}\lambda_1\rho_1(s,a)\ln_q\left(\frac{\rho_1(s,a)}{\sum_{a'}\rho_1(s,a')}\right) - \sum_{s,a}\lambda_2\rho_2(s,a)\ln_q\left(\frac{\rho_2(s,a)}{\sum_{a'}\rho_2(s,a')}\right)\\
&\quad =\lambda_1\bar{S}_q^{\infty}(\rho_1) + \lambda_2\bar{S}_q^{\infty}(\rho_2)
\end{aligned}
$$

Note that this proof holds for every q value greater than zero. □

Corollary 12. *The problem (C.20) is concave with respect to* $\rho \in \mathbf{M}$

Proof. The objective function of (C.20) is concave function w.r.t ρ since the first term is linear and the second term is concave be Lemma 13. All constraints are linear w.r.t ρ. Thus, the problem is a concave problem. □

C.4 Tsallis Bellman Optimality Equation

Proof of Theorem 14. Since the problem (C.20) is concave with respect to ρ, the primal and dual solutions necessarily and sufficiently satisfy a KKT condition. First, the Lagrangian objecitve $\mathcal{L} \triangleq \sum_{s,a}\rho(s,a)\mathbf{r}(s,a) - \sum_{s,a}\rho(s,a)\ln_q\left(\frac{\rho(s,a)}{\sum_{a'}\rho(s,a')}\right) + \sum_{s,a}\lambda(s,a)\rho(s,a) + \sum_s \mu(s)\left(d(s) + \sum_{s',a'}P(s|s',a')\rho(s',a') - \sum_a \rho(s,a)\right)$ where $\lambda(s,a)$ and $\mu(s)$ are dual variables for nonnegativity and Bellman flow constraints.

The KKT conditions of the problem (C.20) are as follows:

$$\begin{aligned}
&\forall s,\ a,\ \rho^\star(s,a) \geq 0,\ d(s) + \sum_{s',a'} P(s|s',a')\rho^\star(s',a') - \sum_a \rho^\star(s,a) = 0 \\
&\forall s,\ a,\ \lambda^\star(s,a) \leq 0 \\
&\forall s,\ a,\ \lambda^\star(s,a)\rho^\star(s,a) = 0 \qquad\qquad (\text{C.22}) \\
&\forall s,\ a,\ 0 = \sum_{s'} \mathbf{r}(s,a,s')P(s'|s,a) + \gamma\sum_{s'}\mu^\star(s')P(s'|s,a) \\
&\qquad - \mu^\star(s) - q\ln_q\left(\frac{\rho^\star(s,a)}{\sum_{a'}\rho^\star(s,a')}\right) - 1 + \sum_a \left(\frac{\rho^\star(s,a)}{\sum_{a'}\rho^\star(s,a')}\right)^q + \lambda^\star(s,a)
\end{aligned}$$

We would like to note that the dervative of $\bar{S}_q^\infty(\rho)$ is computed as follows:

$$\begin{aligned}
\frac{\partial \bar{S}_q^\infty(\rho)}{\partial\rho(s'',a'')} &= -\sum_{s,a}\frac{\partial\rho(s,a)}{\partial\rho(s'',a'')}\ln_q\left(\frac{\rho(s,a)}{\sum_{a'}\rho(s,a')}\right) - \sum_{s,a}\rho(s,a)\frac{\partial\ln_q\left(\rho(s,a)/\sum_{a'}\rho(s,a')\right)}{\partial\rho(s'',a'')} \\
&= -\ln_q\left(\frac{\rho(s'',a'')}{\sum_{a'}\rho(s'',a')}\right) \\
&\quad -\sum_a \rho(s'',a)\left(\frac{\rho(s'',a)}{\sum_{a'}\rho(s'',a')}\right)^{q-2}\left(\frac{\delta_{a''}(a)}{\sum_{a'}\rho(s'',a')} - \frac{\rho(s'',a)}{\left(\sum_{a'}\rho(s'',a')\right)^2}\right) \qquad (\text{C.23}) \\
&= -\ln_q\left(\frac{\rho(s'',a'')}{\sum_{a'}\rho(s'',a')}\right) - \left(\frac{\rho(s'',a'')}{\sum_{a'}\rho(s'',a')}\right)^{q-1} + \sum_a\left(\frac{\rho(s'',a)}{\sum_{a'}\rho(s'',a')}\right)^q \\
&= -q\ln_q\left(\frac{\rho(s'',a'')}{\sum_{a'}\rho(s'',a')}\right) - 1 + \sum_a\left(\frac{\rho(s'',a)}{\sum_{a'}\rho(s'',a')}\right)^q
\end{aligned}$$

Then, we show that $\mu^\star(s)$ is the same as optimal value $V_q^\star(s)$. From the stationary condition, by multiplying $\pi_{\rho^\star}(a|s) = \rho^\star(s,a)/\sum_{a'}\rho^\star(s,a')$ and summing up with respect to a, the following equation is obtained:

$$
\begin{aligned}
0 =& \sum_{a} \sum_{s'} \mathbf{r}(s,a,s') P(s'|s,a) \pi_{\rho^\star}(a|s) + \gamma \sum_{s'} \mu^\star(s') \sum_{a} P(s'|s,a) \pi_{\rho^\star}(a|s) - \mu^\star(s) \\
& - q \sum_{a} \pi_{\rho^\star}(a|s) \ln_q \left(\frac{\rho^\star(s,a)}{\sum_{a'} \rho^\star(s,a')} \right) - 1 + \sum_{a} \pi_{\rho^\star}(a|s) \sum_{a''} \left(\frac{\rho^\star(s,a'')}{\sum_{a'} \rho^\star(s,a')} \right)^q \\
& + \sum_{a} \lambda^\star(s,a) \pi_{\rho^\star}(a|s) \\
=& \sum_{a} \sum_{s'} \mathbf{r}(s,a,s') P(s'|s,a) \pi_{\rho^\star}(a|s) + \gamma \sum_{s'} \mu^\star(s') \sum_{a} P(s'|s,a) \pi_{\rho^\star}(a|s) \\
& - \mu^\star(s) - q \sum_{a} \pi_{\rho^\star}(a|s) \ln_q \left(\pi_{\rho^\star}(a|s) \right) - 1 + \sum_{a''} \pi_{\rho^\star}(s,a)^q \\
& + \sum_{a} \lambda^\star(s,a) \pi_{\rho^\star}(a|s) \\
=& \sum_{a} \sum_{s'} \mathbf{r}(s,a,s') P(s'|s,a) \pi_{\rho^\star}(a|s) + \gamma \sum_{s'} \mu^\star(s') \sum_{a} P(s'|s,a) \pi_{\rho^\star}(a|s) \\
& - \mu^\star(s) - q \sum_{a} \pi_{\rho^\star}(a|s) \ln_q \left(\pi_{\rho^\star}(a|s) \right) - 1 + \sum_{a''} \pi_{\rho^\star}(s,a)^q \\
=& \sum_{a} \sum_{s'} \mathbf{r}(s,a,s') P(s'|s,a) \pi_{\rho^\star}(a|s) + \gamma \sum_{s'} \mu^\star(s') \sum_{a} P(s'|s,a) \pi_{\rho^\star}(a|s) \\
& - \mu^\star(s) - q \sum_{a} \pi_{\rho^\star}(a|s) \ln_q \left(\pi_{\rho^\star}(a|s) \right) + (q-1) \sum_{s,a} \pi_{\rho^\star}(s,a) \ln_q \left(\pi_{\rho^\star}(s,a) \right) \\
=& \sum_{a} \sum_{s'} \mathbf{r}(s,a,s') P(s'|s,a) \pi_{\rho^\star}(a|s) + \gamma \sum_{s'} \mu^\star(s') \sum_{a} P(s'|s,a) \pi_{\rho^\star}(a|s) \\
& - \mu^\star(s) - \sum_{s,a} \pi_{\rho^\star}(s,a) \ln_q \left(\pi_{\rho^\star}(s,a) \right).
\end{aligned}
\tag{C.24}
$$

Finally,

$$
\begin{aligned}
\mu^\star(s) =& \sum_{a} \sum_{s'} \mathbf{r}(s,a,s') P(s'|s,a) \pi_{\rho^\star}(a|s) + \gamma \sum_{s'} \mu^\star(s') \sum_{a} P(s'|s,a) \pi_{\rho^\star}(a|s) \\
& - \sum_{a} \pi_{\rho^\star}(a|s) \ln_q \left(\pi_{\rho^\star}(a|s) \right) \\
=& \mathbb{E}_{s' \sim P, a \sim \pi} \left[\mathbf{r}(s,a,s') + \alpha S_q(\pi_{\rho^\star}(\cdot|s)) + \gamma \mu^\star(s') \middle| s \right]
\end{aligned}
\tag{C.25}
$$

This equation (C.25) exactly satisfies Tsallis Bellman expectation (TBE) equation of $\pi_{\rho^\star}$. Thus, we want to claim that $\mu^\star(s)$ is the value $V^{\pi_{\rho^\star}}(s)$ of optimal policy $\pi_{\rho^\star}$, i.e., $\mu^\star(s) = V_q^\star(s)$. However, to guarantee $\mu^\star(s) = V_q^\star(s)$, we should prove the following statement: *if an arbitrary function $f(s)$ satisfies a TBE equation for π, then, $f(s) = V^\pi(s)$.*

Then, we first analyze a positive state-action visitation $\rho^\star(s,a) > 0$ ($\lambda^\star(s,a) = 0$). Using the fact that $\mu^\star = V_q^\star$, we can obtain $Q_q^\star(s,a) = \mathbb{E}_{s'\sim P}[\mathbf{r}(s,a,s') + \gamma\mu^\star(s')]$. By replacing $\rho^\star(s,a)/\sum_{a'}\rho^\star(s,a')$ with $\pi_{\rho^\star}(a|s)$ and using $Q_q^\star(s,a) = \mathbb{E}_{s'\sim P}[\mathbf{r}(s,a,s') + \gamma\mu^\star(s')]$ and $\mu^\star(s) = V^\star(s)$,

$$\begin{aligned}
&Q_q^\star(s,a) - V_q^\star(s) - q\ln_q\left(\pi_{\rho^\star}(a|s)\right) - 1 + \sum_a \pi_{\rho^\star}(a|s)^q = 0\\
&\frac{Q_q^\star(s,a)}{q} - \frac{V_q^\star(s) + 1 - \sum_a \left(\pi_{\rho^\star}(a|s)\right)^q}{q} = \ln_q\left(\pi_{\rho^\star}(a|s)\right)\\
&\exp_q\left(\frac{Q_q^\star(s,a)}{q} - \frac{V_q^\star(s) + 1 - \sum_a \left(\pi_{\rho^\star}(a|s)\right)^q}{q}\right) = \pi_{\rho^\star}(a|s).
\end{aligned} \tag{C.26}$$

Now, we can use $\sum_a \pi(a|s) = 1$. By summing up with respect to a,

$$\sum_a \exp_q\left(\frac{Q_q^\star(s,a)}{q} - \frac{V_q^\star(s) + 1 - \sum_a \left(\pi_{\rho^\star}(a|s)\right)^q}{q}\right) = 1. \tag{C.27}$$

This equation is the normalization equation of q-exponential distribution (C.5). So, we can obtain the relationship between q-potential and the optimal value function.

$$\psi_q\left(\frac{Q_q^\star(s,\cdot)}{q}\right) = \frac{V_q^\star(s) + 1 - \sum_a \left(\pi_{\rho^\star}(a|s)\right)^q}{q} \tag{C.28}$$

Finally, it is shown that the optimal policy has q-exponential distribution of $Q_q^\star(s,\cdot)$.

$$\exp_q\left(\frac{Q_q^\star(s,a)}{q} - \psi_q\left(\frac{Q_q^\star(s,\cdot)}{q}\right)\right) = \pi_{\rho^\star}(a|s) \tag{C.29}$$

By plugging in this result into (C.25),

$$
\begin{aligned}
V_q^\star(s) &= \sum_a \pi_{\rho^\star}(a|s) \sum_{s'} \left[\mathbf{r}(s,a,s') + \gamma V_q^\star(s') P(s'|s,a)\right] \\
&\quad - \sum_a \pi_{\rho^\star}(a|s) \ln_q\left(\pi_q^\star(a|s)\right) \\
&= \sum_a \pi_{\rho^\star}(a|s) Q_q^\star(s,a) - \sum_a \pi_{\rho^\star}(a|s) \ln_q\left(\pi_q^\star(a|s)\right) \\
&= \sum_a \pi_{\rho^\star}(a|s) Q_q^\star(s,a) - \sum_a \pi_{\rho^\star}(a|s) \left(\frac{Q_q^\star(s,a)}{q} - \psi_q\left(\frac{Q_q^\star(s,\cdot)}{q}\right)\right) \\
&= (q-1) \sum_a \pi_{\rho^\star}(a|s) \frac{Q_q^\star(s,a)}{q} + \psi_q\left(\frac{Q_q^\star(s,\cdot)}{q}\right) \\
&= q\text{-}\max_{a'}\left(Q_q^\star(s,a')\right)
\end{aligned}
\tag{C.30}
$$

where the last equation is derived using the Equation (C.6).

To summarize, we obtain the optimality condition for a Tsallis MDP as follows:

$$
\begin{aligned}
Q_q^\star(s,a) &= \mathbb{E}_{s'}\left[\mathbf{r}(s,a,s') + \gamma V^\star(s') \middle| s,a\right] \\
V_q^\star(s) &= q\text{-}\max_{a'}(Q_q^\star(s,a')) \\
\pi_q^\star(a|s) &= \exp_q\left(\frac{Q_q^\star(s,a)}{q} - \psi_q\left(\frac{Q_q^\star(s,\cdot)}{q}\right)\right)
\end{aligned}
\tag{C.31}
$$

We call these equations Tsallis Bellman optimality (TBO) equations. □

C.5 Tsallis Policy Iteration

C.6 Tsallis Bellman Expectation (TBE) Equation

In Tsallis policy evaluation, for fixed π, the value functions of π have the relationship as follows:

$$
\begin{aligned}
Q_q^\pi(s,a) &= \mathop{\mathbb{E}}_{s'\sim P}[\mathbf{r}(s,a,s') + \gamma V_q^\pi(s')|s,a] \\
V_q^\pi(s) &= \mathop{\mathbb{E}}_{a\sim\pi}[Q_q^\pi(s,a) - \ln_q(\pi(a|s))],
\end{aligned}
\tag{C.32}
$$

These equations are derived from the definition of V_q^π and Q_q^π. Thus, if we have some value functions of Tsallis MDP, then, they satisfies TBE equation trivially. However, main goal of Tsallis policy evaluation is to prove the opposite direction: *if an arbitrary function $f(s)$ satisfies a TBE equation for π, then, $f(s) = V^\pi(s)$.*

C.7 Tsallis Bellman Expectation Operator and Tsallis Policy Evaluation

$$\begin{aligned} \left[\mathcal{T}_q^\pi F\right](s,a) &\triangleq \mathop{\mathbb{E}}_{s'\sim P}[\mathbf{r}(s,a,s') + \gamma V_F(s')|s,a] \\ V_F(s) &\triangleq \mathop{\mathbb{E}}_{a\sim\pi}[F(s,a) - \ln_q(\pi(a|s))], \end{aligned} \tag{C.33}$$

where $s' \sim P$ indicates $s' \sim P(\cdot|s,a)$ and $a' \sim \pi$ indicates $a' \sim \pi(\cdot|s)$. Then, policy evaluation method in a Tsallis MDP can be simply defined as

$$F_{k+1} = \mathcal{T}_q^\pi F_k.$$

Before proving the Tsallis policy evaluation step, we first drive the properties of $\mathcal{T}_q^\pi$.

Lemma 14. *For $F : \mathcal{S}\times\mathcal{A} \to R$ and $c \in R^+$, $\mathcal{T}_q^\pi(F + c\mathbf{1}) = \mathcal{T}_q^\pi F + \gamma c\mathbf{1}$ where $\mathbf{1} : \mathcal{S}\times\mathcal{A} \to 1$*

Proof. For all s, a,

$$\begin{aligned} V_{F+c\mathbf{1}}(s) &= \mathop{\mathbb{E}}_{a\sim\pi}[F(s,a) + c - \ln_q(\pi(a|s))] = \mathop{\mathbb{E}}_{a\sim\pi}[F(s,a) - \ln_q(\pi(a|s))] + c \\ &= V_F(s) + c \end{aligned} \tag{C.34}$$

Thus,

$$\begin{aligned} \left[\mathcal{T}_q^\pi(F + c\mathbf{1})\right](s,a) &= \mathop{\mathbb{E}}_{s'\sim P}[\mathbf{r}(s,a,s') + \gamma V_{F+c\mathbf{1}}(s')|s,a] \\ &= \mathop{\mathbb{E}}_{s'\sim P}[\mathbf{r}(s,a,s') + \gamma V_F(s') + \gamma c|s,a] \\ &= \mathop{\mathbb{E}}_{s'\sim P}[\mathbf{r}(s,a,s') + \gamma V_F(s')|s,a] + \gamma c = \mathcal{T}_q^\pi F(s) + \gamma c \end{aligned} \tag{C.35}$$

□

Lemma 15. *For $F, G : \mathcal{S} \times \mathcal{A} \to R$ and $F \succeq G$, $\mathcal{T}_q^{\pi}(F) \succeq \mathcal{T}_q^{\pi}(G)$ where $\succeq$ indicates $F(s,a) \geq G(s,a)$ for all s, a.*

Proof. For all s, a,

$$V_F(s) = \mathop{\mathbb{E}}_{a\sim\pi}[F(s,a) - \ln_q(\pi(a|s))] < \mathop{\mathbb{E}}_{a\sim\pi}[G(s,a) - \ln_q(\pi(a|s))] = V_G(s) \tag{C.36}$$

Thus,

$$\begin{aligned} \left[\mathcal{T}_q^{\pi} F\right](s,a) &= \mathop{\mathbb{E}}_{s'\sim P}[\mathbf{r}(s,a,s') + \gamma V_F(s')|s,a] \\ &< \mathop{\mathbb{E}}_{s'\sim P}[\mathbf{r}(s,a,s') + \gamma V_G(s')|s,a] = \left[\mathcal{T}_q^{\pi} G\right](s,a) \end{aligned} \tag{C.37}$$

□

Lemma 16. *$\mathcal{T}_q^{\pi}$ is γ-contraction mapping in $(C(\mathcal{S} \times \mathcal{A}, R), |\cdot|_{\infty})$ where $C(\mathcal{S} \times \mathcal{A}, R) \triangleq \{F : \mathcal{S} \times \mathcal{A} \to R\}$ and $|F - G|_{\infty} = \sup_{s,a} |F(s,a) - G(s,a)|$*

Proof. Let $d = |F - G|_{\infty}$. The, $G - d\mathbf{1} \succeq F \succeq G + d\mathbf{1}$. From Lemma 15, $\mathcal{T}_q^{\pi}(G + d\mathbf{1}) \succeq \mathcal{T}_q^{\pi} F \succeq \mathcal{T}_q^{\pi}(G - d\mathbf{1})$. From Lemma 14, $\mathcal{T}_q^{\pi} G + \gamma d\mathbf{1} \succeq \mathcal{T}_q^{\pi} F \succeq \mathcal{T}_q^{\pi} G - \gamma d\mathbf{1}$. Then,$\gamma d\mathbf{1} \succeq \mathcal{T}_q^{\pi} F - \mathcal{T}_q^{\pi} G \succeq -\gamma d\mathbf{1}$. Finally,

$$|\mathcal{T}_q^{\pi} F - \mathcal{T}_q^{\pi} G|_{\infty} \leq \gamma d = \gamma |F - G|_{\infty}.$$

Consequently, $\mathcal{T}_q^{\pi}$ is γ-contraction. □

Proof of Tsallis Policy Evaluation

Proof of Theorem 15. From Lemma 16, $\mathcal{T}_q^{\pi}$ is γ-contraction and has an unique fixed point $F_* = \mathcal{T}_q^{\pi} F_*$ from the Banach fixed point theorem. Then, for any initial function F, a sequence of F_k converges to the fixed point, i.e., $F_* =$

$\lim_{k\to\infty}(\mathcal{T}_q^{\pi})^k F_0$. The fixed point F_* satisfies a TBE equation as follows:

$$\begin{aligned} F_*(s,a) &= \mathop{\mathbb{E}}_{s'\sim P}[\mathbf{r}(s,a,s') + \gamma V_{F_*}(s')|s,a] \\ V_{F_*}(s) &= \mathop{\mathbb{E}}_{a\sim\pi}[F_*(s,a) - \ln_q(\pi(a|s))], \end{aligned} \tag{C.38}$$

Since F_* is unique, F_* is the only function which satisfies a TBE equation. Thus, $F_* = Q_q^{\pi}$. □

C.8 Tsallis Policy Improvement

The value function evaluated from Tsallis policy evaluation can be employed to update the policy distribution. In policy improvement step, the policy will be updated to maximize

$$\forall s,\ \pi_{k+1}(\cdot|s) \triangleq \arg\max_{\pi(\cdot|s)} \mathop{\mathbb{E}}_{a\sim\pi}[Q_q^{\pi_k}(s,a) - \ln_q(\pi(a|s))|s] \tag{C.39}$$

Proof of Theorem 16. Since π_{k+1} is updated by maximizing Equation (C.39) and the maximization in Equation (C.39) is concave with respect to π, the following inequality holds

$$\mathop{\mathbb{E}}_{a\sim\pi_{k+1}}\left[Q_q^{\pi_k}(s,a) - \ln_q(\pi_{k+1}(a|s))\middle|s\right] \geq \mathop{\mathbb{E}}_{a\sim\pi_k}\left[Q_q^{\pi_k}(s,a) - \ln_q(\pi_k(a|s))\middle|s\right] = V_q^{\pi_k}(s), \tag{C.40}$$

where the equality holds when $\pi_{k+1} = \pi_k$. This inequality induces a performance improvement,

$$\begin{aligned}
Q_q^{\pi_k}(s,a) &= \underset{s_1\sim P}{\mathbb{E}}\left[r(s_0,a_0,s_1)+\gamma V_q^{\pi_k}(s_1)\middle|s_0=s,a_0=a\right]\\
&\leq \underset{s_1\sim P}{\mathbb{E}}\left[r(s_0,a_0,s_1)|s_0=s,a_0=a\right]\\
&\quad+\gamma \underset{s_1,a_1\sim P,\pi_{k+1}}{\mathbb{E}}\left[Q_q^{\pi_k}(s_1,a_1)-\ln_q(\pi_{k+1}(a_1|s_1))\middle|s_0=s,a_0=a\right]\\
&= \underset{s_1\sim P}{\mathbb{E}}\left[r(s_0,a_0,s_1)|s_0=s,a_0=a\right]\\
&\quad+\gamma \underset{s_{1:2},a_1\sim P,\pi_{k+1}}{\mathbb{E}}\left[r(s_1,a_1,s_2)-\ln_q(\pi_{k+1}(a_1|s_1))+\gamma V_q^{\pi_k}(s_2)\middle|s_0=s,a_0=a\right]\\
&\leq \underset{s_1\sim P}{\mathbb{E}}\left[r(s_0,a_0,s_1)|s_0=s,a_0=a\right]\\
&\quad+\gamma \underset{s_{1:t+1},a_{1:t}\sim P,\pi_{k+1}}{\mathbb{E}}\left[\sum_{k=1}^{t}\gamma^{k-1}\left(r(s_k,a_k,s_{k+1})-\ln_q(\pi_{k+1}(a_k|s_k))\right)\middle|s_0=s,a_0=a\right]\\
&\quad+\gamma^{t+1} \underset{s_{t+1}\sim P,\pi_{k+1}}{\mathbb{E}}\left[V_q^{\pi_k}(s_{t+1})\middle|s_0=s,a_0=a\right]\\
&\vdots\\
&\leq \underset{s_1\sim P}{\mathbb{E}}\left[r(s_0,a_0,s_1)+\gamma V_q^{\pi_{k+1}}(s_1)\middle|s_0=s,a_0=a\right]=Q_q^{\pi_{k+1}}(s,a),
\end{aligned}\tag{C.41}$$

where $\gamma^{t+1}\,\mathbb{E}_{s_{t+1}\sim P,\pi_{k+1}}\left[V_q^{\pi_k}(s_{t+1})\middle|s_0=s,a_0=a\right]\to 0$ as $t\to\infty$. □

Proof of Theorem 17. From the fact that reward function $\mathbf{r}$ has upper bound $\mathbf{r}_{\max}$ and $\mathcal{S}\times\mathcal{A}$ is bounded, $Q_q^{\pi_k}$ is also bouned. Then, since a sequence of $Q_q^{\pi_k}$ is monotonically non-decreasing and bounded, it converges to some point π_*. Now, proof will be done by showing $\pi_*=\pi_q^\star$. First, from the policy improvement, We have $\pi_*(\cdot|s)=\arg\max_{\pi(\cdot|s)}\mathbb{E}_{a\sim\pi}[Q_q^{\pi_*}(s,a)-\ln_q(\pi(a|s))|s]$ and at π_*, the equality in Equation (C.40) holds, i.e., $V_q^{\pi_*}(s)=\mathbb{E}_{a\sim\pi_*}\left[Q_q^{\pi_*}(s,a)-\alpha\ln_q(\pi_*(a|s))\middle|s\right]$. Then, the following equality holds,

$$V_q^{\pi_*}(s)=\max_{\pi(\cdot|s)}\underset{a\sim\pi}{\mathbb{E}}\left[Q_q^{\pi_*}(s,a)-\alpha\ln_q(\pi(a|s))\middle|s\right],$$

which is equivalent to $V_q^{\pi_*}(s)=q\text{-}\max_{a'}Q^{\pi_*}(s,a')$. It can be also known that π_* is the solution of q-maximum. From the TBE equation, $Q_q^{\pi_*}(s,a)=\mathbb{E}_{s'\sim P}[\mathbf{r}(s,a,s')+\gamma V_q^{\pi_*}(s')|s,a]$. Thus, π_* satisfies a TBO equation and by Theorem 14, $\pi_*=\pi_q^\star$. □

C.9 Tsallis Value Iteration

Tsallis value iteration is derived from the optimality equation. From TBO equation, Tsallis Bellman optimality operator is defined by

$$\begin{aligned} \left[\mathcal{T}_q F\right](s,a) &\triangleq \mathop{\mathbb{E}}_{s'\sim P}\left[\mathbf{r}(s,a,s') + \gamma V_F(s) \middle| s,a\right] \\ V_F(s) &\triangleq q\text{-}\max_{a'}\left(F(s,a')\right). \end{aligned} \tag{C.42}$$

Then, a Tsallis value iteration is defined by repeatedly applying TBO operator:

$$F_{k+1} = \mathcal{T}_q F_k.$$

Before proving the Tsallis value iteration, we first drive the properties of q-maximum and $\mathcal{T}_q$.

Lemma 17. *For any function $f(x)$ defined on finite input space $\mathcal{X}$ and $c \in R$, The following equality hold:*

1. *$q\text{-}\max_x(f(x) + c\mathbf{1}) = q\text{-}\max_x(f(x)) + c$*

2. *$q\text{-}\max_x(f(x))$ is monotone. If $f \preceq g$, then $q\text{-}\max_x(x) \leq q\text{-}\max_x(y)$*

where $\mathbf{1}$ is a constant function whose value is one.

Proof. For property 1,

$$\begin{aligned} q\text{-}\max_x(f(x) + c\mathbf{1}) &= \max_{P\in\Delta}\left[\mathop{\mathbb{E}}_{X\sim P}\left[f(X) + c\mathbf{1}(X)\right] + S_q(P)\right] \\ &= \max_{P\in\Delta}\left[\mathop{\mathbb{E}}_{X\sim P}\left[f(X)\right] + c + S_q(P)\right] \\ &= \max_{P\in\Delta}\left[\mathop{\mathbb{E}}_{X\sim P}\left[f(X)\right] + S_q(P)\right] + c = q\text{-}\max_x(f(x)) + c \end{aligned} \tag{C.43}$$

For property 2,

$$\begin{aligned} q\text{-}\max_x(f(x)) &= \max_{P\in\Delta}\left[\mathop{\mathbb{E}}_{X\sim P}[f(X)] + S_q(P)\right] \\ &= \mathop{\mathbb{E}}_{X\sim P^\star(f)}[f(X)] + S_q(P^\star(f)) \le \mathop{\mathbb{E}}_{X\sim P^\star(f)}[g(X)] + S_q(P^\star(f)) \\ &\quad (\because f \preceq g) \\ &\le \max_{P'\in\Delta}\left[\mathop{\mathbb{E}}_{X\sim P'}[g(X)] + S_q(P')\right] = q\text{-}\max_x(f(x)), \end{aligned} \tag{C.44}$$

where $P^\star(f)$ indicates the optimal distribution of $q\text{-}\max_x(f(x))$. □

Lemma 18. *For $F : \mathcal{S}\times\mathcal{A} \to R$ and $c \in R$, $\mathcal{T}_q(F + c\mathbf{1}) = \mathcal{T}_qF + \gamma c\mathbf{1}$ where $\mathbf{1} : \mathcal{S}\times\mathcal{A} \to 1$*

Proof. For all s, a,

$$\begin{aligned} V_{F+c\mathbf{1}}(s) &= q\text{-}\max_{a'}\big(F(s,a') + c\big) = q\text{-}\max_{a'}\big(F(s,a')\big) + c = V_F(s) + c \\ [\mathcal{T}_qF + c\mathbf{1}](s,a) &= \mathop{\mathbb{E}}_{s'\sim P}\big[r(s,a,s') + \gamma V_{F+c\mathbf{1}}(s')\big|s,a\big] \\ &= \mathop{\mathbb{E}}_{s'\sim P}\big[r(s,a,s') + \gamma V_F(s') + \gamma c\big|s,a\big] \\ &= \mathop{\mathbb{E}}_{s'\sim P}\big[r(s,a,s') + \gamma V_F(s')\big|s,a\big] + \gamma c = [\mathcal{T}_qF](s,a) + \gamma c \end{aligned} \tag{C.45}$$

□

Lemma 19. *For $F, G : \mathcal{S}\times\mathcal{A} \to R$ and $F \succeq G$, $\mathcal{T}_q(F) \succeq \mathcal{T}_q(G)$ where $\succeq$ indicates $F(s,a) \ge G(s,a)$ for all s, a.*

Proof. For all s, a,

$$\begin{aligned} V_F(s) &= q\text{-}\max_{a'}\big(F(s,a')\big) \le q\text{-}\max_{a'}\big(G(s,a')\big) = V_G(s) \\ [\mathcal{T}_qF](s,a) &= \mathop{\mathbb{E}}_{s'\sim P}\big[r(s,a,s') + \gamma V_F(s')\big|s,a\big] \\ &\le \mathop{\mathbb{E}}_{s'\sim P}\big[r(s,a,s') + \gamma V_G(s')\big|s,a\big] = [\mathcal{T}_qG](s,a) \end{aligned} \tag{C.46}$$

□

Lemma 20. *$\mathcal{T}_q$ is γ-contraction mapping in $(C(\mathcal{S}\times\mathcal{A},R),|\cdot|_\infty)$ where $C(\mathcal{S}\times\mathcal{A},R)\triangleq\{F:\mathcal{S}\times\mathcal{A}\to R\}$ and $|F-G|_\infty=\sup_{s,a}|F(s,a)-G(s,a)|$*

Proof. Let $d=|F-G|_\infty$. The, $G-d\mathbf{1}\succeq F\succeq G+d\mathbf{1}$. From Lemma 15, $\mathcal{T}_q(G+d\mathbf{1})\succeq\mathcal{T}_qF\succeq\mathcal{T}_q(G-d\mathbf{1})$. From Lemma 14, $\mathcal{T}_qG+\gamma d\mathbf{1}\succeq\mathcal{T}_qF\succeq\mathcal{T}_qG-\gamma d\mathbf{1}$. Then,$\gamma d\mathbf{1}\succeq\mathcal{T}_qF-\mathcal{T}_qG\succeq-\gamma d\mathbf{1}$. Finally,

$$|\mathcal{T}_qF-\mathcal{T}_qG|_\infty\leq\gamma d=\gamma|F-G|_\infty.$$

Consequently, $\mathcal{T}_q$ is γ-contraction. □

Proof of Tsallis Value Iteration

Proof of Theorem 18. From Lemma 20, $\mathcal{T}_q$ is γ-contraction and has an unique fixed point $F_*=\mathcal{T}_qF_*$ from the Banach fixed point theorem. Then, for any initial function F, a sequence of F_k converges to the fixed point, i.e., $F_*=\lim_{k\to\infty}(\mathcal{T}_q)^kF_0$. The fixed point F_* satisfies a TBO equation as follows:

$$\begin{aligned}F_*(s,a)&=\mathop{\mathbb{E}}_{s'\sim P}[\mathbf{r}(s,a,s')+\gamma V_{F_*}(s')|s,a]\\ V_{F_*}(s)&=q\text{-}\max_a[F_*(s,a)],\end{aligned}\tag{C.47}$$

Since TBO equation is the necessary and sufficient conditions,$F_*=Q_q^\star$. □

C.10 Performance Error Bounds

Lemma 21. *Let*

$$[\mathcal{T}F](s,a)\triangleq\mathop{\mathbb{E}}_{s'\sim P}[\mathbf{r}(s,a,s')+\gamma\max_{a'}F(s',a')|s,a]$$

for a function F. $\mathcal{T}$ is the original Bellman optimality operator which is used for an original value iteration. Then, for all positive integer k and any function F over $\mathcal{S}\times\mathcal{A}$,

$$\mathcal{T}_q^kF\succeq\mathcal{T}^kF$$

where $\mathcal{T}^k$ indicates k tiems application of $\mathcal{T}$. Furthermore, $V_q^\star \succeq V^\star$ holds which means that the optimal value of Tsallis MDP is greater than the optimal value of the original MDP.

Proof. When $k = 1$, from Lemma 17, for all s, a,

$$\begin{aligned}[\mathcal{T}F](s,a) &= \mathop{\mathbb{E}}_{s'\sim P}[\mathbf{r}(s,a,s') + \gamma \max_{a'} F(s',a')|s,a] \\ &\leq \mathop{\mathbb{E}}_{s'\sim P}\left[\mathbf{r}(s,a,s') + \gamma \, q\text{-}\max_{a'} F(s',a')|s,a\right] = [\mathcal{T}_q F](s,a)\end{aligned} \tag{C.48}$$

Now, assume that the statement holds when $k = n$, then,

$$\mathcal{T}^{n+1}F = \mathcal{T}\mathcal{T}^n F \preceq \mathcal{T}_q\mathcal{T}^n F \preceq \mathcal{T}_q\mathcal{T}_q^n F \preceq \mathcal{T}_q^{n+1} F \tag{C.49}$$

From mathematical induction, the statement holds for all positive integers. Furthermore,

$$V^\star = \lim_{k\to\infty} \mathcal{T}^k F \preceq \lim_{k\to\infty} \mathcal{T}_q^k F = V_q^\star$$

□

We would like to note that the gap between $V^\star$ and $V_q^\star$ is induced from the Tsallis entropy.

Proof of Performance Error Bounds

Proof of Theorem 19. The upper bound is trivial. Since the original MDP maximizes $J(\pi)$ without the entropy maximization, it is clear that $J(\pi_q^\star) \leq J(\pi^\star)$

where $J(\pi) \triangleq \mathbb{E}_{\tau\sim\pi,P}[\sum_{t=0}^{\infty}\gamma^t\mathbf{R}_t]$. For the lower bound, using Lemma 21,

$$\begin{aligned}
J(\pi^\star) = \mathbb{E}_{s_0\sim d}\left[V^\star(s_0)\right] \leq \mathbb{E}_{s_0\sim d}\left[V_q^\star(s_0)\right] &= J(\pi_q^\star) + S_q^\infty\left(\pi_q^\star\right) \\
&\leq J(\pi_q^\star) + \mathbb{E}_{\tau\sim\pi,P}\left[\sum_{t=0}^{\infty}\gamma^t S_q\left(\pi_q^\star(\cdot|s_t)\right)\right] \\
&\leq J(\pi_q^\star) + \mathbb{E}_{\tau\sim\pi,P}\left[\sum_{t=0}^{\infty}\gamma^t \max_{\pi(\cdot|s_t)} S_q\left(\pi(\cdot|s_t)\right)\right] \\
&\leq J(\pi_q^\star) - \mathbb{E}_{\tau\sim\pi,P}\left[\sum_{t=0}^{\infty}\gamma^t \ln_q\left(1/|\mathcal{A}|\right)\right] \\
&\leq J(\pi_q^\star) - (1-\gamma)^{-1}\ln_q\left(1/|\mathcal{A}|\right)
\end{aligned} \tag{C.50}$$

□

C.11 q-Scheduling

Proof of Theorem 20. The proof directly follows from Theorem 19.

$$\begin{aligned}
J(\pi^\star) + (1-\gamma)^{-1}\ln_{q_k}\left(1/|\mathcal{A}|\right) &\leq J(\pi_k) \\
J(\pi^\star) + (1-\gamma)^{-1}\lim_{k\to\infty}\ln_{q_k}\left(1/|\mathcal{A}|\right) &\leq \lim_{k\to\infty} J(\pi_k) \\
J(\pi^\star) &\leq \lim_{k\to\infty} J(\pi_k) \;\; (\because \lim_{k\to\infty}\ln_{q_k}\left(1/|\mathcal{A}|\right) = 0) \\
\therefore J(\pi^\star) &= \lim_{k\to\infty} J(\pi_k)
\end{aligned} \tag{C.51}$$

□

Appendix D

Proofs of Chapter 4.2.

D.1 Infinite Exploration

Before starting the proof of Theorem 21, we first prove the following Lemma.

Lemma 22. *The policy of SERN has a constant lower bound greater than zero, i.e.,* $[\pi_t]_a \geq c > 0$*, where* $c = \frac{1}{K}\exp(-\frac{1}{\alpha})$*.*

Proof of Lemma 22. For each round, the proposed method samples an action from

$$\pi_t := \arg\max_{\pi} \left\{ \mathop{\mathbb{E}}_{a\sim\pi} \left[\hat{r}_a(s_t;\theta_{t-1})\right] + \alpha S(\pi) \right\}.$$

Thus, the policy distribution is the optimal solution of

$$\max_{\pi} \left\{ \mathop{\mathbb{E}}_{a\sim\pi} \left[\hat{r}_a(s_t;\theta_{t-1})\right] + \alpha S(\pi) \right\}$$

which is a concave maximization problem since $\mathbb{E}_{a\sim\pi}\left[\hat{r}_a(s_t;\theta_{t-1})\right]$ is linear for π and $\alpha S(\pi)$ is concave for π. The domain of this problem has two constraints, i.e., $\sum_a \pi_a - 1 = 0$ and $\pi_a \geq 0$. Since the problem is concave, strong duality holds and let us denote a dual variable for $\sum_a \pi_a - 1 = 0$ as μ and dual variable for positivity $\pi_a \geq 0$ as λ_a. Then, from Karush-Kuhn-Tucker (KKT) conditions, we have

$$\hat{r}_a(s_t; \theta_{t-1}) - \alpha \ln(\pi_a) - \alpha + \lambda_a + \mu = 0.$$

We first compute μ by multiplying π_a to both sides and summing up with respect to a. Then, $\mu = \alpha - \alpha S(\pi) - \mathbb{E}_{a\sim\pi}\left[\hat{r}_a(s_t; \theta_{t-1})\right]$ where $\lambda_a \pi_a = 0$, one of KKT conditions, is used. By using $S(\pi) \leq -\ln(1/K)$ and $\mathbb{E}_{a\sim\pi}\left[\hat{r}_a(s_t; \theta_{t-1})\right] \leq 1$, $\mu \geq \alpha + \alpha \ln(1/K) - 1$. Since $\ln(x)$ requires $x > 0$ and for all a, $\pi_a > 0$ holds, $\lambda_a = 0$ for all a from KKT conditions. Thus,

$$\ln(\pi_a) = \frac{\hat{r}_a(s_t; \theta_{t-1}) - \alpha + \mu}{\alpha} \geq \ln(1/K) - \frac{1}{\alpha}$$

where $\hat{r}_a \geq 0$. Finally, we get

$$\pi_a \geq \frac{1}{K} \exp\left(-\frac{1}{\alpha}\right).$$

□

The proof of Theorem 21 is as follows.

Proof of Theorem 21. Using Lemma 22, for all t and a, $[\pi_t]_a \geq c$ where $c = \frac{1}{K}\exp(-\frac{1}{\alpha})$. Thus, $\mathbb{E}\left[N_a(t)\right] = \sum_t [\pi_t]_a \geq ct$. □

Proof of Theorem 22. Let $N'_a(t) = N_a(t) - ct$. To prove that $N'_a(t)$ is sub-Martingale, we need to check $\mathbb{E}[N'_a(t)|N'_a(t-1)] \geq N'_a(t-1)$. The inequality holds as follows:

$$\begin{aligned}
\mathbb{E}[N'_a(t)|N'_a(t-1)] &= \mathbb{E}[N_a(t) - ct|N'_a(t-1)] \\
&= \mathbb{E}[N_a(t-1) - c(t-1) + \mathbb{I}(a_t = a) - c|N'_a(t-1)] \\
&= N'_a(t-1) + \mathbb{E}[\mathbb{I}(a_t = a) - c|N'_a(t-1)] \\
&= N'_a(t-1) + [\pi_t]_a - c \\
&\geq N'_a(t-1) \;\; (\because [\pi_t]_a \geq c).
\end{aligned}$$

For sub-Martingale random variable, since $|N'_a(t) - N'_a(t-1)| < 1 + c < 2$ for all t, Azuma-Hoeffding inequality holds, $\mathbb{P}\left(N'_a(t) - N'_a(0) \leq -\delta\right) = \mathbb{P}\left(N_a(t) \leq ct - \delta\right) \leq \exp\left(-\frac{\delta^2}{8t}\right)$. □

D.2 Regret Bound

Before proving the regret bound, we introduce a new lemma for our policy distribution.

Lemma 23. *For any vector $r \in \mathbb{R}^{|\mathcal{A}|}$, let a distribution be*

$$\pi := \arg\max_{\pi'} \left\{ \mathop{\mathbb{E}}_{a \sim \pi'} [r_a] + \alpha S(\pi') \right\}.$$

Then,

$$\max_a r_a - \mathbb{E}_{a \sim \pi} [r_a] \leq \alpha \ln(K)$$

where $K = |\mathcal{A}|$

Proof of Lemma 23. Let $\pi'' := \arg\max_{\pi'} \mathbb{E}_{a \sim \pi'} [r_a]$, Then,

$$\begin{aligned}
\max_a r_a &= \mathop{\mathbb{E}}_{a \sim \pi''} [r_a] = \mathop{\mathbb{E}}_{a \sim \pi''} [r_a] + \alpha S(\pi'') \;\; (\because S(\pi'') = 0) \\
&\leq \mathop{\mathbb{E}}_{a \sim \pi} [r_a] + \alpha S(\pi) \leq \mathop{\mathbb{E}}_{a \sim \pi} [r_a] + \alpha \max_{\pi'} S(\pi') \\
&= \mathop{\mathbb{E}}_{a \sim \pi} [r_a] + \alpha \ln(K)
\end{aligned}$$

Consequently, $\max_a r_a - \mathbb{E}_{a \sim \pi} [r_a] \leq \alpha \ln(K)$ □

By using this Lemma, we prove the Theorem 23.

Proof of Theorem 23.

$$\begin{aligned}\mathcal{R}_T &= \mathop{\mathbb{E}}_{s_{1:T},a_{1:T}}\left[\sum_{t=1}^{T}\max_{a'} r_{a'}(s_t) - r_{a_t}(s_t)\right]\\ &\leq \sum_{t=1}^{T}\max_{a'}\mathop{\mathbb{E}}_{s_{1:T}}\left[r_{a'}(s_t)\right] - \mathop{\mathbb{E}}_{s_{1:T},a_{1:T}}\left[r_{a_t}(s_t)\right]\\ &\leq \sum_{t=1}^{T}\max_{a'}\mathop{\mathbb{E}}_{s_t}\left[r_{a'}(s_t)\right] - \mathop{\mathbb{E}}_{s_t,a_{1:t}}\left[r_{a_t}(s_t)\right].\end{aligned}$$

We first compute the bound of the regret for each round $\max_{a'}\mathbb{E}_{s_t}\left[r_{a'}(s_t)\right] - \mathbb{E}_{s_t,a_{1:t}}\left[r_{a_t}(s_t)\right]$.

Let us define $a^\star := \arg\max_{a'}\mathbb{E}_s\left[r_{a'}(s)\right]$ and $\hat{a}^\star_{t-1} := \arg\max_{a'}\mathbb{E}_s\left[\hat{r}_{a'}(s;\theta_{t-1})\right]$. Then, the regret at round t is

$$\max_{a'}\mathop{\mathbb{E}}_{s_t}\left[r_{a'}(s_t)\right] - \mathop{\mathbb{E}}_{s_t,a_{1:t}}\left[r_{a_t}(s_t)\right] = \mathop{\mathbb{E}}_{s_t}\left[r_{a^\star}(s_t)\right] - \mathop{\mathbb{E}}_{s_{1:t},a_{1:t}}\left[\hat{r}_{a^\star}(s_t;\theta_{t-1})\right] \tag{D.1}$$

$$+ \mathop{\mathbb{E}}_{s_{1:t},a_{1:t}}\left[\hat{r}_{a^\star}(s_t;\theta_{t-1})\right] - \mathop{\mathbb{E}}_{s_{1:t},a_{1:t}}\left[\hat{r}_{\hat{a}^\star_{t-1}}(s_t;\theta_{t-1})\right] \tag{D.2}$$

$$+ \mathop{\mathbb{E}}_{s_{1:t},a_{1:t}}\left[\hat{r}_{\hat{a}^\star_{t-1}}(s_t;\theta_{t-1})\right] - \mathop{\mathbb{E}}_{s_{1:t},a_{1:t}}\left[\hat{r}_{a_t}(s_t;\theta_{t-1})\right] \tag{D.3}$$

$$+ \mathop{\mathbb{E}}_{s_{1:t},a_{1:t}}\left[\hat{r}_{a_t}(s_t;\theta_{t-1})\right] - \mathop{\mathbb{E}}_{s_t,a_{1:t}}\left[r_{a_t}(s_t)\right]. \tag{D.4}$$

From Assumption 3, the (D.1) and (D.4) terms are caused by an estimation error and are bounded as follows:

$$\begin{aligned}\mathop{\mathbb{E}}_{s_{1:t},a_{1:t}}\left[\hat{r}_{a_t}(s_t;\theta_{t-1}) - r_{a_t}(s_t;\theta_{t-1})\right] &\leq \mathop{\mathbb{E}}_{s_{1:t},a_{1:t}}\left[\left|\hat{r}_{a_t}(s_t;\theta_{t-1}) - r_{a_t}(s_t;\theta_{t-1})\right|\right]\\ &\leq \beta\mathop{\mathbb{E}}_{s_{1:t},a_{1:t}}\left[\frac{1}{\sqrt{(N_{a_t}(t-1)+1)}}\right]\end{aligned}$$

and, similarly,

$$\mathop{\mathbb{E}}_{s_{1:t},a_{1:t}}\left[\hat{r}_{a^\star}(s_t;\theta_{t-1}) - r_{a^\star}(s_t;\theta_{t-1})\right] \leq \beta\mathop{\mathbb{E}}_{s_{1:t},a_{1:t}}\left[\frac{1}{\sqrt{(N_{a^\star}(t-1)+1)}}\right].$$

the (D.2) term comes from the failure probability for classifying the optimal action using $\hat{r}_a(s_t)$. Thus, we can rewrite it as follows:

$$\begin{aligned}\mathop{\mathbb{E}}_{s_{1:t},a_{1:t}}\left[\hat{r}_{a^\star}(s_t;\theta_{t-1})\right] - \mathop{\mathbb{E}}_{s_{1:t},a_{1:t}}&\left[\hat{r}_{\hat{a}^\star_{t-1}}(s_t;\theta_{t-1})\right] \\ &= \mathop{\mathbb{E}}_{s_{1:t},a_{1:t}}\left[\mathbb{I}(a^\star \neq \hat{a}^\star_{t-1})(\hat{r}_{a^\star}(s_t;\theta_{t-1}) - \hat{r}_{\hat{a}^\star_{t-1}}(s_t;\theta_{t-1}))\right] \\ &\leq \mathop{\mathbb{E}}_{s_{1:t},a_{1:t}}\left[\mathbb{I}(a^\star \neq \hat{a}^\star_{t-1})\right] = \mathbb{P}(a^\star \neq \hat{a}^\star_{t-1}).\end{aligned}$$

The (D.3) term is bounded by Lemma 23,

$$\begin{aligned}\mathop{\mathbb{E}}_{s_{1:t},a_{1:t}}\left[\hat{r}_{\hat{a}^\star_{t-1}}(s_t;\theta_{t-1})\right] - \mathop{\mathbb{E}}_{s_{1:t},a_{1:t}}&\left[\hat{r}_{a_t}(s_t;\theta_{t-1})\right] \\ &\leq \max_a \mathop{\mathbb{E}}_{s_{1:t},a_{1:t}}\left[\hat{r}_a(s_t;\theta_{t-1})\right] - \mathop{\mathbb{E}}_{a_t\sim\pi_t}\mathop{\mathbb{E}}_{s_{1:t},a_{1:t-1}}\left[\hat{r}_{a_t}(s_t;\theta_{t-1})\right] \\ &\leq \alpha\ln(K)\end{aligned}$$

Finally, we have,

$$\begin{aligned}\max_{a'}\mathop{\mathbb{E}}_{s_t}\left[r_{a'}(s_t)\right] - \mathop{\mathbb{E}}_{s_t,a_{1:t}}\left[r_{a_t}(s_t)\right] \leq& \beta \mathop{\mathbb{E}}_{s_{1:t},a_{1:t}}\left[\frac{1}{\sqrt{(N_{a^\star}(t-1)+1)}}\right] \\ &+ \beta \mathop{\mathbb{E}}_{s_{1:t},a_{1:t}}\left[\frac{1}{\sqrt{(N_{a_t}(t-1)+1)}}\right] \\ &+ \mathbb{P}(a^\star \neq \hat{a}^\star_{t-1}) + \alpha\ln(K).\end{aligned}$$

Consequently, for the expected cumulative regret,

$$\begin{aligned}\mathcal{R}_T \leq& \beta\sum_{t=1}^T \mathop{\mathbb{E}}_{s_{1:t},a_{1:t}}\left[\frac{1}{\sqrt{(N_{a^\star}(t-1)+1)}}\right] + \beta\sum_{t=1}^T \mathop{\mathbb{E}}_{s_{1:t},a_{1:t}}\left[\frac{1}{\sqrt{(N_{a_t}(t-1)+1)}}\right] \\ &+ \sum_{t=1}^T \mathbb{P}(a^\star \neq \hat{a}^\star_{t-1}) + \alpha\ln(K)T.\end{aligned}$$

□

Proof of Theorem 24. From Theorem 23, it is known that the expected regret is bounded by three terms: estimation error, the failure probability, and regularization. For $\mathbb{E}_{s_{1:t},a_{1:t}}\left[\frac{1}{\sqrt{(N_a(t-1)+1)}}\right]$, since the proposed method explores every arms infinitely, estimation errors of all arms become zero. Now, for any a, we can compute the upper bound by using Theorem 21 and 22,

$$\begin{aligned}
\mathop{\mathbb{E}}_{s_{1:t},a_{1:t}}\left[\frac{1}{\sqrt{(N_a(t-1)+1)}}\right] &= \mathop{\mathbb{E}}_{s_{1:t},a_{1:t}}\left[\frac{1}{\sqrt{(N_a(t-1)+1)}}\mathbb{I}\left(N_a(t-1)>\frac{ct}{2}\right)\right] \\
&\quad + \mathop{\mathbb{E}}_{s_{1:t},a_{1:t}}\left[\frac{1}{\sqrt{(N_a(t-1)+1)}}\mathbb{I}\left(N_a(t-1)\le\frac{ct}{2}\right)\right] \\
&\le \mathop{\mathbb{E}}_{s_{1:t},a_{1:t}}\left[\sqrt{\frac{2}{ct}}\mathbb{I}\left(N_a(t-1)>\frac{ct}{2}\right)\right] \\
&\quad + \mathop{\mathbb{E}}_{s_{1:t},a_{1:t}}\left[\mathbb{I}\left(N_a(t-1)\le\frac{ct}{2}\right)\right] \\
&\le \sqrt{\frac{2}{ct}}\mathbb{P}\left(N_a(t-1)>\frac{ct}{2}\right)+\mathbb{P}\left(N_a(t-1)\le\frac{ct}{2}\right) \\
&\le \sqrt{\frac{2}{ct}}\cdot\frac{2\mathbb{E}\left[N_a(t-1)\right]}{ct}+\mathbb{P}\left(N_a(t-1)\le\frac{ct}{2}\right) \\
&\le \sqrt{\frac{2}{ct}}\cdot\frac{2t}{ct}+\mathbb{P}\left(N_a(t-1)\le\frac{ct}{2}\right) \\
&\le \frac{2^{3/2}}{c^{3/2}}\frac{1}{\sqrt{t}}+\exp\left(-\frac{c^2t}{32}\right)
\end{aligned}$$

where for the last inequality we use the Markov inequality and the Azuma Hoeffding inequality, respectively. Finally, we get

$$\begin{aligned}
\mathop{\mathbb{E}}_{s_{1:t-1},a_{1:t}}\left[\frac{1}{\sqrt{(N_{a_t}(t-1)+1)}}\right] &= \mathop{\mathbb{E}}_{a_t}\left[\mathop{\mathbb{E}}_{s_{1:t-1},a_{1:t-1}}\left[\frac{1}{\sqrt{(N_{a_t}(t-1)+1)}}\right]\right] \\
&\le \mathop{\mathbb{E}}_{a_t}\left[\frac{2^{3/2}}{c^{3/2}}\frac{1}{\sqrt{t}}+\exp\left(-\frac{c^2t}{32}\right)\right] \\
&\le \frac{2^{3/2}}{c^{3/2}}\frac{1}{\sqrt{t}}+\exp\left(-\frac{c^2t}{32}\right)
\end{aligned}$$

and

$$\begin{aligned}
\mathop{\mathbb{E}}_{s_{1:t-1},a_{1:t}}\left[\frac{1}{\sqrt{(N_{a^\star}(t-1)+1)}}\right] &= \mathop{\mathbb{E}}_{a_t}\left[\mathop{\mathbb{E}}_{s_{1:t-1},a_{1:t-1}}\left[\frac{1}{\sqrt{(N_{a^\star}(t-1)+1)}}\right]\right] \\
&\le \mathop{\mathbb{E}}_{a_t}\left[\frac{2^{3/2}}{c^{3/2}}\frac{1}{\sqrt{t}}+\exp\left(-\frac{c^2t}{32}\right)\right] \\
&\le \frac{2^{3/2}}{c^{3/2}}\frac{1}{\sqrt{t}}+\exp\left(-\frac{c^2t}{32}\right).
\end{aligned}$$

For the failure probability $\mathbb{P}(a^\star \neq \hat{a}^\star_{t-1})$, let us define an estimation error bound of Assumption 3 as $\beta_{N_a(t-1)} := \frac{\beta}{\sqrt{N_a(t-1)+1}}$. We obtain the bound as follows:

$$\begin{aligned}
\mathbb{P}\left(a^\star \neq \hat{a}^\star_{t-1}\right) &= \mathbb{P}\left(\hat{r}_{a^\star}(s_t) < \hat{r}_{\hat{a}^\star_{t-1}}(s_t)\right) \\
&\leq \sum_{a \neq a^\star} \mathbb{P}\left(\hat{r}_{a^\star}(s_t) < \hat{r}_a(s_t)\right) \\
&\leq \sum_{a \neq a^\star} \mathbb{P}\left(r_{a^\star}(s_t) - \beta_{N_{a^\star}(t-1)} < r_a(s_t) + \beta_{N_a(t-1)}\right) \\
&\leq \sum_{a \neq a^\star} \mathbb{P}\left(\Delta_a(s_t) < \beta_{N_{a^\star}(t-1)} + \beta_{N_a(t-1)}\right) \\
&\leq \sum_{a \neq a^\star} \mathbb{P}\left(\Delta_2 < \beta_{N_{a^\star}(t-1)} + \beta_{N_a(t-1)}\right) \\
&\leq \sum_{a \neq a^\star} \mathbb{P}\left(\frac{\Delta_2}{2} < \beta_{N_{a^\star}(t-1)}\right) + \mathbb{P}\left(\frac{\Delta_2}{2} < \beta_{N_a(t-1)}\right).
\end{aligned}$$

Now, we can bound $\mathbb{P}\left(\frac{\Delta_2}{2} < \beta_{N_a(t-1)}\right)$ using Theorem 22,

$$\begin{aligned}
\mathbb{P}\left(\frac{\Delta_2}{2} < \beta_{N_a(t-1)}\right) &= \mathbb{P}\left(N_a(t-1) < \left(\frac{2\beta}{\Delta_2}\right)^2 - 1\right) \\
&\leq \exp\left(-\frac{(ct - (2\beta/\Delta_2)^2 + 1)^2}{8t}\right) \\
&= \exp\left(-\frac{c^2 t}{8} + \frac{(2\beta/\Delta_2)^2 - 1}{4} - \frac{((2\beta/\Delta_2)^2 - 1)^2}{8t}\right) \\
&\leq \exp\left(\frac{(2\beta/\Delta_2)^2 - 1}{4}\right) \exp\left(-\frac{c^2 t}{8}\right)
\end{aligned}$$

Hence, we get,

$$\begin{aligned}
\mathbb{P}\left(a^\star \neq \hat{a}^\star_{t-1}\right) &\leq \sum_{a \neq a^\star} 2 \exp\left(\frac{(2\beta/\Delta_2)^2 - 1}{4}\right) \exp\left(-\frac{c^2 t}{8}\right) \\
&= 2(K-1) \exp\left((\beta/\Delta_2)^2 - 1/4\right) \exp\left(-\frac{c^2 t}{8}\right)
\end{aligned}$$

Let $C_0 = 2^{\frac{7}{2}} K^{\frac{3}{2}} \beta$, $C_1 = 2\beta$, $C_2 = 2(K-1)\exp((\beta/\Delta_2)^2 - 1/4)$, $d_1 = 1/(32K^2)$, and $d_2 = 1/(8K^2)$. By combining all bounds, $\mathcal{R}_T$ can be bounded as follows:

$$\begin{aligned}
\mathcal{R}_T \le & \frac{2^{5/2}\beta}{c^{3/2}} \sum_{t=1}^{T} \frac{1}{\sqrt{t}} + 2\beta \sum_{t=1}^{T} \exp\left(-\frac{c^2 t}{32}\right) \\
& + 2(K-1)\exp\left((\beta/\Delta_2)^2 - 1/4\right) \sum_{t=1}^{T} \exp\left(-\frac{c^2 t}{8}\right) + \alpha\ln(K)T \\
= & \frac{C_0 K^{-3/2}/2}{c^{3/2}} \sum_{t=1}^{T} \frac{1}{\sqrt{t}} + C_1 \sum_{t=1}^{T} \exp\left(-\frac{c^2 t}{32}\right) \\
& + C_2 \sum_{t=1}^{T} \exp\left(-\frac{c^2 t}{8}\right) + \alpha\ln(K)T \\
\le & \frac{C_0 K^{-3/2}/2}{c^{3/2}} (1 + 2\sqrt{T} - 2\sqrt{2}) \\
& + C_1 \frac{\exp\left(-c^2 T/32\right) - 1}{\exp\left(-c^2/32\right) - 1} + C_2 \frac{\exp\left(-c^2 T/8\right) - 1}{\exp\left(-c^2/8\right) - 1} + \alpha\ln(K)T \\
\le & \frac{C_0 K^{-3/2}}{c^{3/2}} \sqrt{T} + \frac{C_1}{1 - \exp\left(-c^2/32\right)} + \frac{C_2}{1 - \exp\left(-c^2/8\right)} + \alpha\ln(K)T.
\end{aligned}$$

Note that all terms are sub-linear except for $\alpha\ln(K)T$. To make $\alpha\ln(K)T$ sub-linear, we set α to be $\alpha_0(\ln(T^p))^{-1}$ with $\alpha_0 > 0$. Then, the lower bound c becomes $\frac{\exp\left(-\frac{1}{\alpha_0}\right)}{KT^p}$ and let $c_0 := \exp\left(-\frac{1}{\alpha_0}\right)$. Finally,

$$\begin{aligned}
\mathcal{R}_T \le & \frac{C_0 K^{-3/2}}{c^{3/2}} \sqrt{T} + \frac{C_1}{1 - \exp\left(-c^2/32\right)} + \frac{C_2}{1 - \exp\left(-c^2/8\right)} + \alpha\ln(K)T \\
\le & \frac{C_0}{c_0^{3/2}} T^{\frac{3p+1}{2}} + C_1(1 - \exp(-T^{-2p} \cdot c_0^2/(32K^2)))^{-1} \\
& + C_2(1 - \exp(-T^{-2p} \cdot c_0^2/(8K^2)))^{-1} + \alpha_0 \ln(K)T(\ln(T^p))^{-1} \\
\le & \frac{C_0}{c_0^{3/2}} T^{\frac{3p+1}{2}} + C_1(1 - \exp(-c_0^2 d_1 T^{-2p}))^{-1} \\
& + C_2(1 - \exp(-c_0^2 d_2 T^{-2p}))^{-1} + \alpha_0 \ln(K)T(\ln(T^p))^{-1}.
\end{aligned}$$

□

Proof of Theorem 25. To prove that $\lim_{T\to\infty} \frac{\mathcal{R}_T}{T} = 0$, we show that the upper bound of $\mathcal{R}_T/T$ converges to zero, then, proof will be done since the lower bound of $\mathcal{R}_T/T$ is also zero.

$$\begin{aligned}\frac{\mathcal{R}_T}{T} \leq & \frac{C_0}{c_0^{3/2}} T^{\frac{3p-1}{2}} + C_1(1-\exp(-d_1 T^{-2p}))^{-1} T^{-1} \\ & + C_2(1-\exp(-d_2 T^{-2p}))^{-1} T^{-1} \\ & + \ln(K)(\ln(T^p))^{-1}.\end{aligned}$$

Since $1/3 > p > 0$, $T_{(3p-1)/2}$ converges to zero and $\ln(T^p)^{-1}$ also converges to zero. To show that the second and third terms converge to zero, we prove that, for a positive a, $\lim_{x\to\infty}(1-\exp(-ax^{-2p})x)^{-1}x^{-1} = 0$ as follows:

$$\lim_{x\to\infty}(1-\exp(-ax^{-2p}))^{-1}ax^{-2p} \cdot x^{2p-1}/a = 1 \cdot 0 = 0$$

where $\lim_{z\to 0} \frac{z}{\exp(z)-1} = 1$ is used. □

Appendix E

Proofs of Chapter 5.1.

E.1 General Regret Lower Bound of APE

Proof of Theorem 26. We construct a K-armed multi-armed bandit problem with deterministic rewards. Let the optimal arm $a^\star$ give the reward of $\Delta = \frac{1}{2}\sqrt{\frac{c(K-1)}{T}}F^{-1}\left(1 - \frac{1}{K}\right)$ whereas the other arms provide zero rewards. Note that $\Delta \in [0,1]$ for $T \geq \frac{c(K-1)}{4}\left|F^{-1}\left(1-\frac{1}{K}\right)\right|^2$ and the estimator becomes $\hat{r}_a = \Delta\mathbb{I}[a = a^\star]$ since there is no noise. Let E_t be the set of events which satisfy

$$\sum_{a \neq a^\star} n_{t,a} \leq cT$$

If $\mathbb{P}(E_t) \leq 1/2$, then the regret bound is computed as follows

$$\mathbb{E}[\mathcal{R}_T] \geq \frac{1}{2}\mathbb{E}[\mathcal{R}_t|E_t^c] \geq \frac{cT}{2}\Delta = \frac{c\sqrt{c}}{4}\sqrt{(K-1)T}F^{-1}\left(1 - \frac{1}{K}\right)$$

hence it satisfies the lower bound. Otherwise, it is sufficient to prove $\mathbb{P}(a_t \neq a^\star) \geq 1/8$. Then it holds

$$\mathbb{E}[\mathcal{R}_T] = \sum_{t=1}^{T}\Delta\mathbb{P}(a_t = a^\star) \geq \frac{T}{8}\Delta = \frac{\sqrt{c}}{16}\sqrt{(K-1)T}F^{-1}\left(1 - \frac{1}{K}\right)$$

and we get the desired result since $0 < c < \frac{K-1}{K+3}$.

Observe that

$$\begin{aligned}
&\mathbb{P}(a_t \neq a^\star) \\
&= \mathbb{P}\left(\bigcup_{a \neq a^\star} \{\hat{r}_{a^\star} + \beta_{t,a^\star} G_{t,a^\star} \leq \hat{r}_a + \beta_{t,a} G_{t,a}\}\right) \\
&\geq \mathbb{P}\left(E_{t-1}\right)\mathbb{P}\left(\bigcup_{a \neq a^\star} \{\hat{r}_{a^\star} + \beta_{t,a^\star} G_{t,a^\star} \leq 2\Delta \leq \hat{r}_a + \beta_{t,a} G_{t,a}\} \Big| E_{t-1}\right) \\
&\geq \frac{1}{2}\mathbb{E}\Bigg[\mathbb{P}\left(G_{t,a^\star} \leq \frac{\Delta}{\beta_{t,a^\star}} \Big| \mathcal{H}_{t-1}, E_{t-1}\right) \\
&\times \mathbb{P}\left(\bigcup_{a \neq a^\star} \{2\Delta \leq \beta_{t,a} G_{t,a}\} \Big| \mathcal{H}_{t-1}, E_{t-1}\right) \Bigg| E_{t-1}\Bigg] \\
&\geq \frac{1}{2}\mathbb{E}\Bigg[\mathbb{P}\left(G_{t,a^\star} \leq \frac{\Delta\sqrt{(1-c)T}}{c} \Big| \mathcal{H}_{t-1}, E_{t-1}\right) \\
&\times \mathbb{P}\left(\bigcup_{a \neq a^\star} \{2\Delta \leq \beta_{t,a} G_{t,a}\} \Big| \mathcal{H}_{t-1}, E_{t-1}\right) \Bigg| E_{t-1}\Bigg]
\end{aligned}$$

where the last inequality holds due to $n_{t-1,a^\star} \geq (1-c)T$ provided E_{t-1}. Since $c < \frac{K-1}{K+3}$, we have

$$\frac{\Delta\sqrt{(1-c)T}}{c} = \sqrt{\frac{(1-c)(K-1)}{4c}} F^{-1}\left(1 - \frac{1}{K}\right) > F^{-1}\left(1 - \frac{1}{K}\right).$$

Hence, $\mathbb{P}\left(G_{t,a^\star} \leq \frac{\Delta\sqrt{(1-c)T}}{c} \Big| \mathcal{H}_{t-1}, E_{t-1}\right) \geq 1 - \frac{1}{K}$ so that

$$\mathbb{P}(a_t \neq a^\star) \geq \frac{1}{2}\left(1 - \frac{1}{K}\right)\mathbb{E}\left[\mathbb{P}\left(\bigcup_{a \neq a^\star} \{2\Delta \leq \beta_{t,a} G_{t,a}\} \Big| \mathcal{H}_{t-1}, E_{t-1}\right) \Bigg| E_{t-1}\right].$$

Observe that

$$\begin{aligned}
&\mathbb{P}\left(\bigcup_{a\neq a^\star}\{2\Delta \leq \beta_{t,a}G_{t,a}\}\,\Big|\mathcal{H}_{t-1}, E_{t-1}\right)\\
&\geq 1-\mathbb{P}\left(\bigcap_{a\neq a^\star}\left\{G_{t,a}\leq \frac{2\Delta}{\beta_{t,a}}\right\}\Big|\mathcal{H}_{t-1}, E_{t-1}\right)\\
&\geq 1-\prod_{a\neq a^\star}F\left(\frac{2\Delta\sqrt{n_{t-1,a}}}{c}\right)\\
&\geq 1-\left|F\left(2\Delta\frac{\sum_{a\neq a^\star}\sqrt{n_{t-1,a}}}{c(K-1)}\right)\right|^{K-1},
\end{aligned}$$

where the last inequality holds by the log-concavity of F. Under E_{t-1}, note that

$$\sum_{a\neq a^\star}\sqrt{n_{t-1,a}}\leq\sqrt{(K-1)\sum_{a\neq a^\star}n_{t-1,a}}\leq\sqrt{c(K-1)T}$$

which implies

$$F\left(2\Delta\frac{\sum_{a\neq a^\star}\sqrt{n_{t-1,a}}}{c(K-1)}\right)\leq F\left(2\Delta\sqrt{\frac{T}{c(K-1)}}\right)=1-\frac{1}{K}$$

Therefore, we get

$$\mathbb{P}(a_t\neq a^\star)\geq\frac{1}{2}\left(1-\frac{1}{K}\right)\left(1-\left(1-\frac{1}{K}\right)^{K-1}\right)\geq\frac{1}{8}$$

since $1-\frac{1}{K}\geq\frac{1}{2}$ and $1-\left(1-\frac{1}{K}\right)^{K-1}\geq\frac{1}{2}$ hold for $K\geq 2$ and the theorem is proved. □

E.2 General Regret Upper Bound of APE

In this section, assuming sub-Gaussian reward, we provide the proof of Theorem 27 and the related lemmas.

Proof of Lemma 7. Fix arm $a\in\mathcal{A}$. Let τ_k denotes the smallest round when the arm a is sampled for the k-th time i.e. $k=\sum_{t=1}^{\tau_k}\mathbb{I}[E_{t,a}]$. We let $\tau_0:=0$ and

$\tau_k = T$ for $k > n_a(T)$. Then it is easy to see that for $\tau_k < t \leq \tau_{k+1}$

$$\mathbb{I}[E_{t,a}] = \begin{cases} 1 & : t = \tau_{k+1} \\ 0 & : t \neq \tau_{k+1} \end{cases} \tag{E.1}$$

Therefore,

$$\begin{aligned}\sum_{t=1}^{T} \mathbb{P}\left(E_{t,a}^{(1)}\right) &= \sum_{t=1}^{T} \mathbb{E}\left[\mathbb{I}[E_{t,a}^{(1)}]\right] = \sum_{k=0}^{T-1} \mathbb{E}\left[\sum_{t=1+\tau_k}^{\tau_{k+1}} \mathbb{I}[E_{t,a}^{(1)}]\right] \\ &= \mathbb{E}\left[\sum_{t=1}^{\tau_1} \mathbb{I}\left(E_{t,a} \cap \hat{E}_{t,a}^c\right)\right] + \sum_{k=1}^{T-1} \mathbb{E}\left[\sum_{t=1+\tau_k}^{\tau_{k+1}} \mathbb{I}[E_{t,a} \cap \hat{E}_{t,a}^c]\right] \\ &\leq 1 + \sum_{k=1}^{T-1} \mathbb{P}\left(\hat{E}_{\tau_{k+1},a}^c\right)\end{aligned}$$

where the last inequality holds by the definition of τ_k. Also, by the definition of $\hat{E}_{t,a}$ and Chernoff bounds with sub-Gaussian condition with parameter σ,

$$\sum_{k=1}^{T-1} \mathbb{P}\left(\hat{E}_{\tau_{k+1},a}^c\right) \leq \sum_{k=1}^{T-1} \exp\left(-\frac{\Delta_a^2 k}{18\sigma^2}\right) \leq \frac{18\sigma^2}{\Delta_a^2}$$

which implies the desired result. The lemma is proved. □

Lemma 24 (Proof of Lemma 8)**.** *Suppose (i) of Assumption 1. Then for any action $a \in \mathcal{A}$, it holds*

$$\sum_{t=1}^{T} \mathbb{P}\left(E_{t,a}^{(2)}\right) \leq \left[C_1 + \frac{F(0)}{1 - F(0)}\right] \frac{18\sigma^2}{\Delta_a^2} + \frac{144\sigma^2}{\Delta_a^2} + 2\sum_{k=1}^{T} F\left(-\frac{\Delta_a \sqrt{k}}{6c}\right)$$

Proof. If $a = a^\star$, then $\Delta_a = 0$ so the desired result trivially holds. Threfore, we take $a \in \mathcal{A} \setminus \{a^\star\}$. For notational convenience, we write $\tilde{r}_{t,a} := \hat{r}_{t-1,a} + \beta_{t-1,a} G_{t,a}$. Due to the selection rule, $a_t = a$ implies $\tilde{r}_{t,a'} \leq \tilde{r}_{t,a}$ for $a' \in \mathcal{A}$. Therefore, it holds

$$E_{t,a} \cap \tilde{E}_{t,a} \subset \bigcap_{a' \in \mathcal{A}} \{\tilde{r}_{t,a'} \leq y_a\} = \{\tilde{r}_{t,a^\star} \leq y_a\} \cap \{\tilde{r}_{t,a'} \leq y_a, \forall a' \neq a_\star\} \tag{E.2}$$

This implies

$$\mathbb{P}\left(E_{t,a} \cap \tilde{E}_{t,a} | \mathcal{H}_{t-1}\right) \leq \mathbb{P}\left(\bigcap_{a' \in \mathcal{A}} \{\tilde{r}_{t,a'} \leq y_a\} | \mathcal{H}_{t-1}\right) \tag{E.3}$$

Note that events $\{\tilde{r}_{t,a^\star} \leq y_a\}$ and $\{\tilde{r}_{t,a'} \leq y_a, \forall a' \neq a_\star\}$ are independent when $\mathcal{H}_{t-1}$ is given conditionally (see A.2 in [64] for detail). By applying this fact repeatedly, (E.3) is equal to

$$\begin{aligned}
&\mathbb{P}\left(\bigcap_{a' \in \mathcal{A}} \{\tilde{r}_{t,a'} \leq y_a\} | \mathcal{H}_{t-1}\right) \\
&= \mathbb{P}\left(\tilde{r}_{t,a^\star} \leq y_a | \mathcal{H}_{t-1}\right) \mathbb{P}\left(\tilde{r}_{t,a'} \leq y_a, \forall a' \neq a_\star | \mathcal{H}_{t-1}\right) \\
&= \frac{\mathbb{P}\left(\tilde{r}_{t,a^\star} \leq y_a | \mathcal{H}_{t-1}\right)}{\mathbb{P}\left(\tilde{r}_{t,a^\star} > y_a | \mathcal{H}_{t-1}\right)} \mathbb{P}\left(\tilde{r}_{t,a^\star} > y_a | \mathcal{H}_{t-1}\right) \mathbb{P}\left(\tilde{r}_{t,a'} \leq y_a, \forall a' \neq a_\star | \mathcal{H}_{t-1}\right) \\
&= \frac{\mathbb{P}\left(\tilde{r}_{t,a^\star} \leq y_a | \mathcal{H}_{t-1}\right)}{\mathbb{P}\left(\tilde{r}_{t,a^\star} > y_a | \mathcal{H}_{t-1}\right)} \mathbb{P}\left(\{\tilde{r}_{t,a^\star} > y_a\} \cap \{\tilde{r}_{t,a'} \leq y_a, \forall a' \neq a_\star\} | \mathcal{H}_{t-1}\right)
\end{aligned}$$

Recall F is the cumulative density function of G. Since $\hat{r}_{t-1,a^\star}, \beta_{t-1,a^\star}$ are already determined when $\mathcal{H}_{t-1}$ is given conditionally, we get

$$\mathbb{P}\left(\tilde{r}_{t,a^\star} \leq y_a | \mathcal{H}_{t-1}\right) = F\left(\frac{r_{a^\star} - \hat{r}_{t-1,a^\star} - \frac{\Delta_a}{3}}{\beta_{t-1,a^\star}}\right)$$

To avoid notational complexity, we write $\mathfrak{F}_{t,a^\star} := F\left(\frac{r_{a^\star} - \hat{r}_{t-1,a^\star} - \frac{\Delta_a}{3}}{\beta_{t-1,a^\star}}\right)$. Analogous to (E.2), we can see that

$$\{\tilde{r}_{t,a^\star} > y_a\} \cap \{\tilde{r}_{t,a'} \leq y_a, \forall a' \neq a_\star\} \subset E_{t,a^\star} \cap \tilde{E}_{t,a} \tag{E.4}$$

and this implies

$$\mathbb{P}\left(\{\tilde{r}_{t,a^\star} > y_a\} \cap \{\tilde{r}_{t,a'} \leq y_a, \forall a' \neq a_\star\} | \mathcal{H}_{t-1}\right) \leq \mathbb{P}\left(E_{t,a^\star} \cap \tilde{E}_{t,a} | \mathcal{H}_{t-1}\right) \tag{E.5}$$

Therefore,

$$\mathbb{P}\left(E_{t,a} \cap \tilde{E}_{t,a} | \mathcal{H}_{t-1}\right) \leq \frac{\mathfrak{F}_{t,a^\star}}{1 - \mathfrak{F}_{t,a^\star}} \mathbb{P}\left(E_{t,a^\star} \cap \tilde{E}_{t,a} | \mathcal{H}_{t-1}\right) \tag{E.6}$$

By taking the expection and the definition of conditional expectation, we arrive in

$$\mathbb{P}\left(E_{t,a}^{(2)}\right) = \mathbb{P}\left(E_{t,a} \cap \hat{E}_{t,a} \cap \tilde{E}_{t,a}\right) \leq \mathbb{E}\left[\frac{\mathfrak{F}_{t,a^\star}}{1-\mathfrak{F}_{t,a^\star}} \mathbb{I}[E_{t,a^\star} \cap \hat{E}_{t,a} \cap \tilde{E}_{t,a}]\right] \tag{E.7}$$

Now recall the definition of τ_k from Lemma 26. In this case, we set τ_k denotes the smallest round when the optimal arm $a^\star$ is sampled for the k-th time. Then the summation of the right-hand side of E.7 over $t = 1, \ldots, T$ is controlled by

$$\begin{aligned}
&\sum_{t=1}^{T} \mathbb{E}\left[\frac{\mathfrak{F}_{t,a^\star}}{1-\mathfrak{F}_{t,a^\star}} \mathbb{I}[E_{t,a^\star} \cap \hat{E}_{t,a} \cap \tilde{E}_{t,a}]\right] \\
&\quad = \sum_{k=0}^{T-1} \mathbb{E}\left[\sum_{t=\tau_k+1}^{\tau_{k+1}} \frac{\mathfrak{F}_{t,a^\star}}{1-\mathfrak{F}_{t,a^\star}} \mathbb{I}[E_{t,a^\star} \cap \hat{E}_{t,a} \cap \tilde{E}_{t,a}]\right] \\
&\quad = \sum_{k=0}^{T-1} \mathbb{E}\left[\frac{\mathfrak{F}_{\tau_{k+1},a^\star}}{1-\mathfrak{F}_{\tau_{k+1},a^\star}} \mathbb{I}[E_{t,a^\star} \cap \hat{E}_{t,a} \cap \tilde{E}_{t,a}]\right] \\
&\quad \leq \sum_{k=1}^{T} \mathbb{E}\left[\frac{\mathfrak{F}_{\tau_k,a^\star}}{1-\mathfrak{F}_{\tau_k,a^\star}}\right]
\end{aligned}$$

To derive the upper bound of the above expection terms, we analyze the conditional expectation $\mathbb{E}\left[\frac{\mathfrak{F}_{\tau_k,a^\star}}{1-\mathfrak{F}_{\tau_k,a^\star}} \middle| \mathcal{H}_{\tau_k}\right]$ instead. Due to the definition of τ_k and $\beta_{t,a}$, observe that $n_{\tau_k,a} = k$ and $\beta_{\tau_k,a} = \frac{c}{\sqrt{k}}$. Therefore,

$$\begin{aligned}
\mathbb{E}\left[\frac{\mathfrak{F}_{\tau_k,a^\star}}{1-\mathfrak{F}_{\tau_k,a^\star}} \middle| \mathcal{H}_{\tau_k}\right] &= \mathbb{E}\left[\frac{F\left(\frac{\sqrt{k}}{c}\left\{r_{a^\star} - \hat{r}_{\tau_k,a^\star} - \frac{\Delta_a}{3}\right\}\right)}{1-F\left(\frac{\sqrt{k}}{c}\left\{r_{a^\star} - \hat{r}_{\tau_k,a^\star} - \frac{\Delta_a}{3}\right\}\right)} \middle| \mathcal{H}_{\tau_k}\right] \\
&= \int_{\mathbb{R}} \frac{F\left(\frac{\sqrt{k}}{c}\left\{r_{a^\star} - x - \frac{\Delta_a}{3}\right\}\right)}{1-F\left(\frac{\sqrt{k}}{c}\left\{r_{a^\star} - x - \frac{\Delta_a}{3}\right\}\right)} \mathbb{P}(\hat{r} \in \mathrm{d}x)
\end{aligned} \tag{E.8}$$

We decompose $\mathbb{R} = I_1 \cup I_2 \cup I_3$ into three intervals where $I_1 := \{x \leq r_{a^\star} - \frac{\Delta_a}{3}\}$, $I_2 := \{r_{a^\star} - \frac{\Delta_a}{3} < x \leq r_{a^\star} - \frac{\Delta_a}{6}\}$, and $I_3 := \{r_{a^\star} - \frac{\Delta_a}{6} < x\}$. We derive the upper bound of (E.8) on the each interval.

Due to the change of variable formula,

$$\begin{aligned}
&\int_{I_1} \frac{F\left(\frac{\sqrt{k}}{c}\left\{r_{a^\star} - x - \frac{\Delta_a}{3}\right\}\right)}{1 - F\left(\frac{\sqrt{k}}{c}\left\{r_{a^\star} - x - \frac{\Delta_a}{3}\right\}\right)} \mathbb{P}(\hat{r} \in \mathrm{d}x) \\
&\quad = \int_{-\infty}^{r_{a^\star} - \frac{\Delta_a}{3}} \frac{F\left(\frac{\sqrt{k}}{c}\left\{r_{a^\star} - x - \frac{\Delta_a}{3}\right\}\right)}{1 - F\left(\frac{\sqrt{k}}{c}\left\{r_{a^\star} - x - \frac{\Delta_a}{3}\right\}\right)} f_{\hat{r}}(x)\mathrm{d}x \\
&\quad = \frac{c}{\sqrt{k}} \int_0^\infty \frac{F(g)}{1 - F(g)} f_{\hat{r}}\left(r_{a^\star} - \frac{c}{\sqrt{k}}g - \frac{\Delta_a}{3}\right)\mathrm{d}g
\end{aligned}$$

where $f_{\hat{r}}$ is the density function of the measure $\mathbb{P}(\hat{r} \in \mathrm{d}x)$. Note that the following equality holds by the fundamental theorem of calculus

$$\frac{F(g)}{1 - F(g)} = \int_0^g \frac{h(u)}{1 - F(u)}\mathrm{d}u + \frac{F(0)}{1 - F(0)}$$

Therefore,

$$\begin{aligned}
&\frac{c}{\sqrt{k}} \int_0^\infty \frac{F(g)}{1 - F(g)} f_{\hat{r}}\left(r_{a^\star} - \frac{c}{\sqrt{k}}g - \frac{\Delta_a}{3}\right)\mathrm{d}g \\
&= \frac{c}{\sqrt{k}} \int_0^\infty \left(\int_0^g \frac{h(u)}{1 - F(u)}\mathrm{d}u + \frac{F(0)}{1 - F(0)}\right) f_{\hat{r}}\left(r_{a^\star} - \frac{c}{\sqrt{k}}g - \frac{\Delta_a}{3}\right)\mathrm{d}g \\
&= \frac{F(0)}{1 - F(0)}\mathbb{P}\left(\frac{\Delta_a}{3} \le r_{a^\star} - \hat{r}_{\tau_k, a^\star}\right) \qquad \text{(E.9)} \\
&\quad + \frac{c}{\sqrt{k}} \int_0^\infty \left(\int_0^g \frac{h(u)}{1 - F(u)}\mathrm{d}u\right) f_{\hat{r}}\left(r_{a^\star} - \frac{c}{\sqrt{k}}g - \frac{\Delta_a}{3}\right)\mathrm{d}g \qquad \text{(E.10)}
\end{aligned}$$

Since we assume sub-Gaussian noise ϵ_t and

$$r_{a^\star} - \hat{r}_{\tau_k, a^\star} = \frac{1}{k}\sum_{t=1}^k \epsilon_{\tau_t}, \tag{E.11}$$

we have

$$\mathbb{P}\left(\frac{\Delta_a}{3} \le r_{a^\star} - \hat{r}_{\tau_k, a^\star}\right) \le \exp\left(-\frac{k\Delta_a^2}{18\sigma^2}\right) \tag{E.12}$$

Hence we can get the upper bound of the first term in (E.10). Also, by Fubini-

Tonelli theorem, we can transform the second term in (E.10) as follows

$$\begin{aligned}
&\frac{c}{\sqrt{k}}\int_0^\infty\left(\int_0^g\frac{h(u)}{1-F(u)}\mathrm{d}u\right)f_{\hat{r}}\left(r_{a^\star}-\frac{c}{\sqrt{k}}g-\frac{\Delta_a}{3}\right)\mathrm{d}g\\
&=\int_0^\infty\left(\int_u^\infty f_{\hat{r}}\left(r_{a^\star}-\frac{c}{\sqrt{k}}g-\frac{\Delta_a}{3}\right)\frac{c}{\sqrt{k}}\mathrm{d}g\right)\frac{h(u)}{1-F(u)}\mathrm{d}u\\
&=\int_0^\infty\left(\int_{-\infty}^{r_{a^\star}-\frac{c}{\sqrt{k}}u-\frac{\Delta_a}{3}}f_{\hat{r}}\left(g\right)\mathrm{d}g\right)\frac{h(u)}{1-F(u)}\mathrm{d}u\\
&=\int_0^\infty\mathbb{P}\left(r_{a^\star}-\hat{r}_{\tau_k,a^\star}\geq\frac{c}{\sqrt{k}}u+\frac{\Delta_a}{3}\right)\frac{h(u)}{1-F(u)}\mathrm{d}u \qquad \text{(E.13)}
\end{aligned}$$

Similar to (E.12), we have

$$\mathbb{P}\left(r_{a^\star}-\hat{r}_{\tau_k,a^\star}\geq\frac{c}{\sqrt{k}}u+\frac{\Delta_a}{3}\right)\leq\exp\left(-\frac{\left(cu+\frac{\Delta_a}{3}\sqrt{k}\right)^2}{2\sigma^2}\right)$$

Thus, we obtain the following upper bound:

$$\begin{aligned}
&\int_0^\infty\mathbb{P}\left(r_{a^\star}-\hat{r}_{\tau_k,a^\star}\geq\frac{c}{\sqrt{k}}u+\frac{\Delta_a}{3}\right)\frac{h(u)}{1-F(u)}\mathrm{d}u\\
&\leq\int_0^\infty\exp\left(-\frac{\left(cu+\frac{\Delta_a}{3}\sqrt{k}\right)^2}{2\sigma^2}\right)\frac{h(u)}{1-F(u)}\mathrm{d}u\\
&\leq\exp\left(-\frac{k\Delta_a^2}{18\sigma^2}\right)\int_0^\infty\exp\left(-\frac{c^2u^2}{2\sigma^2}\right)\frac{h(u)}{1-F(u)}\mathrm{d}u\\
&\leq C_1\exp\left(-\frac{k\Delta_a^2}{18\sigma^2}\right)
\end{aligned}$$

Therefore,

$$\int_{I_1}\frac{F\left(\frac{\sqrt{k}}{c}\left\{r_{a^\star}-x-\frac{\Delta_a}{3}\right\}\right)}{1-F\left(\frac{\sqrt{k}}{c}\left\{r_{a^\star}-x-\frac{\Delta_a}{3}\right\}\right)}\mathbb{P}(\hat{r}\in\mathrm{d}x)\leq C_1\exp\left(-\frac{k\Delta_a^2}{18\sigma^2}\right)+\frac{F(0)}{1-F(0)}\exp\left(-\frac{k\Delta_a^2}{18\sigma^2}\right) \qquad \text{(E.14)}$$

Now we derive the upper bound of the integrand on $I_2=\{r_{a^\star}-\frac{\Delta_a}{3}<x\leq r_{a^\star}-\frac{\Delta_a}{6}\}$. Since $F(0)\leq 1/2$, it is easy to see that

$$\frac{F\left(\frac{\sqrt{k}}{c}\left\{r_{a^\star}-x-\frac{\Delta_a}{3}\right\}\right)}{1-F\left(\frac{\sqrt{k}}{c}\left\{r_{a^\star}-x-\frac{\Delta_a}{3}\right\}\right)}\leq 2F\left(\frac{\sqrt{k}}{c}\left\{r_{a^\star}-x-\frac{\Delta_a}{3}\right\}\right) \qquad \text{(E.15)}$$

for $x \in I_2 \cup I_3$. Hence

$$\begin{aligned}
&\int_{I_2} \frac{F\left(\frac{\sqrt{k}}{c}\left\{r_{a^\star} - x - \frac{\Delta_a}{3}\right\}\right)}{1 - F\left(\frac{\sqrt{k}}{c}\left\{r_{a^\star} - x - \frac{\Delta_a}{3}\right\}\right)} \mathbb{P}(\hat{r} \in \mathrm{d}x) \\
&\quad = \int_{r_{a^\star} - \frac{\Delta_a}{3}}^{r_{a^\star} - \frac{\Delta_a}{6}} \frac{F\left(\frac{\sqrt{k}}{c}\left\{r_{a^\star} - x - \frac{\Delta_a}{3}\right\}\right)}{1 - F\left(\frac{\sqrt{k}}{c}\left\{r_{a^\star} - x - \frac{\Delta_a}{3}\right\}\right)} \mathbb{P}(\hat{r} \in \mathrm{d}x) \\
&\quad \le \int_{r_{a^\star} - \frac{\Delta_a}{3}}^{r_{a^\star} - \frac{\Delta_a}{6}} 2F\left(\frac{\sqrt{k}}{c}\left\{r_{a^\star} - x - \frac{\Delta_a}{3}\right\}\right) \mathbb{P}(\hat{r} \in \mathrm{d}x) \\
&\quad \le 2\mathbb{P}\left(\frac{\Delta_a}{6} \le r_{a^\star} - \hat{r}_{\tau_k, a^\star}\right)
\end{aligned}$$

Similar to (E.12), we have

$$2\mathbb{P}\left(\frac{\Delta_a}{6} \le r_{a^\star} - \hat{r}_{\tau_k, a^\star}\right) \le 2\exp\left(-\frac{k\Delta_a^2}{72\sigma^2}\right) \tag{E.16}$$

Hence we get the upper bound of the integral on I_2.

Finally, due to (E.15) again,

$$\begin{aligned}
&\int_{I_3} \frac{F\left(\frac{\sqrt{k}}{c}\left\{r_{a^\star} - x - \frac{\Delta_a}{3}\right\}\right)}{1 - F\left(\frac{\sqrt{k}}{c}\left\{r_{a^\star} - x - \frac{\Delta_a}{3}\right\}\right)} \mathbb{P}(\hat{r} \in \mathrm{d}x) && \text{(E.17)} \\
&\quad = \int_{r_{a^\star} - \frac{\Delta_a}{6}}^{\infty} \frac{F\left(\frac{\sqrt{k}}{c}\left\{r_{a^\star} - x - \frac{\Delta_a}{3}\right\}\right)}{1 - F\left(\frac{\sqrt{k}}{c}\left\{r_{a^\star} - x - \frac{\Delta_a}{3}\right\}\right)} \mathbb{P}(\hat{r} \in \mathrm{d}x) \\
&\quad \le 2\int_{r_{a^\star} - \frac{\Delta_a}{6}}^{\infty} F\left(\frac{\sqrt{k}}{c}\left\{r_{a^\star} - x - \frac{\Delta_a}{3}\right\}\right) \mathbb{P}(\hat{r} \in \mathrm{d}x) \\
&\quad \le 2F\left(-\frac{\sqrt{k}\Delta_a}{6c}\right) && \text{(E.18)}
\end{aligned}$$

By combining (E.14), (E.16), and (E.18),

$$\begin{aligned}\sum_{k=1}^{T}\mathbb{E}\left[\frac{\mathfrak{F}_{\tau_k,a^\star}}{1-\mathfrak{F}_{\tau_k,a^\star}}\Big|\mathcal{H}_{\tau_k}\right] &\leq \sum_{k=1}^{T}\left\{C_1\exp\left(-\frac{k\Delta_a^2}{18\sigma^2}\right)+\frac{F(0)}{1-F(0)}\exp\left(-\frac{k\Delta_a^2}{18\sigma^2}\right)\right\}\\ &\quad+\sum_{k=1}^{T}2\exp\left(-\frac{k\Delta_a^2}{72\sigma^2}\right)+\sum_{k=1}^{T}2F\left(-\frac{\sqrt{k}\Delta_a}{6c}\right)\\ &\leq\left\{C_1+\frac{F(0)}{1-F(0)}\right\}\frac{18\sigma^2}{\Delta_a^2}+\frac{144\sigma^2}{\Delta_a^2}+\sum_{k=1}^{T}2F\left(-\frac{\sqrt{k}\Delta_a}{6c}\right)\end{aligned}$$

Therefore, thanks to (E.7), we obtained the desired result. The lemma is proved.

□

Proof of Lemma 9. Recall τ_k from Lemma (26). Obviously,

$$\sum_{t=1}^{T}\mathbb{P}\left(E_{t,a}^{(3)}\right)\leq\sum_{k=1}^{T}\mathbb{P}\left(\hat{E}_{\tau_k,a}\cap\tilde{E}_{\tau_k,a}^c\right)$$

Due to the definition of τ_k and $\beta_{t,a}$, observe that $n_{\tau_k,a}=k$ and $\beta_{\tau_k,a}=\frac{c}{\sqrt{k}}$. Hence by the conditioning on $\mathcal{H}_{\tau_k}$,

$$\begin{aligned}\mathbb{P}\left(\hat{E}_{\tau_k,a}\cap\tilde{E}_{\tau_k,a}^c\Big|\mathcal{H}_{\tau_k}\right)&\leq\mathbb{P}\left(\hat{r}_{\tau_k}\leq x_a, G_{\tau_k,a}>\frac{y_a-\hat{r}_{\tau_k,a}}{\beta_{\tau_k,a}}\Big|\mathcal{H}_{\tau_k}\right)\\ &\leq\mathbb{P}\left(G_{\tau_k,a}>\frac{y_a-x_a}{\beta_{\tau_k,a}}\Big|\mathcal{H}_{\tau_k}\right)\\ &=\mathbb{P}\left(G_{\tau_k,a}>\frac{\Delta_a\sqrt{k}}{3c}\Big|\mathcal{H}_{\tau_k}\right)=1-F\left(\frac{\Delta_a\sqrt{k}}{3c}\right)\end{aligned}\tag{E.19}$$

Let ℓ be the maximal time such as

$$F\left(\frac{\Delta_a\sqrt{\ell}}{3c}\right)\leq 1-\frac{c^2}{T\Delta_a^2}$$

Note that

$$\ell\leq\frac{9c^2}{\Delta_a^2}\left\{F^{-1}\left(1-\frac{c^2}{T\Delta_a^2}\right)\right\}^2\tag{E.20}$$

and for $k>\ell$

$$1-F\left(\frac{\Delta_a\sqrt{k}}{3c}\right)\leq\frac{c^2}{T\Delta_a^2}\tag{E.21}$$

Therefore, by (E.19), (E.20), and (E.21),

$$\begin{aligned}\sum_{k=1}^{T}\mathbb{P}\left(\hat{E}_{\tau_k,a}\cap\tilde{E}^c_{\tau_k,a}\right)&\leq\sum_{k=1}^{T}\left(1-F\left(\frac{\Delta_a\sqrt{k}}{3c}\right)\right)\\&\leq\ell+\sum_{k=\ell+1}^{T}\left(1-F\left(\frac{\Delta_a\sqrt{k}}{3c}\right)\right)\\&\leq\frac{9c^2}{\Delta_a^2}\left\{F^{-1}\left(1-\frac{c^2}{T\Delta_a^2}\right)\right\}^2+\sum_{k=\ell+1}^{T}\frac{c^2}{T\Delta_a^2}\\&\leq\frac{9c^2}{\Delta_a^2}\left\{F^{-1}\left(1-\frac{c^2}{T\Delta_a^2}\right)\right\}^2+\frac{c^2}{\Delta_a^2}\end{aligned}$$

The lemma is proved. □

Proof of Theorem 27. Recall the definition of regret $\mathcal{R}_T$, and the fact $\mathbb{P}(a_t = a) = \mathbb{P}(E_{t,a}) = \sum_{i=1}^{3}\mathbb{P}(E^{(i)}_{t,a})$. Hence

$$\mathbb{E}[\mathcal{R}_T] := \sum_{a\in\mathcal{A}}\sum_{t=1}^{T}\Delta_a\mathbb{P}\left(a_t=a\right)=\sum_{a\neq a^\star}\sum_{i=1}^{3}\sum_{t=1}^{T}\Delta_a\mathbb{P}\left(E^{(i)}_{t,a}\right)\tag{E.22}$$

By Lemmas 7, 8, and 9,

$$\sum_{t=1}^{T}\Delta_a\mathbb{P}\left(E^{(1)}_{t,a}\right)\leq\Delta_a+\frac{18\sigma^2}{\Delta_a}$$

$$\sum_{t=1}^{T}\Delta_a\mathbb{P}\left(E^{(2)}_{t,a}\right)\leq\left(C_1+\frac{F(0)}{1-F(0)}\right)\frac{18\sigma^2}{\Delta_a}+\frac{144\sigma^2}{\Delta_a}+2\Delta_a\sum_{k=1}^{T}F\left(-\frac{\Delta_a\sqrt{k}}{6c}\right)$$

$$\sum_{t=1}^{T}\Delta_a\mathbb{P}\left(E^{(3)}_{t,a}\right)\leq\frac{9c^2}{\Delta_a}\left\{F^{-1}\left(1-\frac{c^2}{T\Delta_a^2}\right)\right\}^2+\frac{c^2}{\Delta_a}$$

Therefore, we can estimate the upper bound of (E.22) by combining the above results as follows

$$\begin{aligned}\mathbb{E}[\mathcal{R}_T]\leq\sum_{a\neq a^\star}\Bigg[&\left(C_1+\frac{F(0)}{1-F(0)}\right)\frac{18\sigma^2}{\Delta_a}+2\Delta_a\sum_{k=1}^{T}F\left(-\frac{\Delta_a\sqrt{k}}{6c}\right)\\&+\frac{9c^2}{\Delta_a}\left\{F^{-1}\left(1-\frac{c^2}{T\Delta_a^2}\right)\right\}^2+\frac{162\sigma^2+c^2}{\Delta_a}+\Delta_a\Bigg]\end{aligned}$$

The theorem is proved. □

E.3 Proofs of Corollaries

Proof of Corollary 1. The cumulative density function of a Weibull distribution is given as

$$F(x) = 1 - \exp\left(-\left(\frac{x}{\lambda}\right)^k\right)$$

Then, its inverse is

$$F^{-1}(y) = \lambda\left[\ln\left(\frac{1}{1-y}\right)\right]^{\frac{1}{k}},$$

Then,

$$\left[F^{-1}\left(1 - \frac{c^2}{T\Delta_a^2}\right)\right]^2 = \lambda\left[\ln\left(\frac{T\Delta_a^2}{c^2}\right)\right]^{\frac{2}{k}}.$$

Unfortunately, $h(x)$ of Weibull distribution with $1 < k \leq 2$ is not bounded. Thus, we compute C_1 instead of M_1 as follows,

$$\begin{aligned}\int_0^\infty \frac{h(z)\exp\left(-\frac{c^2z^2}{2\sigma^2}\right)}{1-F(z)}dz &= \int_0^\infty \frac{k}{\lambda}\left(\frac{z}{\lambda}\right)^{k-1}\frac{\exp\left(-\left(\frac{z}{\lambda}\right)^k\right)\exp\left(-\frac{c^2z^2}{2\sigma^2}\right)}{\exp\left(-2\left(\frac{z}{\lambda}\right)^k\right)}dz \\ &\leq \int_0^\infty \frac{k}{\lambda}\left(\frac{z}{\lambda}\right)^{k-1}\exp\left(-\frac{c^2z^2}{2\sigma^2} + \left(\frac{z}{\lambda}\right)^k\right)dz =: C_1 \\ &\because\ C_1 \text{ exists when } k < 2 \text{ or } k = 2 \text{ and } c > \sqrt{\frac{2\sigma^2}{\lambda^2}}.\end{aligned}$$

where C_1 is the same as in [64]. For $\sum_{k=1}^T \Delta_a F\left(-\frac{\Delta_a\sqrt{k}}{6c}\right)$, we have,

$$\sum_{k=1}^T \Delta_a F\left(-\frac{\Delta_a\sqrt{k}}{6c}\right) = 0$$

since the support of x is $(0, \infty)$. Then, the problem dependent regret bound becomes,

$$\mathbb{E}\left[\mathcal{R}_T\right] \leq \sum_{a \neq a^\star} \left[C_1 + \frac{F\left(0\right)}{1 - F\left(0\right)}\right] \frac{18\sigma^2}{\Delta_a} + 2\sum_{k=1}^{T} \Delta_a F\left(-\frac{\Delta_a \sqrt{k}}{6c}\right) \tag{E.23}$$

$$+ \frac{9c^2}{\Delta_a}\left[F^{-1}\left(1 - \frac{c^2}{T\Delta_a^2}\right)\right]^2 + \frac{c^2 + 162\sigma^2}{\Delta_a} + \Delta_a \tag{E.24}$$

$$\leq \sum_{a \neq a^\star} C_1 \frac{18\sigma^2}{\Delta_a} + \frac{9c^2}{\Delta_a}\left[\ln\left(\frac{T\Delta_a^2}{c^2}\right)\right]^{\frac{2}{k}} + \frac{c^2 + 162\sigma^2}{\Delta_a} + \Delta_a \tag{E.25}$$

$$\leq C\left(\sum_{a \neq a^\star} \Delta_a + \frac{9c^2}{\Delta_a}\left[\ln\left(\frac{T\Delta_a^2}{c^2}\right)\right]^{\frac{2}{k}}\right). \tag{E.26}$$

The problem independent regret bound can be obtained by choosing the threshold of the minimum gap as $\Delta = c\sqrt{K/T}\ln(K)^{1/k}$,

$$\mathbb{E}\left[\mathcal{R}_T\right] \leq \sum_{a \neq a^\star} C_1 \frac{18\sigma^2}{\Delta_a} + \frac{9c^2}{\Delta_a}\left[\ln\left(\frac{T\Delta_a^2}{c^2}\right)\right]^{\frac{2}{k}} + \frac{c^2 + 162\sigma^2}{\Delta_a} + \Delta_a \tag{E.27}$$

$$\leq K\left[C_1 \frac{18\sigma^2}{\Delta} + \frac{9c^2}{\Delta}\left[\ln\left(\frac{T\Delta^2}{c^2}\right)\right]^{\frac{2}{k}} + \frac{c^2 + 162\sigma^2}{\Delta}\right] + \Delta T \tag{E.28}$$

$$\leq \sqrt{KT}\left[\frac{1}{c\ln(K)^{1/k}} C_1 18\sigma^2 + \frac{9c^2}{c\ln(K)^{1/k}}\left[\ln\left(K\ln(K)^{2/k}\right)\right]^{2/k}\right. \tag{E.29}$$

$$\left. + \frac{c^2 + 162\sigma^2}{c\ln(K)^{1/k}} + c\ln(K)^{1/k}\right] \tag{E.30}$$

$$\leq \sqrt{KT}\left[\frac{1}{c\ln(K)^{1/k}} C_1 18\sigma^2 + \frac{9c^2}{c\ln(K)^{1/k}}\left[(1 + 1/k)\ln\left(K\right)\right]^{2/k}\right. \tag{E.31}$$

$$\left. + \frac{c^2 + 162\sigma^2}{c\ln(K)^{1/k}} + c\ln(K)^{1/k}\right] \tag{E.32}$$

$$\because \text{See Appendix 5 in [29]} \tag{E.33}$$

$$\leq O\left(\sqrt{KT}\frac{\left[(1 + 1/k)\ln\left(K\right)\right]^{2/k}}{\ln(K)^{1/k}}\right) \tag{E.34}$$

$$\leq O\left(\sqrt{KT}\ln(K)^{1/k}\right). \tag{E.35}$$

The lower bound is simply obtained by plugging F^{-1} into the general lower bound, so we can conclude that regret bound is tight. The corollary is proved. □

Proof of Corollary 2. The upper and lower bound of cumulative density function of a Gaussian distribution with $\sigma_g > 0$ are given as

$$1 - \frac{1}{2}\exp\left(-\frac{x^2}{2\sigma_g^2}\right) \leq F(x) \leq 1 - \sqrt{\frac{2\sigma_g^2}{\pi}}\frac{\exp\left(-\frac{x^2}{2\sigma_g^2}\right)}{x + \sqrt{x^2 + 4\sigma_g^2}}$$

where the upper bound holds for $x > 0$. Then, its inverse is bounded by using the lower bound of F as follows,

$$F^{-1}(y) \leq \sqrt{2}\sigma_g\sqrt{\ln\left(\frac{1}{2(1-y)}\right)}.$$

Then,

$$\left[F^{-1}\left(1 - \frac{c^2}{T\Delta_a^2}\right)\right]^2 \leq \sqrt{2}\sigma_g \ln\left(\frac{T\Delta_a^2}{c^2}\right) - \sqrt{2}\sigma_g\ln(2) \leq \sqrt{2}\sigma_g\ln\left(\frac{T\Delta_a^2}{c^2}\right).$$

Unfortunately, $h(x)$ of Gaussian distribution is not bounded. Thus, similarly to a Weibull distribution, we compute C_1 instead of M_1 as follows,

$$\begin{aligned}
&\int_0^\infty \frac{h(z)\exp\left(-\frac{c^2z^2}{2\sigma^2}\right)}{1 - F(z)}dz \\
&= \int_0^\infty \frac{1}{\sqrt{2\pi\sigma_g^2}}\frac{\left(z + \sqrt{z^2 + 4\sigma_g^2}\right)^2\exp\left(-\frac{z^2}{2\sigma_g^2}\right)\exp\left(-\frac{c^2z^2}{2\sigma^2}\right)}{\frac{2\sigma_g^2}{\pi}\exp\left(-\frac{z^2}{\sigma_g^2}\right)}dz \\
&\leq \int_0^\infty \frac{\sqrt{\pi}}{(2\pi\sigma_g^2)^{3/2}}\left(z + \sqrt{z^2 + 4\sigma_g^2}\right)^2\exp\left(-\frac{c^2z^2}{2\sigma^2} + \frac{z^2}{2\sigma_g^2}\right)dz =: C_1 \\
&\because\ C_1 \text{ exists since } c > \sqrt{\frac{\sigma^2}{\sigma_g^2}}.
\end{aligned}$$

For $\sum_{k=1}^T \Delta_a F\left(-\frac{\Delta_a\sqrt{k}}{6c}\right)$, we use the following relation of CDF of a Gaussian distribution, for $x > 0$,

$$\frac{1}{\sqrt{2\pi\sigma_g^2}}\int_{-\infty}^{-x}\exp\left(-\frac{z^2}{2\sigma_g^2}\right)dz = \frac{1}{\sqrt{2\pi\sigma_g^2}}\int_x^\infty \exp\left(-\frac{z^2}{2\sigma_g^2}\right)dz \leq \exp\left(-\frac{x^2}{2\sigma_g^2}\right)$$

we have,

$$\sum_{k=1}^{T} \Delta_a F\left(-\frac{\Delta_a\sqrt{k}}{6c}\right) \leq \sum_{k=1}^{T} \Delta_a \exp\left(-\frac{\Delta_a^2 k}{72\sigma_g^2 c^2}\right) \leq \frac{72\sigma_g^2 c^2}{\Delta_a}$$

Then, the problem dependent regret bound becomes,

$$\mathbb{E}\left[\mathcal{R}_T\right] \leq \sum_{a\neq a^\star} \left[C_1 + \frac{F(0)}{1-F(0)}\right]\frac{18\sigma^2}{\Delta_a} + 2\sum_{k=1}^{T} \Delta_a F\left(-\frac{\Delta_a\sqrt{k}}{6c}\right) \tag{E.36}$$

$$+\frac{9c^2}{\Delta_a}\left[F^{-1}\left(1-\frac{c^2}{T\Delta_a^2}\right)\right]^2 + \frac{c^2+162\sigma^2}{\Delta_a} + \Delta_a \tag{E.37}$$

$$\leq \sum_{a\neq a^\star} C_1\frac{18\sigma^2}{\Delta_a} + \frac{72\sigma_g^2c^2}{\Delta_a} + \frac{18c^2}{\Delta_a}\left[\ln\left(\frac{T\Delta_a^2}{c^2}\right)\right] + \frac{c^2+162\sigma^2}{\Delta_a} + \Delta_a \tag{E.38}$$

$$\leq C\left(\sum_{a\neq a^\star} \Delta_a + \frac{18c^2}{\Delta_a}\left[\ln\left(\frac{T\Delta_a^2}{c^2}\right)\right]\right). \tag{E.39}$$

The problem independent regret bound can be obtained by choosing the threshold of the minimum gap as $\Delta = c\sqrt{K/T\ln(K)}$,

$$\mathbb{E}\left[\mathcal{R}_T\right] \leq \sum_{a\neq a^\star} C_1\frac{18\sigma^2}{\Delta_a} + \frac{18c^2}{\Delta_a}\left[\ln\left(\frac{T\Delta_a^2}{c^2}\right)\right] + \frac{(72\sigma_g^2+1)c^2+162\sigma^2}{\Delta_a} + \Delta_a \tag{E.40}$$

$$\leq K\left[C_1\frac{18\sigma^2}{\Delta} + \frac{18c^2}{\Delta}\left[\ln\left(\frac{T\Delta^2}{c^2}\right)\right] + \frac{(72\sigma_g^2+1)c^2+162\sigma^2}{\Delta}\right] + \Delta T \tag{E.41}$$

$$\leq \sqrt{KT}\left[\frac{1}{c\sqrt{\ln(K)}}C_1 18\sigma^2 + \frac{18c^2}{c\sqrt{\ln(K)}}\left[\ln\left(K\ln(K)\right)\right]\right. \tag{E.42}$$

$$\left.+\frac{(72\sigma_g^2+1)c^2+162\sigma^2}{c\sqrt{\ln(K)}} + c\sqrt{\ln(K)}\right] \tag{E.43}$$

$$\leq \sqrt{KT}\left[\frac{1}{c\sqrt{\ln(K)}}C_1 18\sigma^2 + \frac{18c^2}{c\sqrt{\ln(K)}}\frac{3}{2}\ln(K)\right. \tag{E.44}$$

$$\left.+\frac{(72\sigma_g^2+1)c^2+162\sigma^2}{c\sqrt{\ln(K)}} + c\sqrt{\ln(K)}\right] \tag{E.45}$$

$$\because \text{See Appendix 5 in [29]} \tag{E.46}$$

$$\leq O\left(\sqrt{KT\ln(K)}\right). \tag{E.47}$$

For the lower bound, let us define a constant $c_K = F^{-1}\left(1 - \frac{1}{K}\right)$. Note that $c_K > 0$ since $1 - \frac{1}{K} \geq \frac{1}{2}$. Then, we can apply the upper bound of $F(x)$ as follows,

$$1 - \frac{1}{K} < 1 - \sqrt{\frac{2\sigma_g^2}{\pi}} \frac{\exp\left(-\frac{c_K^2}{2\sigma_g^2}\right)}{c_K + \sqrt{c_K^2 + 4\sigma_g^2}} \tag{E.48}$$

$$-\frac{c_K^2}{2\sigma_g^2} + \ln\left(\sqrt{\frac{2\sigma_g^2}{\pi}}\right) - \ln\left(c_K + \sqrt{c_K^2 + 4\sigma_g^2}\right) < -\ln(K) \tag{E.49}$$

$$\sigma_g\sqrt{2\ln(K) + \ln\left(\sqrt{\frac{2\sigma_g^2}{\pi}}\right) - \ln\left(c_K + \sqrt{c_K^2 + 4\sigma_g^2}\right)} < c_K \tag{E.50}$$

$$\Omega\left(\sqrt{\ln(K)}\right) < c_K \tag{E.51}$$

Consequently, the lower bound is simply obtained by the general lower bound, so we can conclude that regret bound is tight. The corollary is proved. □

Proof of Corollary 3. The CDF of a Pareto distribution is given as

$$F(x) = 1 - \frac{1}{(x/\lambda + 1)^{\alpha}}$$

Then, its inverse is

$$F^{-1}(y) = \lambda(1-y)^{-\frac{1}{\alpha}} - \lambda,$$

Then,

$$\left[F^{-1}\left(1 - \frac{c^2}{T\Delta_a^2}\right)\right]^2 = \lambda^2\left[\left(\frac{T\Delta_a^2}{c^2}\right)^{\frac{1}{\alpha}} - 1\right]^2.$$

In [5], the $\sup h$ can be obtained as follows,

$$\sup h = \frac{\alpha}{\lambda}.$$

M_1 can be obtained as,

$$\begin{aligned}\int_0^\infty \frac{\exp\left(-\frac{c^2z^2}{2\sigma^2}\right)}{1-F(z)}dz &= \int_0^\infty \left(\frac{z}{\lambda}+1\right)^\alpha \exp\left(-\frac{c^2z^2}{2\sigma^2}\right)dz\\ &\leq \int_0^\infty \exp\left(-\frac{c^2z^2}{2\sigma^2}+\alpha\ln\left(\frac{z}{\lambda}+1\right)\right)dz\\ &\leq \int_0^\infty \exp\left(-\frac{c^2z^2}{2\sigma^2}+\alpha\frac{z}{\lambda}\right)dz \ \because\ \ln(x+1)\leq x\\ &\leq \sqrt{\frac{2\pi\sigma^2}{c^2}}\exp\left(\frac{\sigma^2\alpha^2}{2c^2\lambda^2}\right)\\ &\leq \sqrt{\frac{2\pi\sigma^2}{c^2}}\exp\left(\frac{\sigma^2}{2c^2}\right) := M_1,\\ &\quad \because\ \alpha\leq\lambda.\end{aligned}$$

For $\sum_{k=1}^T \Delta_a F\left(-\frac{\Delta_a\sqrt{k}}{6c}\right)$, we have,

$$\sum_{k=1}^T \Delta_a F\left(-\frac{\Delta_a\sqrt{k}}{6c}\right)=0$$

since the support of x is $(0,\infty)$. Then, the problem dependent regret bound becomes,

$$\begin{aligned}\mathbb{E}\left[\mathcal{R}_T\right] \leq \sum_{a\neq a^\star}&\left[\|h\|_\infty M_1+\frac{F(0)}{1-F(0)}\right]\frac{18\sigma^2}{\Delta_a}+2\sum_{k=1}^T\Delta_a F\left(-\frac{\Delta_a\sqrt{k}}{6c}\right) &\text{(E.52)}\\ &+\frac{9c^2}{\Delta_a}\left[F^{-1}\left(1-\frac{c^2}{T\Delta_a^2}\right)\right]^2+\frac{c^2+162\sigma^2}{\Delta_a}+\Delta_a &\text{(E.53)}\\ \leq \sum_{a\neq a^\star}&\frac{\alpha}{\lambda}M_1\frac{18\sigma^2}{\Delta_a}+\frac{9c^2\lambda^2}{\Delta_a}\left[\left(\frac{T\Delta_a^2}{c^2}\right)^{\frac{1}{\alpha}}-1\right]^2+\frac{c^2+162\sigma^2}{\Delta_a}+\Delta_a &\text{(E.54)}\\ \leq C&\left(\sum_{a\neq a^\star}\Delta_a+\frac{9c^2\lambda^2}{\Delta_a}\left[\left(\frac{T\Delta_a^2}{c^2}\right)^{\frac{1}{\alpha}}-1\right]^2\right). &\text{(E.55)}\end{aligned}$$

The problem independent regret bound can be obtained by choosing the threshold of the minimum gap as $\Delta = c\sqrt{K/T}\sqrt{\frac{\alpha}{K^{2/\alpha}}}$ and $\lambda = \sqrt{\alpha}$.

$$\mathbb{E}\left[\mathcal{R}_T\right] \leq \sum_{\Delta_a \geq \Delta} \frac{\alpha}{\lambda} M_1 \frac{18\sigma^2}{\Delta_a} + \frac{9c^2\lambda^2}{\Delta_a}\left[\left(\frac{T\Delta_a^2}{c^2}\right)^{\frac{1}{\alpha}} - 1\right]^2 + \frac{c^2 + 162\sigma^2}{\Delta_a} + \Delta T \tag{E.56}$$

$$\leq K\left[\frac{\alpha}{\lambda} M_1 \frac{18\sigma^2}{\Delta} + \frac{9c^2\lambda^2}{\Delta}\left[\left(\frac{T\Delta^2}{c^2}\right)^{\frac{1}{\alpha}} - 1\right]^2 + \frac{c^2 + 162\sigma^2}{\Delta}\right] + \Delta T \tag{E.57}$$

$$\leq \sqrt{KT}\left[\frac{\alpha}{\lambda}\sqrt{\frac{K^{2/\alpha}}{\alpha c^2}} M_1 18\sigma^2 + 9c\lambda^2\sqrt{\frac{K^{2/\alpha}}{\alpha}}\left[\left(K\frac{\alpha}{K^{2/\alpha}}\right)^{\frac{1}{\alpha}} - 1\right]^2\right. \tag{E.58}$$

$$\left. + \frac{c^2 + 162\sigma^2}{c}\sqrt{\frac{K^{2/\alpha}}{\alpha}} + c\sqrt{\frac{\alpha}{K^{2/\alpha}}}\right] \tag{E.59}$$

$$\leq \sqrt{KT}\left[\frac{\alpha}{\lambda}\sqrt{\frac{K^{2/\alpha}}{\alpha c^2}} M_1 18\sigma^2 + 9c\lambda^2 K^{2/\alpha}\left(\frac{\alpha}{K^{2/\alpha}}\right)^{\frac{2}{\alpha} - \frac{1}{2}}\right. \tag{E.60}$$

$$\left. + \frac{c^2 + 162\sigma^2}{c}\sqrt{\frac{K^{2/\alpha}}{\alpha}} + c\sqrt{\frac{\alpha}{K^{2/\alpha}}}\right] \tag{E.61}$$

$$\leq \sqrt{KT}\left[\sqrt{\frac{K^{2/\alpha}}{c^2}} M_1 18\sigma^2 + 9c\alpha^{\frac{1}{2} + \frac{2}{\alpha}} K^{3/\alpha}\right. \tag{E.62}$$

$$\left. + \frac{c^2 + 162\sigma^2}{c}\sqrt{\frac{K^{2/\alpha}}{\alpha}} + c\sqrt{\frac{\alpha}{K^{2/\alpha}}}\right] \tag{E.63}$$

$$\because\ \lambda = \sqrt{\alpha} \text{ and } \frac{3}{2} - \frac{2}{\alpha} \leq \frac{3}{2} \tag{E.64}$$

$$\leq O\left(\sqrt{\alpha^{1 + \frac{4}{\alpha}} K^{1 + 6/\alpha} T}\right). \tag{E.65}$$

For the optimal rate, we set $\alpha = \ln(K)$, then,

$$O\left(\sqrt{\ln(K)^{1 + \frac{4}{\ln(K)}} K^{1 + 6/\ln(K)} T}\right) \leq O\left(\sqrt{KT\ln(K)}\right)$$

where $\ln(K)^{1 + \frac{4}{\ln(K)}} \leq e^{4/e}\ln(K)$ and $K^{6/\ln(K)} = e^6$. The lower bound is simply obtained by plugging F^{-1} into the general lower bound. The corollary is proved.

□

Proof of Corollary 4. The CDF of a Fréchet distribution with $\alpha > 0$ is given as

$$F(x) = \exp\left(-\left(\frac{x}{\lambda}\right)^{-\alpha}\right)$$

Then, its inverse is

$$F^{-1}(y) = \lambda \ln(1/y)^{-1/\alpha} \leq \lambda \left(1 - y\right)^{-1/\alpha},$$

where $\ln(x) \leq x - 1$ is used. Then,

$$\left[F^{-1}\left(1 - \frac{c^2}{T\Delta_a^2}\right)\right]^2 \leq \lambda^2 \left[\frac{T\Delta_a^2}{c^2}\right]^{2/\alpha}.$$

Using the same technique in [5], we have $\sup h \leq 2\alpha/\lambda$ and M_1 can be obtained,

$$\begin{aligned}
\int_0^\infty \frac{\exp\left(-\frac{c^2z^2}{2\sigma^2}\right)}{\left(1 - \exp\left(-\left(\frac{z}{\lambda}\right)^{-\alpha}\right)\right)} dz \leq & \int_0^\infty \left(1 + \left(\frac{z}{\lambda}\right)^\alpha\right) \exp\left(-\frac{c^2z^2}{2\sigma^2}\right) dz \\
& \because\ 1/(1 - \exp(-x^{-1})) \leq 1 + x \\
=& \sqrt{\frac{\pi\sigma^2}{2c^2}} + \int_0^\infty \left(\frac{z}{\lambda}\right)^\alpha \exp\left(-\frac{c^2z^2}{2\sigma^2}\right) dz \\
\leq& \sqrt{\frac{\pi\sigma^2}{2c^2}} + \int_0^\infty \exp\left(-\frac{c^2z^2}{2\sigma^2} + \alpha\left(\ln(\left(\frac{z}{\lambda}\right))\right)\right) dz \\
\leq& \sqrt{\frac{\pi\sigma^2}{2c^2}} + \int_0^\infty \exp\left(-\frac{c^2z^2}{2\sigma^2} + \alpha\left(\left(\frac{z}{\lambda}\right) - 1\right)\right) dz \\
\leq& \sqrt{\frac{\pi\sigma^2}{2c^2}} + \sqrt{\frac{2\pi\sigma^2}{c^2}} \exp\left(\frac{\alpha^2\sigma^2}{2c^2\lambda^2} - \alpha\right) \ \because\ \frac{\sigma^2\alpha}{2c^2} \leq \lambda^2 \\
\leq& \frac{3}{2}\sqrt{\frac{2\pi\sigma^2}{c^2}} =: M_1
\end{aligned}$$

Unlikely other results, for Fréchet distribution, M_1 depends on a parameter of distribution α. For $\sum_{k=1}^T \Delta_a F\left(-\frac{\Delta_a\sqrt{k}}{6c}\right)$, the summation is zero,

$$\sum_{k=1}^T \Delta_a F\left(-\frac{\Delta_a\sqrt{k}}{6c}\right) = 0,$$

since its support is $(0, \infty)$. Then, the problem dependent regret bound becomes,

$$\mathbb{E}\left[\mathcal{R}_T\right] \leq \sum_{a \neq a^\star} \left[\|h\|_\infty M_1 + \frac{F(0)}{1 - F(0)}\right] \frac{18\sigma^2}{\Delta_a} + 2\sum_{k=1}^{T} \Delta_a F\left(-\frac{\Delta_a \sqrt{k}}{6c}\right) \tag{E.66}$$

$$+ \frac{9c^2}{\Delta_a}\left[F^{-1}\left(1 - \frac{c^2}{T\Delta_a^2}\right)\right]^2 + \frac{c^2 + 162\sigma^2}{\Delta_a} + \Delta_a \tag{E.67}$$

$$\leq \sum_{a \neq a^\star} 3\frac{\alpha}{\lambda}\sqrt{\frac{2\pi\sigma^2}{c^2}}\frac{18\sigma^2}{\Delta_a} + \frac{9c^2\lambda^2}{\Delta_a}\left[\frac{T\Delta_a^2}{c^2}\right]^{2/\alpha} + \frac{c^2 + 162\sigma^2}{\Delta_a} + \Delta_a \tag{E.68}$$

$$\leq C\left(\sum_{a \neq a^\star} \Delta_a + \frac{9c^2\lambda^2}{\Delta_a}\left[\frac{T\Delta_a^2}{c^2}\right]^{2/\alpha}\right). \tag{E.69}$$

The problem independent regret bound can be obtained by choosing the threshold of the minimum gap as $\Delta = c\sqrt{K/T}\sqrt{\frac{\alpha}{K^{2/\alpha}}}$.

$$\mathbb{E}\left[\mathcal{R}_T\right] \leq \sum_{a \neq a^\star} 3\frac{\alpha}{\lambda}\sqrt{\frac{2\pi\sigma^2}{c^2}}\frac{18\sigma^2}{\Delta_a} + \frac{9c^2\lambda^2}{\Delta_a}\left[\frac{T\Delta_a^2}{c^2}\right]^{2/\alpha} + \frac{c^2 + 162\sigma^2}{\Delta_a} + \Delta_a \tag{E.70}$$

$$\leq K\left[\sum_{a \neq a^\star} 3\frac{\alpha}{\lambda}\sqrt{\frac{2\pi\sigma^2}{c^2}}\frac{18\sigma^2}{\Delta} + \frac{9c^2\lambda^2}{\Delta}\left[\frac{T\Delta^2}{c^2}\right]^{2/\alpha} + \frac{c^2 + 162\sigma^2}{\Delta}\right] + \Delta T \tag{E.71}$$

$$\leq \sqrt{KT}\left[\frac{3}{\lambda}\sqrt{\frac{2\pi\sigma^2}{c^2}}\frac{18\sigma^2}{c}\sqrt{\alpha K^{2/\alpha}} + 9c\lambda^2\left[K\right]^{2/\alpha}\left[\frac{\alpha}{K^{2/\alpha}}\right]^{\frac{2}{\alpha} - \frac{1}{2}}\right. \tag{E.72}$$

$$\left. + \frac{c^2 + 162\sigma^2}{c}\sqrt{\frac{K^{2/\alpha}}{\alpha}} + c\sqrt{\frac{\alpha}{K^{2/\alpha}}}\right] \tag{E.73}$$

$$\leq \sqrt{KT}\left[6\sqrt{\pi}\frac{18\sigma^2}{c}\sqrt{\alpha K^{2/\alpha}} + \frac{9\sigma^2\alpha^2}{2c}\left(\frac{K^{2/\alpha}}{\alpha}\right)^{\frac{3}{2} - \frac{2}{\alpha}}\right. \tag{E.74}$$

$$\left. + \frac{c^2 + 162\sigma^2}{c}\sqrt{\frac{K^{2/\alpha}}{\alpha}} + c\sqrt{\frac{\alpha}{K^{2/\alpha}}}\right] \tag{E.75}$$

$$\because\ \lambda^2 = \frac{\sigma^2\alpha}{2c^2} \tag{E.76}$$

$$\leq \sqrt{KT}\left[6\sqrt{\pi}\frac{18\sigma^2}{c}\sqrt{\alpha K^{2/\alpha}} + \frac{9\sigma^2\alpha^{\frac{1}{2} + \frac{2}{\alpha}}}{2c}\left(K^{2/\alpha}\right)^{\frac{3}{2} - \frac{2}{\alpha}}\right. \tag{E.77}$$

$$\left. + \frac{c^2 + 162\sigma^2}{c}\sqrt{\frac{K^{2/\alpha}}{\alpha}} + c\sqrt{\frac{\alpha}{K^{2/\alpha}}}\right] \tag{E.78}$$

$$\leq\sqrt{KT}\Bigg[6\sqrt{\pi}\frac{18\sigma^2}{c}\sqrt{\alpha K^{2/\alpha}}+\frac{9\sigma^2\alpha^{\frac{1}{2}+\frac{2}{\alpha}}}{2c}K^{3/\alpha}+\frac{c^2+162\sigma^2}{c}\sqrt{\frac{K^{2/\alpha}}{\alpha}} \tag{E.79}$$

$$+c\sqrt{\frac{\alpha}{K^{2/\alpha}}}\Bigg] \tag{E.80}$$

$$\because\ \frac{3}{2}-\frac{2}{\alpha}\leq\frac{3}{2} \tag{E.81}$$

$$\leq O\left(\sqrt{\alpha^{1+\frac{4}{\alpha}}K^{1+\frac{6}{\alpha}}T}\right). \tag{E.82}$$

The optimal rate is obtained by setting $\alpha=\ln(K)$,

$$O\left(\sqrt{\ln(K)^{1+\frac{4}{\ln(K)}}K^{1+\frac{6}{\ln(K)}}T}\right)\leq O\left(\sqrt{KT\ln(K)}\right),$$

where $\ln(K)^{\frac{1}{2}+\frac{2}{\ln(K)}}\leq e^{2/e}\ln(K)$. Before proving the lower bound, note that

$$F^{-1}\left(1-\frac{1}{K}\right)=\ln\left(\frac{1}{1-\frac{1}{K}}\right)^{-1/\alpha}\geq(K-1)^{1/\alpha}$$

The lower bound is simply obtained by plugging F^{-1} into the general lower bound. The corollary is proved. □

Proof of Corollary 5. The CDF of a generalized extreme value distribution with $0\leq\zeta<1$ is given as

$$F(x)=\exp\left(-(1+\zeta x)^{-1/\zeta}\right)$$

Then, its inverse is

$$F^{-1}(y)=\frac{[\ln(1/y)]^{-\zeta}-1}{\zeta}\leq\frac{[1-y]^{-\zeta}-1}{\zeta},$$

where $\ln(x)\leq x-1$ is used. Then,

$$\left[F^{-1}\left(1-\frac{c^2}{T\Delta_a^2}\right)\right]^2\leq\left[\frac{\left[\frac{T\Delta_a^2}{c^2}\right]^{\zeta}-1}{\zeta}\right]^2.$$

We compute the $\sup h$ can be obtained as follows,

$$\begin{aligned}\sup h = &\sup_{x\in[0,\infty]} \frac{(1+\zeta x)^{-1/\zeta-1}\exp\left(-(1+\zeta x)^{-1/\zeta}\right)}{1-\exp\left(-(1+\zeta x)^{-1/\zeta}\right)} = \sup_{t\in[0,1]} \frac{t^{\zeta+1}\exp(-t)}{1-\exp(-t)}\\ &\le \sup_{t\in[0,1]} \frac{t\exp(-t)}{1-\exp(-t)} = 1.\end{aligned}$$

M_1 can be obtained,

$$\begin{aligned}\int_0^\infty \frac{\exp\left(-\frac{c^2z^2}{2\sigma^2}\right)}{1-F(z)}dz = &\int_0^\infty \frac{\exp\left(-\frac{c^2z^2}{2\sigma^2}\right)}{1-\exp\left(-(1+\zeta z)^{-1/\zeta}\right)}dz\\ \le &\int_0^\infty \left(1+(1+\zeta z)^{1/\zeta}\right)\exp\left(-\frac{c^2z^2}{2\sigma^2}\right)dz\\ =&\sqrt{\frac{\pi\sigma^2}{2c^2}} + \int_0^\infty (1+\zeta z)^{1/\zeta}\exp\left(-\frac{c^2z^2}{2\sigma^2}\right)dz = M_1\\ \le&\sqrt{\frac{\pi\sigma^2}{2c^2}} + \int_0^\infty \exp\left(-\frac{c^2z^2}{2\sigma^2}+\frac{\ln(1+\zeta z)}{\zeta}\right)dz\\ \le&\sqrt{\frac{\pi\sigma^2}{2c^2}} + \int_0^\infty \exp\left(-\frac{c^2z^2}{2\sigma^2}+z\right)dz\\ =&\sqrt{\frac{\pi\sigma^2}{2c^2}} + \sqrt{\frac{\pi\sigma^2}{2c^2}}\exp\left(\frac{2c^2}{\sigma^2}\right)\left(1+\operatorname{erf}\left(\sqrt{\frac{\sigma^2}{2c^2}}\right)\right)\\ \le&\sqrt{\frac{\pi\sigma^2}{2c^2}}\left(1+2\exp\left(\frac{\sigma^2}{2c^2}\right)\right) =: M_1\end{aligned}$$

For $\sum_{k=1}^T \Delta_a F\left(-\frac{\Delta_a\sqrt{k}}{6c}\right)$, we decompose the summation into two parts. The one is the case when $-\frac{\Delta_a\sqrt{k}}{6c}$ is placed on the support and the other is the case when $-\frac{\Delta_a\sqrt{k}}{6c}$ is placed on the outside of the support which is $(-1/\zeta,\infty)$,

$$\begin{aligned}\sum_{k=1}^T \Delta_a F\left(-\frac{\Delta_a\sqrt{k}}{6c}\right) \le &\sum_{k=1}^{\lfloor 36c^2/(\zeta\Delta_a)^2\rfloor} \Delta_a F\left(-\frac{\Delta_a\sqrt{k}}{6c}\right)\\ &+ \sum_{k=\lfloor 36c^2/(\zeta\Delta_a)^2\rfloor+1}^{T} \Delta_a F\left(-\frac{\Delta_a\sqrt{k}}{6c}\right).\end{aligned}$$

Then, the second term will be zero since it is outside of the support and, by using the fact that $F(x)\le\exp(-\exp(-x))$ and $F(x)$ is increasing, the first term can be bounded as follows,

$$\sum_{k=1}^{\lfloor 36c^2/(\zeta\Delta_a)^2 \rfloor} \Delta_a F\left(-\frac{\Delta_a\sqrt{k}}{6c}\right) \leq \int_0^{36c^2/(\zeta\Delta_a)^2} \Delta_a F\left(-\frac{\Delta_a\sqrt{x}}{6c}\right) dx$$
$$\leq \frac{72c^2}{\Delta_a}\int_0^{-\frac{1}{\zeta}} yF(y)dy \leq \frac{72c^2}{\Delta_a}\int_0^{-\infty} ye^{-e^{-y}}dy$$

Note that $\int_0^{-\infty} ye^{-e^{-y}}dy \leq 0.098 \leq 1$. Then, the problem dependent regret bound becomes,

$$\mathbb{E}\left[\mathcal{R}_T\right] \leq \sum_{a\neq a^\star}\left[\|h\|_\infty M_1 + \frac{F(0)}{1-F(0)}\right]\frac{18\sigma^2}{\Delta_a} + 2\sum_{k=1}^{T}\Delta_a F\left(-\frac{\Delta_a\sqrt{k}}{6c}\right) \tag{E.83}$$
$$+ \frac{9c^2}{\Delta_a}\left[F^{-1}\left(1-\frac{c^2}{T\Delta_a^2}\right)\right]^2 + \frac{c^2+162\sigma^2}{\Delta_a} + \Delta_a \tag{E.84}$$
$$\leq \sum_{a\neq a^\star}\left[M_1 + \frac{1}{e-1}\right]\frac{18\sigma^2}{\Delta_a} + \frac{144c^2}{\Delta_a} \tag{E.85}$$
$$+ \frac{9c^2}{\Delta_a}\left[\left(\frac{T\Delta_a^2}{c^2}\right)^\zeta - 1\right]^2/\zeta^2 + \frac{c^2+162\sigma^2}{\Delta_a} + \Delta_a \tag{E.86}$$
$$\leq C\left(\sum_{a\neq a^\star}\Delta_a + \frac{9c^2}{\zeta^2\Delta_a}\left[\left(\frac{T\Delta_a^2}{c^2}\right)^\zeta - 1\right]^2\right). \tag{E.87}$$

The problem independent regret bound can be obtained by choosing the threshold of the minimum gap as $\Delta = c\sqrt{K/T}\ln_\zeta(K)$ where $\ln_\zeta(x) := \frac{x^\zeta-1}{\zeta}$. Note that $\lim_{\zeta\to 0}\frac{x^\zeta-1}{\zeta} = \ln(x)$

$$\mathbb{E}\left[\mathcal{R}_T\right] \leq \sum_{a\neq a^\star}\left[M_1 + \frac{1}{e-1}\right]\frac{18\sigma^2}{\Delta_a} + \frac{9c^2}{\zeta^2\Delta_a}\left[\left(\frac{T\Delta_a^2}{c^2}\right)^\zeta - 1\right]^2 \tag{E.88}$$
$$+ \frac{145c^2+162\sigma^2}{\Delta_a} + \Delta_a \tag{E.89}$$
$$\leq K\left[\left[M_1 + \frac{1}{e-1}\right]\frac{18\sigma^2}{\Delta} + \frac{9c^2}{\zeta^2\Delta}\left[\left(\frac{T\Delta^2}{c^2}\right)^\zeta - 1\right]^2 + \frac{145c^2+162\sigma^2}{\Delta}\right] \tag{E.90}$$
$$+ \Delta T \tag{E.91}$$

$$\leq \sqrt{KT}\Bigg[\left[M_1 + \frac{1}{e-1}\right]\frac{18\sigma^2}{c\ln_\zeta(K)} + \frac{9c}{\ln_\zeta(K)}\left[\ln_\zeta(K\ln_\zeta(K)^2)\right]^2 \tag{E.92}$$

$$+ \frac{145c^2 + 162\sigma^2}{c\ln_\zeta(K)} + c\ln_\zeta(K)\Bigg] \tag{E.93}$$

$$\leq \sqrt{KT}\Bigg[\left[M_1 + \frac{1}{e-1}\right]\frac{18\sigma^2}{c\ln_\zeta(K)} + \frac{9c}{\ln_\zeta(K)}\left[\ln_\zeta(K^{2+\zeta})\right]^2 \tag{E.94}$$

$$+ \frac{145c^2 + 162\sigma^2}{c\ln_\zeta(K)} + c\ln_\zeta(K)\Bigg] \tag{E.95}$$

$$\because \ln_\zeta(x\ln_\zeta(x)^{2+\zeta}) \leq \ln_\zeta(x^2) \text{ for } x > 2 \tag{E.96}$$

$$\leq O\left(\sqrt{KT}\frac{\ln_\zeta\left(K^{2+\zeta}\right)^2}{\ln_\zeta(K)}\right). \tag{E.97}$$

The lower bound is simply obtained by plugging F^{-1} into the general lower bound. The corollary is proved. □

Appendix F

Proofs of Chapter 5.2.

F.1 Regret Lower Bound for Robust Upper Confidence Bound

In this section, we prove the lower bound of the expected cumulative regret of robust UCB [26]. First, we recall Assumption 5 in the main paper.

Assumption 7. *Let $\{Y_k\}_{k=1}^{\infty}$ be i.i.d. random variables with the finite p-th moment for $p \in (1, 2]$. Let ν_p be a bound of the p-th moment and y be the mean of Y_k. Assume that, for all $\delta \in (0, 1)$ and n number of observations, there exists an estimator $\hat{Y}_n(\eta, \nu_p, \delta)$ with a parameter η such that*

$$\mathbb{P}\left(\hat{Y}_n > y + \nu_p^{1/p}\left(\frac{\eta\ln(1/\delta)}{n}\right)^{1-1/p}\right) \leq \delta$$

$$\mathbb{P}\left(y > \hat{Y}_n + \nu_p^{1/p}\left(\frac{\eta\ln(1/\delta)}{n}\right)^{1-1/p}\right) \leq \delta.$$

Assumption 7 provides the confidence bound of the estimator $\hat{Y}_n$. Note that $\hat{Y}_n = \hat{Y}_n(\eta, \nu_p, \delta)$ requires ν_p and δ. By using this confidence bound, at round t, robust UCB selects an action based on the following strategy,

$$a_t := \arg\max_{a\in\mathcal{A}} \left\{ \hat{r}_{t-1,a} + \nu_p^{1/p} \left(\eta \ln(t^2)/n_{t-1,a}\right)^{1-1/p} \right\} \tag{F.1}$$

where $\hat{r}_{t-1,a}$ is an estimator which satisfies Assumption 7 with $\delta = t^{-2}$ and $n_{t-1,a}$ denotes the number of times $a \in \mathcal{A}$ have been selected. Under the strategy (F.1), we prove Theorem 1 in the main paper.

Proof of Theorem 28. The proof is done by constructing a counter example. We construct a K-armed bandit problem with deterministic rewards. Let the optimal arm $a^\star$ give the reward of $\Delta = \nu^{\frac{1}{p}} \left(\frac{\eta(K-1)\ln(T)}{T}\right)^{\frac{p-1}{p}}$ whereas the other arms provide zero rewards. Note that $\Delta \leq \nu^{\frac{1}{p}} \left(\frac{\eta(K-1)}{T^{\frac{1}{2}}}\right)^{\frac{p-1}{p}} < 1$ and the estimator we used satisfies $\hat{r}_a \leq \Delta\mathbb{I}[a = a^\star]$ for all a since rewards are Δ or 0 in this MAB problem. Let E_t be the set of events which satisfy

$$\sum_{a\neq a^\star} n_{t-1,a} \leq \frac{\nu^{\frac{1}{p-1}}\eta(K-1)}{2\left(\left(1+5^{\frac{p-1}{p}}\right)\Delta\right)^{\frac{p}{p-1}}} \ln(T^2) = \frac{T}{\left(1+5^{\frac{p-1}{p}}\right)^{\frac{p}{p-1}}}.$$

If $\mathbb{P}(E_t) \leq 1/2$ for some $t \in [1, \cdots, T]$, then, the regret bound is computed as follows,

$$\mathbb{E}\left[\mathcal{R}_T\right] \geq \frac{1}{2}\mathbb{E}\left[\mathcal{R}_t|E_t^c\right] \geq \frac{1}{2}\Delta\mathbb{E}\left[\sum_{a\neq a^\star} n_{t,a}\middle| E_t^c\right] \geq \frac{1}{2}\Delta\mathbb{E}\left[\sum_{a\neq a^\star} n_{t-1,a}\middle| E_t^c\right] \tag{F.2}$$

$$\geq \frac{\Delta}{2}\frac{T}{\left(1+5^{\frac{p-1}{p}}\right)^{\frac{p}{p-1}}} = \frac{\nu^{\frac{1}{p}}}{2\left(1+5^{\frac{p-1}{p}}\right)^{\frac{p}{p-1}}} \left(\eta(K-1)\ln(T)\right)^{\frac{p-1}{p}} T^{\frac{1}{p}}. \tag{F.3}$$

Hence, if $\mathbb{P}(E_t) \leq 1/2$ for some $t \in [1, \cdots, T]$, then, the lower bound holds. On the contrary, if $\mathbb{P}(E_t) > 1/2$ for all $t \in [1, \cdots, T]$, then, the proof is done by showing $\mathbb{P}(a_t \neq a^\star) \geq \frac{1}{2}$ for $t \geq t_0$ where

$$t_0 := \max\left(1 + \frac{2T}{5(K-1)} + \frac{2T}{\left(1+5^{\frac{p-1}{p}}\right)^{\frac{p}{p-1}}}, T^{\frac{1}{2}}\right).$$

Note that $T > t_0$ holds since $T > \frac{4T}{5} + 1 > 1 + \frac{2T}{5(K-1)} + \frac{2T}{\left(1+5^{\frac{p-1}{p}}\right)^{\frac{p}{p-1}}}$ holds for $T > 10$ and $T > \sqrt{T}$ holds. In other words, $\{t \in [1, \ldots, T] : t \geq t_0\}$ is not empty.

Before showing that $\mathbb{P}(a_t \neq a^\star) \geq \frac{1}{2}$ holds, we first check the lower bound. When $\mathbb{P}(E_t) > 1/2$ holds for all $t \in [1, \cdots, T]$, if $\mathbb{P}(a_t \neq a^\star) \geq \frac{1}{2}$ holds for $t \geq t_0$, then, the lower bound of the regret can be obtained as follows,

$$\mathbb{E}\left[\mathcal{R}_T\right] \geq \Delta \sum_{t=t_0}^{T} \mathbb{P}\left(a_t \neq a^\star\right) \geq \frac{\Delta(T-t_0)}{2} \tag{F.4}$$

$$= \frac{\Delta}{2} \min\left(\left(1 - \frac{2}{5(K-1)} - \frac{2}{\left(1+5^{\frac{p-1}{p}}\right)^{\frac{p}{p-1}}}\right) T - 1, T(1 - T^{-\frac{1}{2}})\right) \tag{F.5}$$

$$\geq \frac{\Delta}{2} \min\left(\left(1 - \frac{2}{5} - \frac{2}{5}\right) T - 1, T(1 - T^{-\frac{1}{2}})\right) \tag{F.6}$$

where the last inequality holds since $K - 1 > 1$ and $\left(1+5^{\frac{p-1}{p}}\right)^{\frac{p}{p-1}} > 5$. Then, by $T > 10$,

$$\frac{\Delta}{2} \min\left(\left(1 - \frac{2}{5} - \frac{2}{5}\right) T - 1, T(1 - T^{-\frac{1}{2}})\right) \tag{F.7}$$

$$\geq \frac{\Delta T}{2} \min\left(\frac{1}{5} - T^{-1}, 1 - T^{-\frac{1}{2}}\right) \tag{F.8}$$

$$= \nu^{\frac{1}{p}} \left(\eta(K-1)\ln(T)\right)^{\frac{p-1}{p}} T^{\frac{1}{p}} \min\left(\frac{1}{5} - T^{-1}, 1 - T^{-\frac{1}{2}}\right) \tag{F.9}$$

$$= \frac{1}{10} \nu^{\frac{1}{p}} \left(\eta(K-1)\ln(T)\right)^{\frac{p-1}{p}} T^{\frac{1}{p}}. \tag{F.10}$$

Note that $\frac{1}{10} < 1 - \frac{1}{\sqrt{10}}$. Thus, we obtain $\mathbb{E}\left[\mathcal{R}_T\right] \geq \Omega\left(\left(K\ln(T)\right)^{\frac{p-1}{p}} T^{\frac{1}{p}}\right)$, if $\mathbb{P}(a_t \neq a^\star) \geq \frac{1}{2}$ holds for $t \geq t_0$.

The remaining part is to prove that $\mathbb{P}(a_t \neq a^\star) \geq \frac{1}{2}$ holds for $t > t_0$ when $\mathbb{P}(E_t) \geq 1/2$ for all $t > 0$. We mainly prove that, if E_t occurs, $a_t = a^\star$ never occurs since the confidence bound cannot overcome the estimation error between sub-optimal arms and optimal arm under the condition of E_t. In other words,

$\mathbb{P}\left(a_t \neq a^\star | E_t\right) = 1$. If $\mathbb{P}\left(a_t \neq a^\star | E_t\right) = 1$ holds, then, we can simply show that

$$\mathbb{P}(a_t \neq a^\star) \geq \frac{1}{2}\mathbb{P}\left(a_t \neq a^\star | E_t\right) = \frac{1}{2}. \tag{F.11}$$

Now, we analyze the set of event, $\{a_t \neq a^\star\}$, as follows,

$$\{a_t \neq a^\star\} = \bigcup_{a \neq a^\star} \left\{ \hat{r}_{a^\star} + \nu^{\frac{1}{p}} \left(\frac{\eta \ln(t^2)}{n_{t-1,a^\star}}\right)^{\frac{p-1}{p}} \leq \hat{r}_a + \nu^{\frac{1}{p}} \left(\frac{\eta \ln(t^2)}{n_{t-1,a}}\right)^{\frac{p-1}{p}} \right\} \tag{F.12}$$

$$\supset \bigcup_{a \neq a^\star} \left\{ \Delta + \nu^{\frac{1}{p}} \left(\frac{\eta \ln(t^2)}{n_{t-1,a^\star}}\right)^{\frac{p-1}{p}} \leq \nu^{\frac{1}{p}} \left(\frac{\eta \ln(t^2)}{n_{t-1,a}}\right)^{\frac{p-1}{p}} \right\} \tag{F.13}$$

$$\because \hat{r}_{a^\star} \leq \Delta \text{ and } \hat{r}_{a \neq a^\star} = 0 \tag{F.14}$$

$$\supset \bigcup_{a \neq a^\star} \left\{ \Delta + \nu^{\frac{1}{p}} \left(\frac{\eta \ln(t^2)}{n_{t-1,a^\star}}\right)^{\frac{p-1}{p}} \leq \left(1 + 5^{\frac{p-1}{p}}\right) \Delta \leq \nu^{\frac{1}{p}} \left(\frac{\eta \ln(t^2)}{n_{t-1,a}}\right)^{\frac{p-1}{p}} \right\} \tag{F.15}$$

$$= \left\{ \nu^{\frac{1}{p}} \left(\frac{\eta \ln(t^2)}{n_{t-1,a^\star}}\right)^{\frac{p-1}{p}} \leq 5^{\frac{p-1}{p}} \Delta \right\} \bigcap \bigcup_{a \neq a^\star} \left\{ \left(1 + 5^{\frac{p-1}{p}}\right) \Delta \leq \nu^{\frac{1}{p}} \left(\frac{\eta \ln(t^2)}{n_{t-1,a}}\right)^{\frac{p-1}{p}} \right\} \tag{F.16}$$

$$= \left\{ \frac{2\nu^{\frac{1}{p-1}}}{5\Delta^{\frac{p}{p-1}}} \eta \ln(t) \leq n_{t-1,a^\star} \right\} \bigcap \bigcup_{a \neq a^\star} \left\{ n_{t-1,a} \leq \frac{2\nu^{\frac{1}{p-1}}}{\left(\left(1 + 5^{\frac{p-1}{p}}\right)\Delta\right)^{\frac{p}{p-1}}} \eta \ln(t) \right\} \tag{F.17}$$

$$\supset \left\{ \frac{2\nu^{\frac{1}{p-1}}}{5\Delta^{\frac{p}{p-1}}} \eta \ln(T) \leq n_{t-1,a^\star} \right\} \bigcap \bigcup_{a \neq a^\star} \left\{ n_{t-1,a} \leq \frac{2\nu^{\frac{1}{p-1}}}{\left(\left(1 + 5^{\frac{p-1}{p}}\right)\Delta\right)^{\frac{p}{p-1}}} \eta \ln(t_0) \right\} \tag{F.18}$$

$$\because T > t > t_0 \tag{F.19}$$

$$\supset \left\{ \frac{2T}{5(K-1)} \leq n_{t-1,a^\star} \right\} \bigcap \bigcup_{a \neq a^\star} \left\{ n_{t-1,a} \leq \frac{2T}{\left(1 + 5^{\frac{p-1}{p}}\right)^{\frac{p}{p-1}} (K-1)} \frac{\ln(t_0)}{\ln(T)} \right\} \tag{F.20}$$

$$\supset \left\{ \frac{2T}{5(K-1)} \leq n_{t-1,a^\star} \right\} \bigcap \left\{ \sum_{a \neq a^\star} n_{t-1,a} \leq \frac{2T}{\left(1 + 5^{\frac{p-1}{p}}\right)^{\frac{p}{p-1}}} \frac{\ln(t_0)}{\ln(T)} \right\}. \tag{F.21}$$

Let $A := \left\{ \frac{2T}{5(K-1)} \leq n_{t-1,a^\star} \right\}$ and $B := \left\{ \sum_{a \neq a^\star} n_{t-1,a} \leq \frac{2T}{\left(1 + 5^{\frac{p-1}{p}}\right)^{\frac{p}{p-1}}} \frac{\ln(t_0)}{\ln(T)} \right\}$.

Now, we check that $A \cap B$ contains E_t for $t \geq t_0$ where

$$t_0 := \max\left(1 + \frac{2T}{5(K-1)} + \frac{2T}{\left(1 + 5^{\frac{p-1}{p}}\right)^{\frac{p}{p-1}}}, T^{\frac{1}{2}}\right).$$

For the set A, if $\omega \in E_t$, then,

$$\begin{aligned} n_{t-1,a^\star} &= t - 1 - \sum_{a \neq a^\star} n_{t-1,a} \geq t - 1 - \frac{T}{\left(1 + 5^{\frac{p-1}{p}}\right)^{\frac{p}{p-1}}} \quad \because \omega \in E_t & \text{(F.22)} \\ &\geq t_0 - 1 - \frac{T}{\left(1 + 5^{\frac{p-1}{p}}\right)^{\frac{p}{p-1}}} \geq \frac{2T}{5(K-1)} + \frac{T}{\left(1 + 5^{\frac{p-1}{p}}\right)^{\frac{p}{p-1}}} & \text{(F.23)} \\ &\geq \frac{2T}{5(K-1)}, & \text{(F.24)} \end{aligned}$$

which implies $\omega \in A$.

For the set B, we have,

$$\frac{\ln(t_0)}{\ln(T)} \geq \frac{\ln(T^{\frac{1}{2}})}{\ln(T)} = \frac{1}{2}.$$

By using this fact, we get

$$\frac{2T}{\left(1 + 5^{\frac{p-1}{p}}\right)^{\frac{p}{p-1}}} \frac{\ln(t_0)}{\ln(T)} \geq \frac{T}{\left(1 + 5^{\frac{p-1}{p}}\right)^{\frac{p}{p-1}}} \geq \sum_{a \neq a^\star} n_{t-1,a} \quad \because \omega \in E_t, \tag{F.25}$$

which implies $\omega \in B$. In summary, $\omega \in E_t$ implies $\omega \in A \cap B$. Consequently, we have,

$$\begin{aligned} \mathbb{P}(a_t \neq a^\star) &\geq \frac{1}{2}\mathbb{P}\left(a_t \neq a^\star | E_t\right) & \text{(F.26)} \\ &\geq \frac{1}{2}\mathbb{P}\left(A \cap B | E_t\right) = \frac{1}{2}. & \text{(F.27)} \end{aligned}$$

Thus,

$$\mathbb{E}\left[\mathcal{R}_T\right] \geq \Omega\left(\left(K\ln(T)\right)^{\frac{p-1}{p}} T^{\frac{1}{p}}\right).$$

□

F.2 Bounds on Tail Probability of A p-Robust Estimator

Before deriving the bound of tail probability of a new estimator, we first analyze the property of the influence function $\psi(x)$. Then, using the property of $\psi(x)$, we show that the tail probability has an exponential upper bound.

Lemma 25. *For $p \in (1,2]$, assume that a positive constant b_p satisfies the following inequality,*

$$b_p^{\frac{2}{p}}\left[2\left(\frac{2-p}{p-1}\right)^{1-\frac{2}{p}}+\left(\frac{2-p}{p-1}\right)^{2-\frac{2}{p}}\right] \geq 1.$$

Then, the following inequality holds, for all $x \in \mathbb{R}$,

$$\ln\left(1+x+b_p|x|^p\right) \geq -\ln\left(1-x+b_p|x|^p\right).$$

Proof. Let $f(x) := 1+x+b_p|x|^p$. Then, the inequality is represented as $\ln(f(x)) \geq -\ln(f(-x))$. Before starting the proof, first, we show that $f(x) > 0$ by checking $\min_x f(x) > 0$. For $x \geq 0$,

$$f'(x) = 1 + b_p \cdot px^{p-1} > 0.$$

which is non-zero for all $x \geq 0$. Thus, the minimum of $f(x)$ will appear at $x < 0$. For $x < 0$, its derivative is

$$f'(x) = 1 - b_p \cdot p(-x)^{p-1}.$$

Then, $f'(x)$ become zero at $x = -\left(pb_p\right)^{-\frac{1}{p-1}}$. Thus, the minimum of $f(x)$ is

$$f\left(-\left(pb_p\right)^{-\frac{1}{p-1}}\right) = 1-\left(pb_p\right)^{-\frac{1}{p-1}} + b_p\left(pb_p\right)^{-\frac{p}{p-1}} = 1-\left(p^{-\frac{1}{p-1}} - p^{-\frac{p}{p-1}}\right) b_p^{-\frac{1}{p-1}} \tag{F.28}$$

$$\geq 1-\left(p^{-\frac{1}{p-1}} - p^{-\frac{p}{p-1}}\right)\left[2\left(\frac{2-p}{p-1}\right)^{1-\frac{2}{p}} + \left(\frac{2-p}{p-1}\right)^{2-\frac{2}{p}}\right]^{\frac{p}{2(p-1)}} \tag{F.29}$$

$$\because \left[2\left(\frac{2-p}{p-1}\right)^{1-\frac{2}{p}} + \left(\frac{2-p}{p-1}\right)^{2-\frac{2}{p}}\right]^{\frac{p}{2(p-1)}} \geq b_p^{-\frac{1}{p-1}} \tag{F.30}$$

$$= 1-p^{-\frac{p}{p-1}}\left[2\left(p-1\right)\left(2-p\right)^{1-\frac{2}{p}} + \left(2-p\right)^{2-\frac{2}{p}}\right]^{\frac{p}{2(p-1)}} \tag{F.31}$$

$$= 1-p^{-\frac{p}{p-1}}\left[2\left(p-1\right)+\left(2-p\right)\right]^{\frac{p}{2(p-1)}}\left(2-p\right)^{\frac{p-2}{2(p-1)}} \tag{F.32}$$

$$= 1-p^{-\frac{p}{2(p-1)}}\left(2-p\right)^{\frac{p-2}{2(p-1)}} > 0. \tag{F.33}$$

Note that $\frac{1}{2} \leq p^{-\frac{p}{2(p-1)}}\left(2-p\right)^{\frac{p-2}{2(p-1)}} < 1$ holds for $p \in (1, 2]$. Since $f(-x)$ and $f(x)$ are symmetric to the y-axis, $f(-x)$ is also positive for all $x \in \mathbb{R}$.

By noticing that $\ln(f(x)) \geq -\ln(f(-x))$ is equivalent to $f(x)f(-x) > 1$, We show that the following inequality holds,

$$(1+x+b_p|x|^p)(1-x+b_p|x|^p) \geq 1 \tag{F.34}$$

$$b_p^2|x|^{2p} + 2b_p|x|^p + 1 - x^2 \geq 1 \tag{F.35}$$

$$b_p^2|x|^{2p-2} + 2b_p|x|^{p-2} - 1 \geq 0 \;\; (\because \; x^2 \geq 0). \tag{F.36}$$

Let us define $g(z) := b_p^2 z^{2p-2} + 2b_p z^{p-2}$ for $z > 0$. Now, we show that $g(z) > 1$ holds for $z > 0$. First, we analyze the derivative of $g(z)$ computed as follows,

$$g'(z) = 2b_p z^{p-3}\left(b_p(p-1)z^p + (p-2)\right).$$

Since $b_p > 0$ and $z^{p-3} > 0$, the sign of $g'(z)$ is determined by the term:

$$\left(b_p(p-1)z^p + (p-2)\right),$$

which is an increasing function and, hence, has a unique root at $z_0 := \left(\frac{(2-p)}{(p-1)}\right)^{\frac{1}{p}} b_p^{-\frac{1}{p}}$. In other words, since $(b_p(p-1)z^p + (p-2))$ has the unique root at z_0 for $z > 0$, $g'(z)$ also has a unique root at z_0 which is the minimum point. Finally,

$$g\left(z_0\right) - 1 = b_p^{\frac{2}{p}} \left[2\left(\frac{2-p}{p-1}\right)^{1-\frac{2}{p}} + \left(\frac{2-p}{p-1}\right)^{2-\frac{2}{p}}\right] - 1 \geq 0.$$

where the last inequality holds by the assumption. Consequently, $g(z) - 1 \geq g\left(z_0\right) - 1 \geq 0$ holds and, hence, $f(x)f(-x) \geq 1$ holds. The lemma is proved. □

Corollary 13. *Let* $b_p := \left[2\left(\frac{2-p}{p-1}\right)^{1-\frac{2}{p}} + \left(\frac{2-p}{p-1}\right)^{2-\frac{2}{p}}\right]^{-\frac{p}{2}}$. *For all* $x \in \mathbb{R}$, *the following inequality holds*

$$\ln\left(1 + x + b_p|x|^p\right) \geq -\ln\left(1 - x + b_p|x|^p\right).$$

Proof. The proof is done by directly applying the Lemma 25 with

$$b_p = \left[2\left(\frac{2-p}{p-1}\right)^{1-\frac{2}{p}} + \left(\frac{2-p}{p-1}\right)^{2-\frac{2}{p}}\right]^{-\frac{p}{2}}.$$

□

Proof of Theorem 29. From the Markov's inequality,

$$\mathbb{P}\left(\frac{n^{1-\frac{1}{p}}}{c}\hat{Y}_n > \frac{n^{1-\frac{1}{p}}}{c}(y+\delta)\right) \leq \exp\left(-\frac{n^{1-\frac{1}{p}}}{c}(y+\delta)\right)\mathbb{E}\left[\exp\left(\frac{n^{1-\frac{1}{p}}}{c}\hat{Y}_n\right)\right] \tag{F.37}$$

Since $\psi(x) \leq \ln\left(b_p|x|^p + x + 1\right)$ holds by its definition, we have

$$\mathbb{E}\left[\exp\left(\frac{n^{1-\frac{1}{p}}}{c}\hat{Y}_n\right)\right] \leq \mathbb{E}\left[\prod_{k=1}^{n}\left(1 + \frac{Y_k}{cn^{\frac{1}{p}}} + b_p\frac{Y_k^p}{2(cn^{\frac{1}{p}})^p}\right)\right] \tag{F.38}$$

$$= \prod_{k=1}^{n}\mathbb{E}\left[1 + \frac{Y_k}{cn^{\frac{1}{p}}} + b_p\frac{Y_k^p}{2c^p n}\right] \tag{F.39}$$

$$= \left(1 + \frac{y}{cn^{\frac{1}{p}}} + b_p\frac{v_p}{2c^p n}\right)^n \tag{F.40}$$

$$\leq \exp\left(\frac{n^{1-\frac{1}{p}}}{c}y + b_p\frac{v_p}{2c^p}\right) \tag{F.41}$$

Combining (F.37) and (F.41), we have

$$\begin{aligned}\mathbb{P}\left(\hat{Y}_n - y > \delta\right) &\leq \exp\left(-\frac{n^{1-\frac{1}{p}}}{c}(y+\delta)\right)\exp\left(\frac{n^{1-\frac{1}{p}}}{c}y + \frac{b_p\nu_p}{2c^p}\right)\\ &= \exp\left(-\frac{n^{1-\frac{1}{p}}}{c}\delta + \frac{b_p\nu_p}{2c^p}\right)\end{aligned}$$

The upper bound of $\mathbb{P}\left(y - \hat{Y}_n > \delta\right)$ can be obtained by the similar way. Hence we obtain the desired result. The theorem is proved. □

F.3 General Regret Upper Bound of APE^2

To analyze the regret $\mathcal{R}_T$ in the view of expectation, we borrow the notion of filtration $\{\mathcal{H}_t : t = 1, \dots, T\}$ from [7] and [64] where the filtration $\mathcal{H}_t$ is defined as the history of plays until time t as follows

$$\mathcal{H}_t := \{a_\ell, \mathbf{R}_{a_\ell} : \ell = 1, \dots, t\}$$

By definition, $\mathcal{H}_1 \subset \mathcal{H}_2 \subset \cdots \subset \mathcal{H}_{T-1}$ holds. Finally, we separates the event $\{a_t = a\}$ into three groups based on the threshold $x_a := r_a + \Delta_a/3$ and $y_a := r_{a^\star} - \Delta_a/3$. Finally, for a given reward estimator $\hat{r}_{t,a}$, let us define the following sets which will be used to partition the event $\{a_t = a\}$:

$$E_{t,a} := \{a_t = a\}, \quad \hat{E}_{t,a} := \{\hat{r}_{t,a} \leq x_a\}, \quad \tilde{E}_{t,a} := \{\hat{r}_{t-1,a} + \beta_{t-1,a}G_{t,a} \leq y_a\}$$

We separate $E_{t,a}$ into three subsets:

$$E_{t,a} = E_{t,a}^{(1)} \cup E_{t,a}^{(2)} \cup E_{t,a}^{(3)} \tag{F.42}$$

where

$$\begin{aligned}E_{t,a}^{(1)} &= E_{t,a} \cap \hat{E}_{t,a}^c\\ E_{t,a}^{(2)} &= E_{t,a} \cap \hat{E}_{t,a} \cap \tilde{E}_{t,a}\\ E_{t,a}^{(3)} &= E_{t,a} \cap \hat{E}_{t,a} \cap \tilde{E}_{t,a}^c\end{aligned}$$

In the following sections, we estimate the upper bound of the probability of the event $E_{t,a}$ based on the decomposition (F.42).

Lemma 26. *Assume that the p-th moment of rewards is bounded by a constant $\nu_p < \infty$, $\hat{r}_{t,a}$ is a p-robust estimator and $F(x)$ satisfies Assumption 2. Then for any action $a \in \mathcal{A}$, it holds*

$$\sum_{t=1}^{T} \mathbb{P}\left(E_{t,a}^{(1)}\right) \leq 1 + \exp\left(\frac{b_p \nu_p}{2c^p}\right)\left(\frac{3c}{\Delta_a}\right)^{\frac{p}{p-1}} \Gamma\left(\frac{2p-1}{p-1}\right).$$

Proof. Fix arm $a \in \mathcal{A}$. Let τ_k denotes the smallest round when the arm a is sampled for the k-th time i.e. $k = \sum_{t=1}^{\tau_k} \mathbb{I}[E_{t,a}]$. We let $\tau_0 := 0$ and $\tau_k = T$ for $k > n_a(T)$. Then it is easy to see that for $\tau_k < t \leq \tau_{k+1}$

$$\mathbb{I}[E_{t,a}] = \begin{cases} 1 & : t = \tau_{k+1} \\ 0 & : t \neq \tau_{k+1} \end{cases} \tag{F.43}$$

Therefore,

$$\begin{aligned} \sum_{t=1}^{T} \mathbb{P}\left(E_{t,a}^{(1)}\right) = \sum_{t=1}^{T} \mathbb{E}\left[\mathbb{I}[E_{t,a}^{(1)}]\right] &= \sum_{k=0}^{T-1} \mathbb{E}\left[\sum_{t=1+\tau_k}^{\tau_{k+1}} \mathbb{I}[E_{t,a}^{(1)}]\right] \\ &= \mathbb{E}\left[\sum_{t=1}^{\tau_1} \mathbb{I}\left(E_{t,a} \cap \hat{E}_{t,a}^{c}\right)\right] \\ &+ \sum_{k=1}^{T-1} \mathbb{E}\left[\sum_{t=1+\tau_k}^{\tau_{k+1}} \mathbb{I}[E_{t,a} \cap \hat{E}_{t,a}^{c}]\right] \\ &\leq 1 + \sum_{k=1}^{T-1} \mathbb{P}\left(\hat{E}_{\tau_{k+1},a}^{c}\right) \end{aligned}$$

where the last inequality holds by (F.43). Also, by the definition of $\hat{E}_{t,a}$ and

Theorem 29,

$$\begin{aligned}\sum_{k=1}^{T-1}\mathbb{P}\left(\hat{E}^c_{\tau_{k+1},a}\right) &\leq \sum_{k=1}^{T-1}\exp\left(-\frac{\Delta_a k^{1-\frac{1}{p}}}{3c}+\frac{b_p\nu_p}{2c^p}\right)\\ &\leq \exp\left(\frac{b_p\nu_p}{2c^p}\right)\int_0^\infty \exp\left(-\frac{\Delta_a x^{1-\frac{1}{p}}}{3c}\right)dx\\ &\leq \exp\left(\frac{b_p\nu_p}{2c^p}\right)\left(\frac{3c}{\Delta_a}\right)^{\frac{p}{p-1}}\frac{p}{p-1}\int_0^\infty \exp\left(-t\right)t^{\frac{1}{p-1}}dt\\ &\because\ t=\frac{\Delta_a x^{1-\frac{1}{p}}}{3c}\\ &= \exp\left(\frac{b_p\nu_p}{2c^p}\right)\left(\frac{3c}{\Delta_a}\right)^{\frac{p}{p-1}}\frac{p}{p-1}\Gamma\left(\frac{p}{p-1}\right)\\ &= \exp\left(\frac{b_p\nu_p}{2c^p}\right)\left(\frac{3c}{\Delta_a}\right)^{\frac{p}{p-1}}\Gamma\left(\frac{2p-1}{p-1}\right).\end{aligned}$$

where the last equality holds by $\Gamma(x+1)=x\Gamma(x)$. The lemma is proved. □

Next we estimate $E^{(2)}_{t,a}$. From now on, we let ρ stand for the following ratio

$$\rho(g):=\frac{F(g)}{1-F(g)}=\frac{\mathbb{P}(G<g)}{\mathbb{P}(G\geq g)}$$

where F is a cumulative density function of perturbation G.

Lemma 27. *Assume that the p-th moment of rewards is bounded by a constant $\nu_p<\infty$, $\hat{r}_{t,a}$ is a p-robust estimator and $F(x)$ satisfies Assumption 2. For any action $a\in\mathcal{A}$, it holds*

$$\begin{aligned}\sum_{t=1}^{T}\mathbb{P}\left(E^{(2)}_{t,a}\right) \leq \exp\left(\frac{b_p\nu_p}{2c^p}\right)&\left\{C_1+\frac{F(0)}{1-F(0)}+2^{\frac{2p-1}{p-1}}\right\}\Gamma\left(\frac{2p-1}{p-1}\right)\left(\frac{3c}{\Delta_a}\right)^{\frac{p}{p-1}}\\ &+2\left(\frac{6c}{\Delta_a}\right)^{\frac{p}{p-1}}\left\{-F^{-1}\left(\frac{1}{T}\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}}\right)\right\}_+^{\frac{p}{p-1}}+2\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}}\end{aligned}$$

Proof. If $a=a^\star$, then $\Delta_a=0$ so the desired result trivially holds. Threfore, we take $a\in\mathcal{A}\setminus\{a^\star\}$. For the convenience of the notation, we write $\tilde{r}_{t,a}:=$

$\hat{r}_{t-1,a} + \beta_{t-1,a} G_{t,a}$. Due to the decision rule of the perturbation method, $a_t = a$ implies $\tilde{r}_{t,a'} \leq \tilde{r}_{t,a}$ for $a' \in \mathcal{A}$. Therefore, it holds

$$E_{t,a} \cap \tilde{E}_{t,a} \subset \bigcap_{a' \in \mathcal{A}} \{\tilde{r}_{t,a'} \leq y_a\} = \{\tilde{r}_{t,a^\star} \leq y_a\} \cap \{\tilde{r}_{t,a'} \leq y_a, \forall a' \neq a_\star\}. \tag{F.44}$$

This fact implies

$$\mathbb{P}\left(E_{t,a} \cap \tilde{E}_{t,a} | \mathcal{H}_{t-1}\right) \leq \mathbb{P}\left(\bigcap_{a' \in \mathcal{A}} \{\tilde{r}_{t,a'} \leq y_a\} | \mathcal{H}_{t-1}\right) \tag{F.45}$$

Note that events $\{\tilde{r}_{t,a^\star} \leq y_a\}$ and $\{\tilde{r}_{t,a'} \leq y_a, \forall a' \neq a_\star\}$ are independent if $\mathcal{H}_{t-1}$ is given. From this fact, (F.45) is equivalent to

$$\begin{aligned}\mathbb{P}\left(\bigcap_{a' \in \mathcal{A}} \{\tilde{r}_{t,a'} \leq y_a\} | \mathcal{H}_{t-1}\right) &= \mathbb{P}\left(\tilde{r}_{t,a^\star} \leq y_a | \mathcal{H}_{t-1}\right) \mathbb{P}\left(\tilde{r}_{t,a'} \leq y_a, \forall a' \neq a_\star | \mathcal{H}_{t-1}\right) \\ &= \frac{\mathbb{P}\left(\tilde{r}_{t,a^\star} \leq y_a | \mathcal{H}_{t-1}\right)}{\mathbb{P}\left(\tilde{r}_{t,a^\star} > y_a | \mathcal{H}_{t-1}\right)} \\ &\qquad \times \mathbb{P}\left(\{\tilde{r}_{t,a^\star} > y_a\} \cap \{\tilde{r}_{t,a'} \leq y_a, \forall a' \neq a_\star\} | \mathcal{H}_{t-1}\right)\end{aligned}$$

Since $\hat{r}_{t-1,a^\star}, \beta_{t-1,a^\star}$ are already determined under the condition $\mathcal{H}_{t-1}$, we get

$$\mathbb{P}\left(\tilde{r}_{t,a^\star} \leq y_a | \mathcal{H}_{t-1}\right) = F\left(\frac{r_{a^\star} - \hat{r}_{t-1,a^\star} - \frac{\Delta_a}{3}}{\beta_{t-1,a^\star}}\right)$$

Similarly to (F.44), we can observe that

$$\{\tilde{r}_{t,a^\star} > y_a\} \cap \{\tilde{r}_{t,a'} \leq y_a, \forall a' \neq a_\star\} \subset E_{t,a^\star} \cap \tilde{E}_{t,a} \tag{F.46}$$

and this implies

$$\mathbb{P}\left(\{\tilde{r}_{t,a^\star} > y_a\} \cap \{\tilde{r}_{t,a'} \leq y_a, \forall a' \neq a_\star\} | \mathcal{H}_{t-1}\right) \leq \mathbb{P}\left(E_{t,a^\star} \cap \tilde{E}_{t,a} | \mathcal{H}_{t-1}\right) \tag{F.47}$$

Therefore,

$$\mathbb{P}\left(E_{t,a} \cap \tilde{E}_{t,a} | \mathcal{H}_{t-1}\right) \leq \frac{Q_{t,a^\star}}{1 - Q_{t,a^\star}} \mathbb{P}\left(E_{t,a^\star} \cap \tilde{E}_{t,a} | \mathcal{H}_{t-1}\right), \tag{F.48}$$

where $Q_{t,a^\star} := F\left(\frac{r_{a^\star} - \hat{r}_{t-1,a^\star} - \frac{\Delta_a}{3}}{\beta_{t-1,a^\star}}\right)$. By taking an expectation on both sides, we have,

$$\mathbb{P}\left(E^{(2)}_{t,a}\right) = \mathbb{P}\left(E_{t,a} \cap \hat{E}_{t,a} \cap \tilde{E}_{t,a}\right) \leq \mathbb{E}\left[\frac{Q_{t,a^\star}}{1 - Q_{t,a^\star}} \mathbb{I}[E_{t,a^\star} \cap \hat{E}_{t,a} \cap \tilde{E}_{t,a}]\right]. \quad \text{(F.49)}$$

Now, we set τ_k to denote the smallest round when the optimal arm $a^\star$ is sampled for the k-th time. Then, the summation of the right-hand side of F.49 over $t = 1, \ldots, T$ is bounded as follows,

$$\begin{aligned}
&\sum_{t=1}^{T} \mathbb{E}\left[\frac{Q_{t,a^\star}}{1 - Q_{t,a^\star}} \mathbb{I}[E_{t,a^\star} \cap \hat{E}_{t,a} \cap \tilde{E}_{t,a}]\right] \\
&= \sum_{k=0}^{T-1} \mathbb{E}\left[\sum_{t=\tau_k+1}^{\tau_{k+1}} \frac{Q_{t,a^\star}}{1 - Q_{t,a^\star}} \mathbb{I}[E_{t,a^\star} \cap \hat{E}_{t,a} \cap \tilde{E}_{t,a}]\right] \\
&= \sum_{k=0}^{T-1} \mathbb{E}\left[\frac{Q_{\tau_{k+1},a^\star}}{1 - Q_{\tau_{k+1},a^\star}} \mathbb{I}[\hat{E}_{\tau_{k+1},a} \cap \tilde{E}_{\tau_{k+1},a}]\right] \\
&\leq \sum_{k=1}^{T} \mathbb{E}\left[\frac{Q_{\tau_k,a^\star}}{1 - Q_{\tau_k,a^\star}}\right].
\end{aligned}$$

We first compute the upper bound of the conditional expectation

$$\mathbb{E}\left[\frac{Q_{\tau_k,a^\star}}{1 - Q_{\tau_k,a^\star}} \middle| \mathcal{H}_{\tau_k}\right].$$

From the definition of τ_k, we have $n_{\tau_k,a} = k$ and $\beta_{\tau_k,a} = \frac{c}{k^{1-\frac{1}{p}}}$. By using this fact, we get,

$$\begin{aligned}
\mathbb{E}\left[\frac{Q_{\tau_k,a^\star}}{1 - Q_{\tau_k,a^\star}} \middle| \mathcal{H}_{\tau_k}\right] &= \mathbb{E}\left[\rho\left(\frac{k^{1-\frac{1}{p}}}{c}\left\{r_{a^\star} - \hat{r}_{\tau_k,a^\star} - \frac{\Delta_a}{3}\right\}\right) \middle| \mathcal{H}_{\tau_k}\right] \\
&= \int_{\mathbb{R}} \rho\left(\frac{k^{1-\frac{1}{p}}}{c}\left\{r_{a^\star} - x - \frac{\Delta_a}{3}\right\}\right) \mathbb{P}(\hat{r} \in \mathrm{d}x) \qquad \text{(F.50)}
\end{aligned}$$

We decompose $\mathbb{R} = I_1 \cup I_2 \cup I_3$ into three intervals where $I_1 := \{x \leq r_{a^\star} - \frac{\Delta_a}{3}\}$, $I_2 := \{r_{a^\star} - \frac{\Delta_a}{3} < x \leq r_{a^\star} - \frac{\Delta_a}{6}\}$, and $I_3 := \{r_{a^\star} - \frac{\Delta_a}{6} < x\}$. We derive the upper bound of (F.50) on the each interval.

By using the change of variable formula,

$$\begin{aligned}
&\int_{I_1} \rho\left(\frac{k^{1-\frac{1}{p}}}{c}\left\{r_{a^\star} - x - \frac{\Delta_a}{3}\right\}\right)\mathbb{P}(\hat{r} \in \mathrm{d}x) \\
&= \int_{-\infty}^{r_{a^\star}-\frac{\Delta_a}{3}} \rho\left(\frac{k^{1-\frac{1}{p}}}{c}\left\{r_{a^\star} - x - \frac{\Delta_a}{3}\right\}\right) f_{\hat{r}}(x)\mathrm{d}x \\
&= \frac{c}{k^{1-\frac{1}{p}}}\int_0^\infty \rho(g) f_{\hat{r}}\left(r_{a^\star} - \frac{c}{k^{1-\frac{1}{p}}}g - \frac{\Delta_a}{3}\right)\mathrm{d}g
\end{aligned}$$

where $f_{\hat{r}}$ is the density function of the measure $\mathbb{P}(\hat{r} \in \mathrm{d}x)$. Note that the following equality holds by the fundamental theorem of calculus

$$\rho(g) = \frac{F(g)}{1-F(g)} = \int_0^g \frac{h(u)}{1-F(u)}\mathrm{d}u + \frac{F(0)}{1-F(0)}$$

Therefore,

$$\begin{aligned}
&\frac{c}{k^{1-\frac{1}{p}}}\int_0^\infty \frac{F(g)}{1-F(g)} f_{\hat{r}}\left(r_{a^\star} - \frac{c}{k^{1-\frac{1}{p}}}g - \frac{\Delta_a}{3}\right)\mathrm{d}g \\
&\quad= \frac{c}{k^{1-\frac{1}{p}}}\int_0^\infty \left(\int_0^g \frac{h(u)}{1-F(u)}\mathrm{d}u + \frac{F(0)}{1-F(0)}\right) f_{\hat{r}}\left(r_{a^\star} - \frac{c}{k^{1-\frac{1}{p}}}g - \frac{\Delta_a}{3}\right)\mathrm{d}g \\
&\quad= \frac{F(0)}{1-F(0)}\mathbb{P}\left(\frac{\Delta_a}{3} \le r_{a^\star} - \hat{r}_{\tau_k, a^\star}\right) \\
&\qquad + \frac{c}{k^{1-\frac{1}{p}}}\int_0^\infty \left(\int_0^g \frac{h(u)}{1-F(u)}\mathrm{d}u\right) f_{\hat{r}}\left(r_{a^\star} - \frac{c}{k^{1-\frac{1}{p}}}g - \frac{\Delta_a}{3}\right)\mathrm{d}g. \qquad \text{(F.51)}
\end{aligned}$$

From the tail bound of the proposed estimator, we have,

$$\mathbb{P}\left(\frac{\Delta_a}{3} \le r_{a^\star} - \hat{r}_{\tau_k, a^\star}\right) \le \exp\left(-\frac{\Delta_a k^{1-\frac{1}{p}}}{3c} + \frac{b_p \nu_p}{2c^p}\right) \qquad \text{(F.52)}$$

Hence we can get the upper bound of the first term in (F.51). Also, by Fubini-

Tonelli theorem, we can transform the second term of (F.51) as follows

$$\begin{aligned}
&\frac{c}{k^{1-\frac{1}{p}}}\int_0^\infty \left(\int_0^g \frac{h(u)}{1-F(u)}\mathrm{d}u\right) f_{\hat{r}}\left(r_{a^\star}-\frac{c}{k^{1-\frac{1}{p}}}g-\frac{\Delta_a}{3}\right)\mathrm{d}g \\
&\qquad=\int_0^\infty \left(\int_u^\infty f_{\hat{r}}\left(r_{a^\star}-\frac{c}{k^{1-\frac{1}{p}}}g-\frac{\Delta_a}{3}\right)\frac{c}{k^{1-\frac{1}{p}}}\mathrm{d}g\right)\frac{h(u)}{1-F(u)}\mathrm{d}u \\
&\qquad=\int_0^\infty \left(\int_{-\infty}^{r_{a^\star}-\frac{c}{k^{1-\frac{1}{p}}}u-\frac{\Delta_a}{3}} f_{\hat{r}}\left(g\right)\mathrm{d}g\right)\frac{h(u)}{1-F(u)}\mathrm{d}u \\
&\qquad=\int_0^\infty \mathbb{P}\left(r_{a^\star}-\hat{r}_{\tau_k,a^\star}\geq \frac{c}{k^{1-\frac{1}{p}}}u+\frac{\Delta_a}{3}\right)\frac{h(u)}{1-F(u)}\mathrm{d}u \qquad \text{(F.53)}
\end{aligned}$$

Similar to (F.52), we have

$$\mathbb{P}\left(r_{a^\star}-\hat{r}_{\tau_k,a^\star}\geq \frac{c}{k^{1-\frac{1}{p}}}u+\frac{\Delta_a}{3}\right)\leq \exp\left(-u-\frac{\Delta_a k^{1-\frac{1}{p}}}{3c}+\frac{b_p\nu_p}{2c^p}\right)$$

Thus, we obtain the upper bound of (F.53) as follows

$$\begin{aligned}
&\int_0^\infty \mathbb{P}\left(r_{a^\star}-\hat{r}_{\tau_k,a^\star}\geq \frac{c}{k^{1-\frac{1}{p}}}u+\frac{\Delta_a}{3}\right)\frac{h(u)}{1-F(u)}\mathrm{d}u \\
&\quad\leq \int_0^\infty \exp\left(-u-\frac{\Delta_a k^{1-\frac{1}{p}}}{3c}+\frac{b_p\nu_p}{2c^p}\right)\frac{h(u)}{1-F(u)}\mathrm{d}u \\
&\quad\leq \exp\left(-\frac{\Delta_a k^{1-\frac{1}{p}}}{3c}+\frac{b_p\nu_p}{2c^p}\right)\int_0^\infty \frac{\exp\left(-u\right)h(u)}{1-F(u)}\mathrm{d}u \\
&\quad\leq C\exp\left(-\frac{\Delta_a k^{1-\frac{1}{p}}}{3c}+\frac{b_p\nu_p}{2c^p}\right),
\end{aligned}$$

where the last inequality holds due to the assumption on $F(x)$. Therefore,

$$\begin{aligned}
\int_{I_1}\rho\left(\frac{k^{1-\frac{1}{p}}}{c}\left\{r_{a^\star}-x-\frac{\Delta_a}{3}\right\}\right)\mathbb{P}(\hat{r}\in \mathrm{d}x) &\leq C\exp\left(-\frac{\Delta_a k^{1-\frac{1}{p}}}{3c}+\frac{b_p\nu_p}{2c^p}\right) \qquad \text{(F.54)}\\
&\quad+\frac{F(0)}{1-F(0)}\exp\left(-\frac{\Delta_a k^{1-\frac{1}{p}}}{3c}+\frac{b_p\nu_p}{2c^p}\right) \\
&\qquad\qquad \text{(F.55)}
\end{aligned}$$

Now we derive the upper bound of the second interval $I_2=\{r_{a^\star}-\frac{\Delta_a}{3}<x\leq r_{a^\star}-\frac{\Delta_a}{6}\}$. Since $F(0)\leq 1/2$, it is easy to see that

$$\rho\left(\frac{k^{1-\frac{1}{p}}}{c}\left\{r_{a^\star}-x-\frac{\Delta_a}{3}\right\}\right)\leq 2F\left(\frac{k^{1-\frac{1}{p}}}{c}\left\{r_{a^\star}-x-\frac{\Delta_a}{3}\right\}\right) \qquad \text{(F.56)}$$

for $x \in I_2 \cup I_3$. Hence, for $x \in I_2$,

$$\begin{aligned}
&\int_{I_2} \rho\left(\frac{k^{1-\frac{1}{p}}}{c}\left\{r_{a^\star} - x - \frac{\Delta_a}{3}\right\}\right)\mathbb{P}(\hat{r} \in \mathrm{d}x) \\
&\quad\le \int_{r_{a^\star}-\frac{\Delta_a}{3}}^{r_{a^\star}-\frac{\Delta_a}{6}} 2F\left(\frac{k^{1-\frac{1}{p}}}{c}\left\{r_{a^\star} - x - \frac{\Delta_a}{3}\right\}\right)\mathbb{P}(\hat{r} \in \mathrm{d}x) \\
&\quad\le 2\mathbb{P}\left(\frac{\Delta_a}{6} \le r_{a^\star} - \hat{r}_{\tau_k,a^\star}\right).
\end{aligned}$$

Similar to (F.52), we have

$$2\mathbb{P}\left(\frac{\Delta_a}{6} \le r_{a^\star} - \hat{r}_{\tau_k,a^\star}\right) \le 2\exp\left(-\frac{\Delta_a k^{1-\frac{1}{p}}}{6c} + \frac{b_p \nu_p}{2c^p}\right). \tag{F.57}$$

Hence, we get the upper bound of the integral on I_2 as follows,

$$\sum_{k=1}^{T} 2\exp\left(-\frac{\Delta_a k^{1-\frac{1}{p}}}{6c} + \frac{b_p \nu_p}{2c^p}\right) \le 2\exp\left(\frac{b_p \nu_p}{2c^p}\right)\Gamma\left(\frac{2p-1}{p-1}\right).$$

Finally, due to (F.56) again,

$$\begin{aligned}
&\int_{I_3} \rho\left(\frac{k^{1-\frac{1}{p}}}{c}\left\{r_{a^\star} - x - \frac{\Delta_a}{3}\right\}\right)\mathbb{P}(\hat{r} \in \mathrm{d}x) \\
&\quad\le 2\int_{r_{a^\star}-\frac{\Delta_a}{6}}^{\infty} F\left(\frac{k^{1-\frac{1}{p}}}{c}\left\{r_{a^\star} - x - \frac{\Delta_a}{3}\right\}\right)\mathbb{P}(\hat{r} \in \mathrm{d}x) \le 2F\left(-\frac{\Delta_a k^{1-\frac{1}{p}}}{6c}\right).
\end{aligned} \tag{F.58}$$

By combining (F.55), (F.57), and (F.58),

$$\begin{aligned}
\sum_{k=1}^{T} \mathbb{E}\left[\frac{Q_{\tau_k,a^\star}}{1-Q_{\tau_k,a^\star}}\middle|\mathcal{H}_{\tau_k}\right] \le \sum_{k=1}^{T}&\left\{C\exp\left(-\frac{\Delta_a k^{1-\frac{1}{p}}}{3c} + \frac{b_p \nu_p}{2c^p}\right)\right. \\
&\left.+\frac{F(0)}{1-F(0)}\exp\left(-\frac{\Delta_a k^{1-\frac{1}{p}}}{3c} + \frac{b_p \nu_p}{2c^p}\right)\right\} \\
&+\sum_{k=1}^{T} 2\exp\left(-\frac{\Delta_a k^{1-\frac{1}{p}}}{6c} + \frac{b_p \nu_p}{2c^p}\right) \\
&+\sum_{k=1}^{T} 2F\left(-\frac{k^{1-\frac{1}{p}}\Delta_a}{6c}\right)
\end{aligned}$$

$$
\begin{aligned}
&\leq \exp\left(\frac{b_p \nu_p}{2c^p}\right)\left\{C + \frac{F(0)}{1-F(0)}\right\}\Gamma\left(\frac{2p-1}{p-1}\right)\left(\frac{3c}{\Delta_a}\right)^{\frac{p}{p-1}} \\
&\quad + 2\exp\left(\frac{b_p \nu_p}{2c^p}\right)\Gamma\left(\frac{2p-1}{p-1}\right)\left(\frac{6c}{\Delta_a}\right)^{\frac{p}{p-1}} \\
&\quad + \sum_{k=1}^{T} 2F\left(-\frac{k^{1-\frac{1}{p}}\Delta_a}{6c}\right) \\
&\leq \exp\left(\frac{b_p \nu_p}{2c^p}\right)\left\{C + \frac{F(0)}{1-F(0)} + 2^{\frac{2p-1}{p-1}}\right\}\Gamma\left(\frac{2p-1}{p-1}\right)\left(\frac{3c}{\Delta_a}\right)^{\frac{p}{p-1}} \\
&\quad + \sum_{k=1}^{T} 2F\left(-\frac{k^{1-\frac{1}{p}}\Delta_a}{6c}\right).
\end{aligned}
$$

The remaining part is to derive the upper bound of the last term. For $T > 2\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}}$, let ℓ_- be the maximal time such that

$$
F\left(-\frac{\ell_-^{1-\frac{1}{p}}\Delta_a}{6c}\right) \geq \frac{1}{T}\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}}.
$$

Then, we have ℓ_- as follows,

$$
\ell_- = \left(\frac{6c}{\Delta_a}\right)^{\frac{p}{p-1}}\left\{-F^{-1}\left(\frac{1}{T}\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}}\right)\right\}^{\frac{p}{p-1}}.
$$

For $k > \ell_-$, the following inequality holds,

$$
F\left(-\frac{\ell_-^{1-\frac{1}{p}}\Delta_a}{6c}\right) < \frac{1}{T}\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}}.
$$

Note that $\frac{1}{T}\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}} \leq \frac{1}{2}$ for $T > \left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}}$ and $F^{-1}\left(\frac{1}{2T}\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}}\right) < 0$ from the assumption $F(0) < \frac{1}{2}$.

Therefore,

$$\begin{aligned}\sum_{k=1}^{T} 2F\left(-\frac{k^{1-\frac{1}{p}}\Delta_a}{6c}\right) &\leq 2\ell_- + \sum_{k=\ell_-+1}^{T} 2F\left(-\frac{k^{1-\frac{1}{p}}\Delta_a}{6c}\right)\\ &\leq 2\ell_- + \sum_{k=\ell_-+1}^{T} \frac{2}{T}\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}}\\ &\leq 2\left(\frac{6c}{\Delta_a}\right)^{\frac{p}{p-1}}\left\{-F^{-1}\left(\frac{1}{2T}\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}}\right)\right\}^{\frac{p}{p-1}} + 2\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}}\\ &\leq 2\left(\frac{6c}{\Delta_a}\right)^{\frac{p}{p-1}}\left\{-F^{-1}\left(\frac{1}{2T}\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}}\right)\right\}_+^{\frac{p}{p-1}} + 2\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}}.\end{aligned}$$

For $T \leq 2\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}}$,

$$\sum_{t=1}^{T} \mathbb{P}\left(E_{t,a}^{(2)}\right) \leq T \leq 2\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}} + 2\left(\frac{6c}{\Delta_a}\right)^{\frac{p}{p-1}}\left\{-F^{-1}\left(\frac{1}{T}\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}}\right)\right\}_+^{\frac{p}{p-1}}.$$

Thus, the upper bound also holds. By combining this upper bound, the Lemma is proved. □

Lastly, we estimate the upper bound of $E_{t,a}^{(3)}$.

Lemma 28. *Assume that the p-th moment of rewards is bounded by a constant $\nu_p < \infty$, $\hat{r}_{t,a}$ is a p-robust estimator and $F(x)$ satisfies Assumption 2. For any action $a \in \mathcal{A}$, it holds*

$$\sum_{t=1}^{T} \mathbb{P}\left(E_{t,a}^{(3)}\right) \leq \left(\frac{3c}{\Delta_a}\right)^{\frac{p}{p-1}}\left\{F^{-1}\left(1-\frac{1}{T}\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}}\right)\right\}_+^{\frac{p}{p-1}} + 2\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}}$$

Proof. Recall τ_k from Lemma 26. Obviously,

$$\sum_{t=1}^{T} \mathbb{P}\left(E_{t,a}^{(3)}\right) \leq \sum_{k=1}^{T} \mathbb{P}\left(\hat{E}_{\tau_k,a} \cap \tilde{E}_{\tau_k,a}^{c}\right)$$

Due to the decision rule of the perturbation method and the definition of τ_k, observe that $n_{\tau_k,a} = k$ and $\beta_{\tau_k,a} = \frac{c}{k^{1-\frac{1}{p}}}$. By the conditioning on $\mathcal{H}_{\tau_k}$,

$$\begin{aligned}\mathbb{P}\left(\hat{E}_{\tau_k,a} \cap \tilde{E}^c_{\tau_k,a} \middle| \mathcal{H}_{\tau_k}\right) &\leq \mathbb{P}\left(\hat{r}_{\tau_k} \leq x_a, G_{\tau_k,a} > \frac{y_a - \hat{r}_{\tau_k,a}}{\beta_{\tau_k,a}} \middle| \mathcal{H}_{\tau_k}\right) \\ &\leq \mathbb{P}\left(G_{\tau_k,a} > \frac{y_a - x_a}{\beta_{\tau_k,a}} \middle| \mathcal{H}_{\tau_k}\right) \\ &= \mathbb{P}\left(G_{\tau_k,a} > \frac{\Delta_a k^{1-\frac{1}{p}}}{3c} \middle| \mathcal{H}_{\tau_k}\right) = 1 - F\left(\frac{\Delta_a k^{1-\frac{1}{p}}}{3c}\right).\end{aligned} \tag{F.59}$$

We first show that the bound holds for $T > \left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}}$ and check the case of $T \leq \left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}}$.

For $T > 2\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}}$, let ℓ_+ be the maximal time such as

$$F\left(\frac{\Delta_a \ell^{1-\frac{1}{p}}}{3c}\right) \leq 1 - \frac{1}{T}\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}}.$$

There exists a positive ℓ_+ since $1 - \frac{1}{T}\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}} > \frac{1}{2}$ and the assumption $F(0) < \frac{1}{2}$. Note that

$$\ell_+ \leq \left(\frac{3c}{\Delta_a}\right)^{\frac{p}{p-1}} \left\{F^{-1}\left(1 - \frac{1}{T}\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}}\right)\right\}^{\frac{p}{p-1}}. \tag{F.60}$$

and for $k > \ell_+$

$$1 - F\left(\frac{\Delta_a k^{1-\frac{1}{p}}}{3c}\right) \leq \frac{1}{T}\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}}. \tag{F.61}$$

Therefore, by (F.59), (F.60), and (F.61),

$$\begin{aligned}\sum_{k=1}^{T}\mathbb{P}\left(\hat{E}_{\tau_k,a}\cap\tilde{E}^c_{\tau_k,a}\right) &\le \sum_{k=1}^{T}\left(1-F\left(\frac{\Delta_a k^{1-\frac{1}{p}}}{3c}\right)\right)\\ &\le \ell_+ + \sum_{k=\ell_++1}^{T}\left(1-F\left(\frac{\Delta_a k^{1-\frac{1}{p}}}{3c}\right)\right)\\ &\le \left(\frac{3c}{\Delta_a}\right)^{\frac{p}{p-1}}\left\{F^{-1}\left(1-\frac{1}{T}\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}}\right)\right\}^{\frac{p}{p-1}} + \sum_{k=\ell+1}^{T}\frac{1}{T}\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}}\\ &\le \left(\frac{3c}{\Delta_a}\right)^{\frac{p}{p-1}}\left\{F^{-1}\left(1-\frac{1}{T}\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}}\right)\right\}_+^{\frac{p}{p-1}} + 2\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}}.\end{aligned}$$

For $T \le 2\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}}$,

$$\sum_{t=1}^{T}\mathbb{P}\left(E^{(3)}_{t,a}\right) \le T \le 2\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}} + \left(\frac{3c}{\Delta_a}\right)^{\frac{p}{p-1}}\left\{F^{-1}\left(1-\frac{1}{T}\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}}\right)\right\}_+^{\frac{p}{p-1}}.$$

Thus, the bound also holds. Consequently, the lemma is proved. □

Proof of Theorem 30. Recall the definition of regret $\mathcal{R}_T$, and the fact $\mathbb{P}(a_t = a) = \mathbb{P}(E_{t,a}) = \sum_{i=1}^{3}\mathbb{P}(E^{(i)}_{t,a})$. Hence

$$\mathbb{E}[\mathcal{R}_T] := \sum_{a\in\mathcal{A}}\sum_{t=1}^{T}\Delta_a\mathbb{P}\left(a_t = a\right) = \sum_{a\neq a^\star}\sum_{i=1}^{3}\sum_{t=1}^{T}\Delta_a\mathbb{P}\left(E^{(i)}_{t,a}\right) \tag{F.62}$$

By Lemmas 26, 27, and 28,

$$\sum_{t=1}^{T}\Delta_a\mathbb{P}\left(E^{(1)}_{t,a}\right) \le \Delta_a + \exp\left(\frac{b_p\nu_p}{2c^p}\right)\left(\frac{(3c)^p}{\Delta_a}\right)^{\frac{1}{p-1}}\Gamma\left(\frac{2p-1}{p-1}\right).$$

$$\begin{aligned}\sum_{t=1}^{T}\Delta_a\mathbb{P}\left(E^{(2)}_{t,a}\right) \le \exp\left(\frac{b_p\nu_p}{2c^p}\right)&\left\{C+\frac{F(0)}{1-F(0)}+2^{\frac{2p-1}{p-1}}\right\}\Gamma\left(\frac{2p-1}{p-1}\right)\left(\frac{(3c)^p}{\Delta_a}\right)^{\frac{1}{p-1}}\\ &+2\left(\frac{(6c)^p}{\Delta_a}\right)^{\frac{1}{p-1}}\left\{-F^{-1}\left(\frac{1}{T}\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}}\right)\right\}_+^{\frac{p}{p-1}} + 2\left(\frac{c^p}{\Delta_a}\right)^{\frac{1}{p-1}}\end{aligned}$$

$$\sum_{t=1}^{T}\Delta_a\mathbb{P}\left(E_{t,a}^{(3)}\right)\leq\left(\frac{(3c)^p}{\Delta_a}\right)^{\frac{1}{p-1}}\left\{F^{-1}\left(1-\frac{1}{T}\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}}\right)\right\}_+^{\frac{p}{p-1}}+2\left(\frac{c^p}{\Delta_a}\right)^{\frac{1}{p-1}}$$

Therefore, we can estimate the upper bound of (F.62) by combining the above results as follows

$$\begin{aligned}\mathbb{E}[\mathcal{R}_T]\leq\sum_{a\neq a^\star}&\left[\exp\left(\frac{b_p\nu_p}{2c^p}\right)\left\{C+\frac{F(0)}{1-F(0)}+2^{\frac{2p-1}{p-1}}+1\right\}\Gamma\left(\frac{2p-1}{p-1}\right)\left(\frac{(3c)^p}{\Delta_a}\right)^{\frac{1}{p-1}}\right.\\&+2\left(\frac{(6c)^p}{\Delta_a}\right)^{\frac{1}{p-1}}\left\{-F^{-1}\left(\frac{1}{T}\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}}\right)\right\}_+^{\frac{p}{p-1}}\\&+\left(\frac{(3c)^p}{\Delta_a}\right)^{\frac{1}{p-1}}\left\{F^{-1}\left(1-\frac{1}{T}\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}}\right)\right\}_+^{\frac{p}{p-1}}\\&\left.+4\left(\frac{c^p}{\Delta_a}\right)^{\frac{1}{p-1}}+\Delta_a\right]\\\leq O&\left(\sum_{a\neq a^\star}\frac{C_{c,p,\nu_p,F}}{\Delta_a^{\frac{1}{p-1}}}+\frac{(6c)^{\frac{p}{p-1}}}{\Delta_a^{\frac{1}{p-1}}}\left[-F^{-1}\left(\frac{c^{\frac{p}{p-1}}}{T\Delta_a^{\frac{p}{p-1}}}\right)\right]_+^{\frac{p}{p-1}}\right.\\&\left.+\frac{(3c)^{\frac{p}{p-1}}}{\Delta_a^{\frac{1}{p-1}}}\left[F^{-1}\left(1-\frac{c^{\frac{p}{p-1}}}{T\Delta_a^{\frac{p}{p-1}}}\right)\right]_+^{\frac{p}{p-1}}+\Delta_a\right)\end{aligned}$$

The theorem is proved. □

F.4 General Regret Lower Bound of APE^2

Proof of Theorem 31. We construct a K-armed multi-armed bandit problem with deterministic rewards to prove the regret lower bound of APE^2. Let the optimal arm $a^\star$ give the reward of $\Delta=\frac{1}{2}c^{\frac{1}{p}}\left(\frac{(K-1)}{T}\right)^{1-\frac{1}{p}}F^{-1}\left(1-\frac{1}{K}\right)$ whereas the other arms provide zero rewards. Note that $\Delta\in[0,1]$ for

$$T\geq\frac{c^{\frac{1}{p-1}}(K-1)}{2^{\frac{p}{p-1}}}\left|F^{-1}\left(1-\frac{1}{K}\right)\right|^{\frac{p}{p-1}}$$

and the estimator becomes $\hat{r}_a = \Delta\mathbb{I}[a = a^\star]$ since there is no noise. Let E_t be the set of events which satisfy

$$\sum_{a\neq a^\star} n_{t,a} \leq cT$$

If $\mathbb{P}(E_t) \leq 1/2$ holds for some $t \in [1, \cdots, T]$, then the regret bound is computed as follows

$$\mathbb{E}[\mathcal{R}_T] \geq \frac{1}{2}\mathbb{E}[\mathcal{R}_t|E_t^c] \geq \frac{cT}{2}\Delta = \frac{c^{1+\frac{1}{p}}}{4}(K-1)^{1-\frac{1}{p}}T^{\frac{1}{p}}F^{-1}\left(1-\frac{1}{K}\right)$$

hence it satisfies the lower bound. Otherwise, if $\mathbb{P}(E_t) > 1/2$ holds for all $t \in [1, \cdots, T]$, it is sufficient to prove $\mathbb{P}(a_t \neq a^\star) \geq 1/8$. Then, it holds

$$\mathbb{E}[\mathcal{R}_T] = \sum_{t=1}^{T} \Delta\mathbb{P}(a_t = a^\star) \geq \frac{T}{8}\Delta = \frac{c^{\frac{1}{p}}}{16}(K-1)^{1-\frac{1}{p}}T^{\frac{1}{p}}F^{-1}\left(1-\frac{1}{K}\right)$$

and we get the desired result since $0 < c < \frac{K-1}{K-1+2^{\frac{p}{p-1}}}$.

Now, the remaining part is to prove that $\mathbb{P}(a_t \neq a^\star) \geq 1/8$ holds. First, we observe that

$$\begin{aligned}
\mathbb{P}(a_t \neq a^\star) &= \mathbb{P}\left(\bigcup_{a\neq a^\star} \{\hat{r}_{a^\star} + \beta_{t,a^\star}G_{t,a^\star} \leq \hat{r}_a + \beta_{t,a}G_{t,a}\}\right) \\
&\geq \mathbb{P}\left(E_{t-1}\right)\mathbb{P}\left(\bigcup_{a\neq a^\star} \{\hat{r}_{a^\star} + \beta_{t,a^\star}G_{t,a^\star} \leq 2\Delta \leq \hat{r}_a + \beta_{t,a}G_{t,a}\}\middle|E_{t-1}\right) \\
&\geq \frac{1}{2}\mathbb{E}\Bigg[\mathbb{P}\left(G_{t,a^\star} \leq \frac{\Delta}{\beta_{t,a^\star}}\middle|\mathcal{H}_{t-1}, E_{t-1}\right) \\
&\qquad \times \mathbb{P}\left(\bigcup_{a\neq a^\star} \{2\Delta \leq \beta_{t,a}G_{t,a}\}\middle|\mathcal{H}_{t-1}, E_{t-1}\right)\Bigg|E_{t-1}\Bigg] \\
&\geq \frac{1}{2}\mathbb{E}\Bigg[\mathbb{P}\left(G_{t,a^\star} \leq \frac{\Delta\left((1-c)T\right)^{1-\frac{1}{p}}}{c}\middle|\mathcal{H}_{t-1}, E_{t-1}\right) \\
&\qquad \times \mathbb{P}\left(\bigcup_{a\neq a^\star} \{2\Delta \leq \beta_{t,a}G_{t,a}\}\middle|\mathcal{H}_{t-1}, E_{t-1}\right)\Bigg|E_{t-1}\Bigg]
\end{aligned}$$

where the last inequality holds due to $n_{t-1,a^\star} \geq (1-c)T$ provided E_{t-1}. Since $c < \frac{K-1}{K-1+2^{\frac{p}{p-1}}}$, we have,

$$\frac{\Delta\left((1-c)T\right)^{1-\frac{1}{p}}}{c} = \left(\frac{(1-c)(K-1)}{2^{\frac{p}{p-1}}c}\right)^{1-\frac{1}{p}} F^{-1}\left(1-\frac{1}{K}\right) > F^{-1}\left(1-\frac{1}{K}\right).$$

Hence, $\mathbb{P}\left(G_{t,a^\star} \leq \frac{\Delta((1-c)T)^{1-\frac{1}{p}}}{c} \middle| \mathcal{H}_{t-1}, E_{t-1}\right) \geq 1-\frac{1}{K}$ so that

$$\mathbb{P}(a_t \neq a^\star) \geq \frac{1}{2}\left(1-\frac{1}{K}\right)\mathbb{E}\left[\mathbb{P}\left(\bigcup_{a\neq a^\star}\{2\Delta \leq \beta_{t,a}G_{t,a}\} \middle| \mathcal{H}_{t-1}, E_{t-1}\right)\middle| E_{t-1}\right].$$

Observe that

$$\begin{aligned}
&\mathbb{P}\left(\bigcup_{a\neq a^\star}\{2\Delta \leq \beta_{t,a}G_{t,a}\} \middle| \mathcal{H}_{t-1}, E_{t-1}\right)\\
&\geq 1-\mathbb{P}\left(\bigcap_{a\neq a^\star}\left\{G_{t,a} \leq \frac{2\Delta}{\beta_{t,a}}\right\} \middle| \mathcal{H}_{t-1}, E_{t-1}\right)\\
&\geq 1-\prod_{a\neq a^\star} F\left(\frac{2\Delta\left(n_{t-1,a}\right)^{1-\frac{1}{p}}}{c}\right)\\
&\geq 1-\left|F\left(2\Delta\frac{\sum_{a\neq a^\star}\left(n_{t-1,a}\right)^{1-\frac{1}{p}}}{c(K-1)}\right)\right|^{K-1},
\end{aligned}$$

where the last inequality holds by the log-concavity of F. Under E_{t-1}, note that

$$\sum_{a\neq a^\star}\left(n_{t-1,a}\right)^{1-\frac{1}{p}} \leq \left(\sum_{a\neq a^\star}1^p\right)^{\frac{1}{p}}\left(\sum_{a\neq a^\star}n_{t-1,a}\right)^{1-\frac{1}{p}} \leq (K-1)^{\frac{1}{p}}\left(cT\right)^{1-\frac{1}{p}}$$

which implies

$$F\left(2\Delta\frac{\sum_{a\neq a^\star}\left(n_{t-1,a}\right)^{1-\frac{1}{p}}}{c(K-1)}\right) \leq F\left(2\Delta c^{-\frac{1}{p}}\left(\frac{T}{(K-1)}\right)^{1-\frac{1}{p}}\right) = 1-\frac{1}{K}$$

Therefore, we get

$$\mathbb{P}(a_t \neq a^\star) \geq \frac{1}{2}\left(1-\frac{1}{K}\right)\left(1-\left(1-\frac{1}{K}\right)^{K-1}\right) \geq \frac{1}{8}$$

since $1-\frac{1}{K} \geq \frac{1}{2}$ and $1-\left(1-\frac{1}{K}\right)^{K-1} \geq \frac{1}{2}$ hold for $K \geq 2$ and the theorem is proved. □

F.5 Proofs of Corollaries

Proof of Corollary 6. The CDF of a Weibull distribution with $k \leq 1$ is given as

$$F(x) = 1 - \exp\left(-\left(\frac{x}{\lambda}\right)^k\right)$$

Then, its inverse is

$$F^{-1}(y) = \lambda\left[\ln\left(\frac{1}{1-y}\right)\right]^{\frac{1}{k}},$$

Then,

$$F^{-1}\left(1 - \frac{c^{\frac{p}{p-1}}}{T\Delta_a^{\frac{p}{p-1}}}\right)^{\frac{p}{p-1}} = \lambda^{\frac{p}{p-1}}\left[\ln\left(\frac{T\Delta_a^{\frac{p}{p-1}}}{c^{\frac{p}{p-1}}}\right)\right]^{\frac{p}{k(p-1)}}.$$

Thus, we compute C as follows,

$$\begin{aligned}
\int_0^\infty \frac{h(z)\exp(-z)}{1-F(z)}dz &= \int_0^\infty \frac{k}{\lambda}\left(\frac{z}{\lambda}\right)^{k-1}\frac{\exp\left(-\left(\frac{z}{\lambda}\right)^k\right)\exp(-z)}{\exp\left(-2\left(\frac{z}{\lambda}\right)^k\right)}dz \\
&= \int_0^\infty \frac{k}{\lambda}\left(\frac{z}{\lambda}\right)^{k-1}\exp\left(-z+\left(\frac{z}{\lambda}\right)^k\right)dz \\
&\leq \int_0^\infty \frac{k}{\lambda}\left(\frac{z}{\lambda}\right)^{k-1}\exp\left(-\frac{\lambda-1}{\lambda}z\right)dz \\
&= \frac{k}{(\lambda-1)^k}\int_0^\infty z^{k-1}\exp(-z)\,dz \\
&= \frac{k\Gamma(k)}{(\lambda-1)^k} = \frac{\Gamma(k+1)}{(\lambda-1)^k} \\
&\leq \frac{\Gamma(2)}{(\lambda-1)^k} = (\lambda-1)^{-k}.
\end{aligned}$$

For $\frac{(6c)^{\frac{p}{p-1}}}{\Delta_a^{\frac{1}{p-1}}}\left[-F^{-1}\left(\frac{c^{\frac{p}{p-1}}}{T\Delta_a^{\frac{p}{p-1}}}\right)\right]_+^{\frac{p}{p-1}}$, we have,

$$\frac{(6c)^{\frac{p}{p-1}}}{\Delta_a^{\frac{1}{p-1}}}\left[-F^{-1}\left(\frac{c^{\frac{p}{p-1}}}{T\Delta_a^{\frac{p}{p-1}}}\right)\right]_+^{\frac{p}{p-1}} = 0$$

since the support of x is $(0, \infty)$. Then, the problem dependent regret bound becomes,

$$\mathbb{E}\left[\mathcal{R}_T\right] \leq \sum_{a \neq a^\star} \left[\exp\left(\frac{b_p \nu_p}{2c^p}\right) \left\{ C_1 + \frac{F(0)}{1-F(0)} + 2^{\frac{2p-1}{p-1}} + 1 \right\} \right. \tag{F.63}$$

$$\times \Gamma\left(\frac{2p-1}{p-1}\right) \left(\frac{(3c)^p}{\Delta_a}\right)^{\frac{1}{p-1}} \tag{F.64}$$

$$+ \frac{(6c)^{\frac{p}{p-1}}}{\Delta_a^{\frac{1}{p-1}}} \left[-F^{-1}\left(\frac{c^{\frac{p}{p-1}}}{T\Delta_a^{\frac{p}{p-1}}} \right) \right]_+^{\frac{p}{p-1}} \tag{F.65}$$

$$+ \left(\frac{(3c)^p}{\Delta_a}\right)^{\frac{1}{p-1}} \left\{ F^{-1}\left(1 - \frac{1}{T}\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}} \right) \right\}^{\frac{p}{p-1}} \tag{F.66}$$

$$\left. + \left(\frac{c^p}{\Delta_a}\right)^{\frac{1}{p-1}} + \Delta_a \right] \tag{F.67}$$

$$\leq \sum_{a \neq a^\star} \left[\exp\left(\frac{b_p \nu_p}{2c^p}\right) \left[(\lambda - 1)^{-k} + 2^{\frac{2p-1}{p-1}} + 1 \right] \Gamma\left(\frac{2p-1}{p-1}\right) \left(\frac{(3c)^p}{\Delta_a}\right)^{\frac{1}{p-1}} \right. \tag{F.68}$$

$$\left. + \left(\frac{(3c\lambda)^p}{\Delta_a}\right)^{\frac{1}{p-1}} \left[\ln\left(\frac{T\Delta_a^{\frac{p}{p-1}}}{c^{\frac{p}{p-1}}} \right) \right]^{\frac{p}{k(p-1)}} + \left(\frac{c^p}{\Delta_a}\right)^{\frac{1}{p-1}} + \Delta_a \right] \tag{F.69}$$

$$\leq O\left(\sum_{a \neq a^\star} \frac{C_{c,p,\nu_p,F}}{\Delta_a^{\frac{1}{p-1}}} + \left(\frac{(3c\lambda)^p}{\Delta_a}\right)^{\frac{1}{p-1}} \left[\ln\left(\frac{T\Delta_a^{\frac{p}{p-1}}}{c^{\frac{p}{p-1}}} \right) \right]^{\frac{p}{k(p-1)}} + \Delta_a \right). \tag{F.70}$$

The problem independent regret bound can be obtained by choosing the threshold of the minimum gap as $\Delta = c\,(K/T)^{1-\frac{1}{p}} \ln(K)^{\frac{1}{k}}$.

$$\mathbb{E}\left[\mathcal{R}_T\right] \leq \sum_{a \neq a^\star, \Delta_a > \Delta} \frac{C_{c,p,\nu_p,F}}{\Delta_a^{\frac{1}{p-1}}} + \left(\frac{(3c\lambda)^p}{\Delta_a}\right)^{\frac{1}{p-1}} \left[\ln\left(\frac{T\Delta_a^{\frac{p}{p-1}}}{c^{\frac{p}{p-1}}} \right) \right]^{\frac{p}{k(p-1)}} + \Delta T \tag{F.71}$$

$$\leq K\left(\frac{C_{c,p,\nu_p,F}}{\Delta^{\frac{1}{p-1}}} + \left(\frac{(3c\lambda)^p}{\Delta}\right)^{\frac{1}{p-1}} \left[\ln\left(\frac{T\Delta^{\frac{p}{p-1}}}{c^{\frac{p}{p-1}}} \right) \right]^{\frac{p}{k(p-1)}} \right) + \Delta T \tag{F.72}$$

$$\leq K\frac{C_{c,p,\nu_p,F}\cdot T^{\frac{1}{p}}}{K^{\frac{1}{p}}\ln(K)^{\frac{1}{k(p-1)}}}+K\left(\frac{(3\lambda)^{\frac{p}{p-1}}cT^{\frac{1}{p}}}{K^{\frac{1}{p}}\ln(K)^{\frac{1}{k(p-1)}}}\right)\left[\ln\left(K\ln(K)^{\frac{p}{k(p-1)}}\right)\right]^{\frac{p}{k(p-1)}} \tag{F.73}$$

$$+cK^{1-\frac{1}{p}}T^{\frac{1}{p}}\ln(K)^{\frac{1}{k}} \tag{F.74}$$

$$\leq\frac{C_{c,p,\nu_p,F}\cdot K^{1-\frac{1}{p}}T^{\frac{1}{p}}}{\ln(K)^{\frac{1}{k(p-1)}}}+c(3\lambda)^{\frac{p}{p-1}}K^{1-\frac{1}{p}}T^{\frac{1}{p}}\left(\frac{\left[\left(1+\frac{p}{k(p-1)}\right)\ln(K)\right]^{\frac{p}{k(p-1)}}}{\ln(K)^{\frac{1}{k(p-1)}}}\right) \tag{F.75}$$

$$+cK^{1-\frac{1}{p}}T^{\frac{1}{p}}\ln(K)^{\frac{1}{k}} \tag{F.76}$$

$$\leq O\left((c\lambda)^{\frac{p}{p-1}}K^{1-\frac{1}{p}}T^{\frac{1}{p}}\left(\frac{\ln(K)^{\frac{p}{k(p-1)}}}{\ln(K)^{\frac{1}{k(p-1)}}}\right)\right)=O\left((c\lambda)^{\frac{p}{p-1}}K^{1-\frac{1}{p}}T^{\frac{1}{p}}\ln(K)^{\frac{1}{k}}\right). \tag{F.77}$$

Consequently, the lower bound is simply obtained by the general lower bound, so we can conclude that regret bound is tight. The corollary is proved. □

Proof of Corollary 7. The CDF of a generalized extreme value distribution with $0\leq\zeta<1$ is given as

$$F(x)=\exp\left(-\left(1+\zeta\frac{x}{\lambda}\right)^{-1/\zeta}\right).$$

Then, its inverse is

$$F^{-1}(y)=\lambda\frac{[\ln(1/y)]^{-\zeta}-1}{\zeta}\leq\lambda\frac{[1-y]^{-\zeta}-1}{\zeta},$$

and

$$\lambda\frac{[\ln(1/y)]^{-\zeta}-1}{\zeta}\geq\lambda\frac{\left[\frac{y}{1-y}\right]^{\zeta}-1}{\zeta}$$

where $\ln(x)\leq x-1$ is used. Then,

$$\left[F^{-1}\left(1-\frac{c^{\frac{p}{p-1}}}{T\Delta_a^{\frac{p}{p-1}}}\right)\right]^{\frac{p}{p-1}}\leq\lambda^{\frac{p}{p-1}}\left[\frac{\left(T\Delta_a^{\frac{p}{p-1}}/c^{\frac{p}{p-1}}\right)^{\zeta}-1}{\zeta}\right]^{\frac{p}{p-1}}.$$

We compute the $\sup h$ can be obtained as follows,

$$\begin{aligned}\sup h = &\sup_{x\in[0,\infty]} \frac{\left(1+\zeta\frac{x}{\lambda}\right)^{-1/\zeta-1}\exp\left(-\left(1+\zeta\frac{x}{\lambda}\right)^{-1/\zeta}\right)}{\lambda\left(1-\exp\left(-\left(1+\zeta\frac{x}{\lambda}\right)^{-1/\zeta}\right)\right)}\\ =&\sup_{t\in[0,1]}\frac{t^{\zeta+1}\exp(-t)}{\lambda(1-\exp(-t))} \leq \sup_{t\in[0,1]}\frac{t\exp(-t)}{\lambda(1-\exp(-t))} = \frac{1}{\lambda}.\end{aligned}$$

M can be obtained as,

$$\begin{aligned}\int_0^\infty \frac{\exp\left(-z\right)}{1-F\left(z\right)}dz = &\int_0^\infty \frac{\exp\left(-z\right)}{1-\exp\left(-\left(1+\zeta\frac{z}{\lambda}\right)^{-1/\zeta}\right)}dz\\ \leq &\int_0^\infty \left(1+\left(1+\zeta\frac{z}{\lambda}\right)^{1/\zeta}\right)\exp\left(-z\right)dz\\ =&1+\int_0^\infty \left(1+\zeta\frac{z}{\lambda}\right)^{1/\zeta}\exp\left(-z\right)dz\\ \leq&1+\int_0^\infty \exp\left(-z+\frac{\ln(1+\zeta\frac{z}{\lambda})}{\zeta}\right)dz\\ \leq&1+\int_0^\infty \exp\left(-z+\frac{z}{\lambda}\right)dz\\ =&1+\frac{\lambda}{\lambda-1} \ \because\ \lambda>1\\ =&\frac{2\lambda-1}{\lambda-1} =: M_1.\end{aligned}$$

Hence, $\sup h\cdot M_1 \leq \frac{2\lambda-1}{\lambda(\lambda-1)} \leq \frac{2}{\lambda-1}$.

For $\frac{(6c)^{\frac{p}{p-1}}}{\Delta_a^{\frac{1}{p-1}}}\left[-F^{-1}\left(\frac{c^{\frac{p}{p-1}}}{T\Delta_a^{\frac{p}{p-1}}}\right)\right]_+^{\frac{p}{p-1}}$, we have,

$$\begin{aligned}
\frac{(6c)^{\frac{p}{p-1}}}{\Delta_a^{\frac{1}{p-1}}}\left[-F^{-1}\left(\frac{c^{\frac{p}{p-1}}}{T\Delta_a^{\frac{p}{p-1}}}\right)\right]_+^{\frac{p}{p-1}} &= \frac{(6c\lambda)^{\frac{p}{p-1}}}{\Delta_a^{\frac{1}{p-1}}}\left[\frac{1-\ln\left(\frac{T\Delta_a^{\frac{p}{p-1}}}{c^{\frac{p}{p-1}}}\right)^{-\zeta}}{\zeta}\right]^{\frac{p}{p-1}} \\
&\le \frac{(6c\lambda)^{\frac{p}{p-1}}}{\Delta_a^{\frac{1}{p-1}}}\left[\frac{\ln\left(\frac{T\Delta_a^{\frac{p}{p-1}}}{c^{\frac{p}{p-1}}}\right)^{\zeta}-1}{\zeta}\right]^{\frac{p}{p-1}} \\
&\le \frac{(6c\lambda)^{\frac{p}{p-1}}}{\Delta_a^{\frac{1}{p-1}}}\left[\frac{\left(\frac{T\Delta_a^{\frac{p}{p-1}}}{c^{\frac{p}{p-1}}}\right)^{\zeta}-1}{\zeta}\right]^{\frac{p}{p-1}} \\
&\le \frac{(6c\lambda)^{\frac{p}{p-1}}}{\Delta_a^{\frac{1}{p-1}}}\ln_\zeta\left(\frac{T\Delta_a^{\frac{p}{p-1}}}{c^{\frac{p}{p-1}}}\right)^{\frac{p}{p-1}},
\end{aligned}$$

where $-\ln_\zeta(1/\ln(x)) \le \ln_\zeta(\ln(x)) \le \ln_\zeta(x)$ is used.

Then, the problem dependent regret bound becomes,

$$\mathbb{E}\left[\mathcal{R}_T\right] \le \sum_{a\ne a^\star}\left[\exp\left(\frac{b_p\nu_p}{2c^p}\right)\left\{\|h\|_\infty M + \frac{F(0)}{1-F(0)} + 2^{\frac{2p-1}{p-1}} + 1\right\}\right. \tag{F.78}$$

$$\times\,\Gamma\left(\frac{2p-1}{p-1}\right)\left(\frac{(3c)^p}{\Delta_a}\right)^{\frac{1}{p-1}} \tag{F.79}$$

$$+\frac{(6c)^{\frac{p}{p-1}}}{\Delta_a^{\frac{1}{p-1}}}\left[-F^{-1}\left(\frac{c^{\frac{p}{p-1}}}{T\Delta_a^{\frac{p}{p-1}}}\right)\right]_+^{\frac{p}{p-1}} \tag{F.80}$$

$$+\left(\frac{(3c)^p}{\Delta_a}\right)^{\frac{1}{p-1}}\left[F^{-1}\left(1-\frac{1}{T}\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}}\right)\right]_+^{\frac{p}{p-1}} \tag{F.81}$$

$$\left.+\left(\frac{c^p}{\Delta_a}\right)^{\frac{1}{p-1}} + \Delta_a\right] \tag{F.82}$$

$$\le \sum_{a\ne a^\star}\left[\exp\left(\frac{b_p\nu_p}{2c^p}\right)\left[\frac{2}{\lambda-1} + \frac{e}{e-1} + 2^{\frac{2p-1}{p-1}} + 1\right]\right. \tag{F.83}$$

$$\times \Gamma\left(\frac{2p-1}{p-1}\right)\left(\frac{(3c)^p}{\Delta_a}\right)^{\frac{1}{p-1}} \tag{F.84}$$

$$+\frac{(6c\lambda)^{\frac{p}{p-1}}}{\Delta_a^{\frac{1}{p-1}}}\ln_\zeta\left(\frac{T\Delta_a^{\frac{p}{p-1}}}{c^{\frac{p}{p-1}}}\right)^{\frac{p}{p-1}} + \left(\frac{(3c\lambda)^p}{\Delta_a}\right)^{\frac{1}{p-1}}\ln_\zeta\left(\frac{T\Delta_a^{\frac{p}{p-1}}}{c^{\frac{p}{p-1}}}\right)^{\frac{p}{p-1}} \tag{F.85}$$

$$+\left(\frac{c^p}{\Delta_a}\right)^{\frac{1}{p-1}}+\Delta_a\Bigg] \tag{F.86}$$

$$\leq O\left(\sum_{a\neq a^\star}\frac{C_{c,p,\nu_p,F}}{\Delta_a^{\frac{1}{p-1}}}+2\left(\frac{(6c\lambda)^p}{\Delta_a}\right)^{\frac{1}{p-1}}\ln_\zeta\left(\frac{T\Delta_a^{\frac{p}{p-1}}}{c^{\frac{p}{p-1}}}\right)^{\frac{p}{p-1}}+\Delta_a\right), \tag{F.87}$$

where $\ln_\zeta(x) := \frac{x^\zeta - 1}{\zeta}$.

The problem independent regret bound can be obtained by choosing the threshold of the minimum gap as $\Delta = c\left(\frac{K}{T}\right)^{1-\frac{1}{p}}\ln_\zeta(K)$ Note that $\lim_{\zeta\to 0}\frac{x^\zeta-1}{\zeta} = \ln(x)$

$$\mathbb{E}\left[\mathcal{R}_T\right] \leq \sum_{\Delta_a>\Delta}\Bigg[\exp\left(\frac{b_p\nu_p}{2c^p}\right)\left[\frac{\lambda+1}{\lambda-1}+\frac{e}{e-1}+2^{\frac{2p-1}{p-1}}\right]\Gamma\left(\frac{2p-1}{p-1}\right)\left(\frac{(3c)^p}{\Delta_a}\right)^{\frac{1}{p-1}} \tag{F.88}$$

$$+2\left(\frac{(6c\lambda)^p}{\Delta_a}\right)^{\frac{1}{p-1}}\ln_\zeta\left(\frac{T\Delta_a^{\frac{p}{p-1}}}{c^{\frac{p}{p-1}}}\right)^{\frac{p}{p-1}} \tag{F.89}$$

$$+\left(\frac{c^p}{\Delta_a}\right)^{\frac{1}{p-1}}\Bigg]+\Delta T \tag{F.90}$$

$$\leq K\Bigg[\exp\left(\frac{b_p\nu_p}{2c^p}\right)\left[\frac{\lambda+1}{\lambda-1}+\frac{e}{e-1}+2^{\frac{2p-1}{p-1}}\right]\Gamma\left(\frac{2p-1}{p-1}\right)\left(\frac{(3c)^p}{\Delta}\right)^{\frac{1}{p-1}} \tag{F.91}$$

$$+2\left(\frac{(6c\lambda)^p}{\Delta}\right)^{\frac{1}{p-1}}\ln_\zeta\left(\frac{T\Delta^{\frac{p}{p-1}}}{c^{\frac{p}{p-1}}}\right)^{\frac{p}{p-1}} \tag{F.92}$$

$$+\left(\frac{c^p}{\Delta}\right)^{\frac{1}{p-1}}\Bigg]+\Delta T \tag{F.93}$$

$$\leq \exp\left(\frac{b_p \nu_p}{2c^p}\right)\left[\frac{\lambda+1}{\lambda-1}+\frac{e}{e-1}+2^{\frac{2p-1}{p-1}}\right]\Gamma\left(\frac{2p-1}{p-1}\right)(3\lambda)^{\frac{p}{p-1}}\frac{cK^{1-\frac{1}{p}}T^{\frac{1}{p}}}{\ln_\zeta(K)^{\frac{1}{p-1}}} \tag{F.94}$$

$$+\,2(6\lambda)^{\frac{p}{p-1}}cK^{1-\frac{1}{p}}T^{\frac{1}{p}}\left(\frac{\ln_\zeta\left(K\ln_\zeta(K)^{\frac{p}{p-1}}\right)^{\frac{p}{p-1}}}{\ln_\zeta(K)^{\frac{1}{p-1}}}\right) \tag{F.95}$$

$$+\,c\frac{K^{1-\frac{1}{p}}T^{\frac{1}{p}}}{\ln_\zeta(K)^{\frac{1}{p-1}}}+cK^{1-\frac{1}{p}}T^{\frac{1}{p}}\ln_\zeta(K) \tag{F.96}$$

$$\leq \exp\left(\frac{b_p \nu_p}{2c^p}\right)\left[\frac{\lambda+1}{\lambda-1}+\frac{e}{e-1}+2^{\frac{2p-1}{p-1}}\right]\Gamma\left(\frac{2p-1}{p-1}\right)(3\lambda)^{\frac{p}{p-1}}\frac{cK^{1-\frac{1}{p}}T^{\frac{1}{p}}}{\ln_\zeta(K)^{\frac{1}{p-1}}} \tag{F.97}$$

$$+\,2(6\lambda)^{\frac{p}{p-1}}cK^{1-\frac{1}{p}}T^{\frac{1}{p}}\left(\frac{\ln_\zeta\left(K^{\frac{2p-1}{p-1}}\right)^{\frac{p}{p-1}}}{\ln_\zeta(K)^{\frac{1}{p-1}}}\right) \tag{F.98}$$

$$+\,c\frac{K^{1-\frac{1}{p}}T^{\frac{1}{p}}}{\ln_\zeta(K)^{\frac{1}{p-1}}}+cK^{1-\frac{1}{p}}T^{\frac{1}{p}}\ln_\zeta(K) \tag{F.99}$$

$$\because \ln_\zeta(x\ln_\zeta(x)^{\frac{p}{p-1}}) \leq \ln_\zeta\left(x^{1+\frac{p}{p-1}}\right) \text{ for } x>2 \tag{F.100}$$

$$\leq O\left(K^{1-\frac{1}{p}}T^{\frac{1}{p}}\frac{\ln_\zeta\left(K^{\frac{2p-1}{p-1}}\right)^{\frac{p}{p-1}}}{\ln_\zeta(K)^{\frac{1}{p-1}}}\right). \tag{F.101}$$

For the lower bound,

$$\lambda\frac{\left[\ln\left(\frac{1}{1-\frac{1}{K}}\right)\right]^{-\zeta}-1}{\zeta} \geq \lambda\frac{[K-1]^\zeta-1}{\zeta} = \lambda\ln_\zeta(K-1).$$

Consequently, the lower bound is simply obtained by the general lower bound. The corollary is proved. □

Proof of Corollary 8. The CDF of a Gamma distribution is given as

$$F(x)=\frac{\gamma(x;\alpha,\lambda)}{\Gamma(\alpha)},$$

where $\Gamma(\alpha)$ is a (complete) Gamma function and $\gamma(x;\alpha,\lambda)$ is an incomplete Gamma function defined as

$$\gamma(x;\alpha,\lambda) := \int_0^x \frac{z^{\alpha-1}\exp\left(-\frac{z}{\lambda}\right)}{\lambda^\alpha} dz.$$

Before finding a lower and upper bound of F^{-1}, we introduce a lower and upper bound of a Gamma distribution. In [8], the bounds of $F(x)$ is provided as follows, for $\alpha > 1$

$$\left(1-\exp\left(-\frac{x}{\lambda\Gamma(1+\alpha)^{\frac{1}{\alpha}}}\right)\right)^\alpha \le F(x) \le \left(1-\exp\left(-\frac{x}{\lambda}\right)\right)^\alpha.$$

From these bounds, we have,

$$\lambda \ln\left(\frac{1}{1-y^{\frac{1}{\alpha}}}\right) \le F^{-1}(y) \le \lambda\Gamma(1+\alpha)^{\frac{1}{\alpha}} \ln\left(\frac{1}{1-y^{\frac{1}{\alpha}}}\right).$$

Note that the following inequality holds: for $\alpha > 1$,

$$\Gamma(\alpha+1) = \alpha(\alpha-1)\cdots(\alpha-\lfloor\alpha\rfloor+1)\Gamma\left(\alpha-\lfloor\alpha\rfloor+1\right) \le \alpha^{\lfloor\alpha\rfloor}\Gamma\left(1\right) \le \alpha^\alpha.$$

We have a simpler upper bound as

$$F^{-1}(y) \le \lambda\Gamma(1+\alpha)^{\frac{1}{\alpha}} \ln\left(\frac{1}{1-y^{\frac{1}{\alpha}}}\right) \le \lambda\alpha\ln\left(\frac{\alpha}{1-y}\right).$$

Then,

$$\left[F^{-1}\left(1-\frac{1}{T}\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}}\right)\right]^{\frac{p}{p-1}} \le \lambda^{\frac{p}{p-1}}\alpha^{\frac{p}{p-1}}\ln\left(\frac{\alpha T\Delta_a^{\frac{p}{p-1}}}{c^{\frac{p}{p-1}}}\right)^{\frac{p}{p-1}}.$$

C can be obtained as,

$$\begin{aligned}
\int_0^\infty \frac{h(z)\exp\left(-z\right)}{1-F\left(z\right)}dz &= \int_0^\infty \frac{z^{\alpha-1}\exp\left(-\frac{z}{\lambda}-z\right)}{\lambda^\alpha\Gamma(\alpha)\left(1-\left(1-\exp\left(-\frac{z}{\lambda}\right)\right)^\alpha\right)^2}dz \\
&\le \int_0^\infty \frac{z^{\alpha-1}\exp\left(-\frac{z}{\lambda}-z\right)}{\lambda^\alpha\Gamma(\alpha)\exp\left(-2\frac{z}{\lambda}\right)}dz \\
&= \int_0^\infty \frac{z^{\alpha-1}\exp\left(-z+\frac{z}{\lambda}\right)}{\lambda^\alpha\Gamma(\alpha)}dz = \int_0^\infty \frac{t^{\alpha-1}\exp\left(-t\right)}{(\lambda-1)^\alpha\Gamma(\alpha)}dt \\
&= \frac{1}{(\lambda-1)^\alpha}.
\end{aligned}$$

For $\frac{(6c)^{\frac{p}{p-1}}}{\Delta_a^{\frac{1}{p-1}}}\left[-F^{-1}\left(\frac{c^{\frac{p}{p-1}}}{T\Delta_a^{\frac{p}{p-1}}}\right)\right]_+^{\frac{p}{p-1}}$, we have,

$$\frac{(6c)^{\frac{p}{p-1}}}{\Delta_a^{\frac{1}{p-1}}}\left[-F^{-1}\left(\frac{c^{\frac{p}{p-1}}}{T\Delta_a^{\frac{p}{p-1}}}\right)\right]_+^{\frac{p}{p-1}} = 0.$$

since $x \in (0, \infty)$. Then, the problem dependent regret bound becomes,

$$\begin{aligned}
\mathbb{E}\left[\mathcal{R}_T\right] \leq & \sum_{a \neq a^\star}\left[\exp\left(\frac{b_p \nu_p}{2c^p}\right)\left\{C + \frac{F(0)}{1-F(0)} + 2^{\frac{2p-1}{p-1}} + 1\right\}\right. && \text{(F.102)}\\
& \times \Gamma\left(\frac{2p-1}{p-1}\right)\left(\frac{(3c)^p}{\Delta_a}\right)^{\frac{1}{p-1}} && \text{(F.103)}\\
& + \frac{(6c)^{\frac{p}{p-1}}}{\Delta_a^{\frac{1}{p-1}}}\left[-F^{-1}\left(\frac{c^{\frac{p}{p-1}}}{T\Delta_a^{\frac{p}{p-1}}}\right)\right]_+^{\frac{p}{p-1}} && \text{(F.104)}\\
& + \left(\frac{(3c)^p}{\Delta_a}\right)^{\frac{1}{p-1}}\left[F^{-1}\left(1 - \frac{1}{T}\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}}\right)\right]_+^{\frac{p}{p-1}} && \text{(F.105)}\\
& \left. + \left(\frac{c^p}{\Delta_a}\right)^{\frac{1}{p-1}} + \Delta_a\right] && \text{(F.106)}\\
\leq & \sum_{a \neq a^\star}\left[\exp\left(\frac{b_p \nu_p}{2c^p}\right)\left[(\lambda-1)^{-\alpha} + 2^{\frac{2p-1}{p-1}} + 1\right]\Gamma\left(\frac{2p-1}{p-1}\right)\left(\frac{(3c)^p}{\Delta_a}\right)^{\frac{1}{p-1}}\right. && \text{(F.107)}\\
& \left. + \left(\frac{(3\lambda\alpha c)^p}{\Delta_a}\right)^{\frac{1}{p-1}} \ln\left(\frac{\alpha T \Delta_a^{\frac{p}{p-1}}}{c^{\frac{p}{p-1}}}\right)^{\frac{p}{p-1}} + \left(\frac{c^p}{\Delta_a}\right)^{\frac{1}{p-1}} + \Delta_a\right] && \text{(F.108)}\\
\leq & O\left(\sum_{a \neq a^\star}\frac{C_{c,p,\nu_p,F}}{\Delta_a^{\frac{1}{p-1}}} + \left(\frac{(3\lambda\alpha c)^p}{\Delta_a}\right)^{\frac{1}{p-1}} \ln\left(\frac{\alpha T \Delta_a^{\frac{p}{p-1}}}{c^{\frac{p}{p-1}}}\right)^{\frac{p}{p-1}} + \Delta_a\right). && \text{(F.109)}
\end{aligned}$$

The problem independent regret bound can be obtained by choosing the threshold of the minimum gap as $\Delta = c\,(K/T)^{1-\frac{1}{p}}\ln(K)$.

$$\mathbb{E}\left[\mathcal{R}_T\right] \leq \sum_{\Delta_a>\Delta}\Bigg[\exp\left(\frac{b_p\nu_p}{2c^p}\right)\left[(\lambda-1)^{-\alpha}+2^{\frac{2p-1}{p-1}}+1\right]\Gamma\left(\frac{2p-1}{p-1}\right)\left(\frac{(3c)^p}{\Delta_a}\right)^{\frac{1}{p-1}} \tag{F.110}$$

$$+\left(\frac{(3\lambda\alpha c)^p}{\Delta_a}\right)^{\frac{1}{p-1}}\ln\left(\frac{\alpha T\Delta_a^{\frac{p}{p-1}}}{c^{\frac{p}{p-1}}}\right)^{\frac{p}{p-1}}+\left(\frac{c^p}{\Delta_a}\right)^{\frac{1}{p-1}}\Bigg]+\Delta T \tag{F.111}$$

$$\leq K\Bigg[\exp\left(\frac{b_p\nu_p}{2c^p}\right)\left[(\lambda-1)^{-\alpha}+2^{\frac{2p-1}{p-1}}+1\right]\Gamma\left(\frac{2p-1}{p-1}\right)\left(\frac{(3c)^p}{\Delta}\right)^{\frac{1}{p-1}} \tag{F.112}$$

$$+\left(\frac{(3\lambda\alpha c)^p}{\Delta}\right)^{\frac{1}{p-1}}\ln\left(\frac{\alpha T\Delta^{\frac{p}{p-1}}}{c^{\frac{p}{p-1}}}\right)^{\frac{p}{p-1}}+\left(\frac{c^p}{\Delta}\right)^{\frac{1}{p-1}}\Bigg]+\Delta T \tag{F.113}$$

$$\leq \exp\left(\frac{b_p\nu_p}{2c^p}\right)\left[(\lambda-1)^{-\alpha}+2^{\frac{2p-1}{p-1}}+1\right] \tag{F.114}$$

$$\times\Gamma\left(\frac{2p-1}{p-1}\right)3^{\frac{p}{p-1}}cK^{1-\frac{1}{p}}T^{\frac{1}{p}}\ln(K)^{-\frac{1}{p-1}} \tag{F.115}$$

$$+(3\lambda\alpha)^{\frac{1}{p-1}}cK^{1-\frac{1}{p}}T^{\frac{1}{p}}\frac{\ln\left(\alpha K\ln(K)^{\frac{p}{p-1}}\right)^{\frac{p}{p-1}}}{\ln(K)^{\frac{1}{p-1}}} \tag{F.116}$$

$$+cK^{1-\frac{1}{p}}T^{\frac{1}{p}}\ln(K)^{-\frac{1}{p-1}}+cK^{1-\frac{1}{p}}T^{\frac{1}{p}}\ln(K) \tag{F.117}$$

$$\leq \exp\left(\frac{b_p\nu_p}{2c^p}\right)\left[(\lambda-1)^{-\alpha}+2^{\frac{2p-1}{p-1}}+1\right] \tag{F.118}$$

$$\times\Gamma\left(\frac{2p-1}{p-1}\right)3^{\frac{p}{p-1}}cK^{1-\frac{1}{p}}T^{\frac{1}{p}}\ln(K)^{-\frac{1}{p-1}} \tag{F.119}$$

$$+(3\lambda\alpha)^{\frac{1}{p-1}}cK^{1-\frac{1}{p}}T^{\frac{1}{p}}\frac{\ln\left(\alpha K^{1+\frac{p}{p-1}}\right)^{\frac{p}{p-1}}}{\ln(K)^{\frac{1}{p-1}}} \tag{F.120}$$

$$+cK^{1-\frac{1}{p}}T^{\frac{1}{p}}\ln(K)^{-\frac{1}{p-1}}+cK^{1-\frac{1}{p}}T^{\frac{1}{p}}\ln(K) \tag{F.121}$$

$$\leq O\left((\lambda\alpha)^{\frac{1}{p-1}}cK^{1-\frac{1}{p}}T^{\frac{1}{p}}\frac{\ln\left(\alpha K^{1+\frac{p}{p-1}}\right)^{\frac{p}{p-1}}}{\ln(K)^{\frac{1}{p-1}}}\right) \tag{F.122}$$

For the lower bound, we use,

$$F^{-1}(y) \geq \lambda \ln \left(\frac{1}{1 - y^{\frac{1}{\alpha}}} \right) \geq \lambda \ln \left(\frac{y}{1-y} \right).$$

Thus, the lower bound becomes

$$\Omega \left(\lambda K^{1-\frac{1}{p}} T^{\frac{1}{p}} \ln(K) \right).$$

□

Proof of Corollary 9. The CDF of a Pareto distribution is given as

$$F(x) = 1 - \frac{1}{(x/\lambda)^{\alpha}}$$

Then, its inverse is

$$F^{-1}(y) = \lambda \left(1 - y\right)^{-\frac{1}{\alpha}},$$

Then,

$$\left[F^{-1} \left(1 - \frac{1}{T} \left(\frac{c}{\Delta_a} \right)^{\frac{p}{p-1}} \right) \right]^{\frac{p}{p-1}} = \lambda^{\frac{p}{p-1}} \left[\frac{T \Delta_a^{\frac{p}{p-1}}}{c^{\frac{p}{p-1}}} \right]^{\frac{p}{\alpha(p-1)}}.$$

C can be obtained as,

$$\begin{aligned} \int_0^{\infty} \frac{h(z) \exp\left(-z\right)}{1 - F\left(z\right)} dz &= \int_0^{\infty} \frac{\alpha \lambda^{\alpha} z^{-\alpha-1} \exp\left(-z\right)}{(z/\lambda)^{-2\alpha}} dz \\ &= \int_0^{\infty} \frac{\alpha z^{\alpha-1} \exp\left(-z\right)}{\lambda^{\alpha}} dz \\ &= \frac{\alpha \Gamma(\alpha)}{\lambda^{\alpha}} = \frac{\Gamma(\alpha+1)}{\lambda^{\alpha}} \\ &\leq 1 \quad \because \ \lambda \geq \alpha. \end{aligned}$$

For $\frac{(6c)^{\frac{p}{p-1}}}{\Delta_a^{\frac{1}{p-1}}} \left[-F^{-1} \left(\frac{c^{\frac{p}{p-1}}}{T \Delta_a^{\frac{p}{p-1}}} \right) \right]_+^{\frac{p}{p-1}}$, we have,

$$\frac{(6c)^{\frac{p}{p-1}}}{\Delta_a^{\frac{1}{p-1}}} \left[-F^{-1} \left(\frac{c^{\frac{p}{p-1}}}{T \Delta_a^{\frac{p}{p-1}}} \right) \right]_+^{\frac{p}{p-1}} = 0$$

where $-F^{-1}(y)$ is always negative since the support of x is (λ, ∞). Then, the problem dependent regret bound becomes,

$$\mathbb{E}\left[\mathcal{R}_T\right] \leq \sum_{a \neq a^\star} \Bigg[\exp\left(\frac{b_p \nu_p}{2c^p}\right) \left\{ C + \frac{F(0)}{1-F(0)} + 2^{\frac{2p-1}{p-1}} + 1 \right\} \tag{F.123}$$

$$\times \Gamma\left(\frac{2p-1}{p-1}\right) \left(\frac{(3c)^p}{\Delta_a}\right)^{\frac{1}{p-1}} + \frac{(6c)^{\frac{p}{p-1}}}{\Delta_a^{\frac{1}{p-1}}} \left[-F^{-1}\left(\frac{c^{\frac{p}{p-1}}}{T\Delta_a^{\frac{p}{p-1}}}\right)\right]_+^{\frac{p}{p-1}} \tag{F.124}$$

$$+ \left(\frac{(3c)^p}{\Delta_a}\right)^{\frac{1}{p-1}} \left[F^{-1}\left(1 - \frac{1}{T}\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}}\right)\right]_+^{\frac{p}{p-1}} \tag{F.125}$$

$$+ \left(\frac{c^p}{\Delta_a}\right)^{\frac{1}{p-1}} + \Delta_a \Bigg] \tag{F.126}$$

$$\leq \sum_{a \neq a^\star} \Bigg[\exp\left(\frac{b_p \nu_p}{2c^p}\right) \left[2^{\frac{2p-1}{p-1}} + 2\right] \Gamma\left(\frac{2p-1}{p-1}\right) \left(\frac{(3c)^p}{\Delta_a}\right)^{\frac{1}{p-1}} \tag{F.127}$$

$$+ \left(\frac{(3\lambda c)^p}{\Delta_a}\right)^{\frac{1}{p-1}} \left[\frac{T\Delta_a^{\frac{p}{p-1}}}{c^{\frac{p}{p-1}}}\right]^{\frac{p}{\alpha(p-1)}} + \left(\frac{c^p}{\Delta_a}\right)^{\frac{1}{p-1}} + \Delta_a \Bigg] \tag{F.128}$$

$$\leq O\left(\sum_{a \neq a^\star} \frac{C_{c,p,\nu_p,F}}{\Delta_a^{\frac{1}{p-1}}} + \left(\frac{(3\lambda c)^p}{\Delta_a}\right)^{\frac{1}{p-1}} \left[\frac{T\Delta_a^{\frac{p}{p-1}}}{c^{\frac{p}{p-1}}}\right]^{\frac{p}{\alpha(p-1)}} + \Delta_a \right). \tag{F.129}$$

The problem independent regret bound can be obtained by choosing the threshold of the minimum gap as $\Delta = c\left(K/T\right)^{1-\frac{1}{p}}\alpha$.

$$\mathbb{E}\left[\mathcal{R}_T\right] \leq \sum_{\Delta_a > \Delta} \Bigg[\exp\left(\frac{b_p \nu_p}{2c^p}\right) \left[2^{\frac{2p-1}{p-1}} + 2\right] \Gamma\left(\frac{2p-1}{p-1}\right) \left(\frac{(3c)^p}{\Delta_a}\right)^{\frac{1}{p-1}} \tag{F.130}$$

$$+ \left(\frac{(3\lambda c)^p}{\Delta_a}\right)^{\frac{1}{p-1}} \left[\frac{T\Delta_a^{\frac{p}{p-1}}}{c^{\frac{p}{p-1}}}\right]^{\frac{p}{\alpha(p-1)}} + \left(\frac{c^p}{\Delta_a}\right)^{\frac{1}{p-1}} \Bigg] + \Delta T \tag{F.131}$$

$$\leq K \Bigg[\exp\left(\frac{b_p \nu_p}{2c^p}\right) \left[2^{\frac{2p-1}{p-1}} + 2\right] \Gamma\left(\frac{2p-1}{p-1}\right) \left(\frac{(3c)^p}{\Delta}\right)^{\frac{1}{p-1}} \tag{F.132}$$

$$+ \left(\frac{(3\lambda c)^p}{\Delta}\right)^{\frac{1}{p-1}} \left[\frac{T\Delta^{\frac{p}{p-1}}}{c^{\frac{p}{p-1}}}\right]^{\frac{p}{\alpha(p-1)}} + \left(\frac{c^p}{\Delta}\right)^{\frac{1}{p-1}} \Bigg] + \Delta T \tag{F.133}$$

$$\because\ x^{\frac{p^2}{\alpha(p-1)^2} - \frac{1}{p-1}} \text{ is decreasing for } \alpha > \frac{p^2}{p-1} \tag{F.134}$$

$$\leq \exp\left(\frac{b_p \nu_p}{2c^p}\right)\left[2^{\frac{2p-1}{p-1}}+2\right]\Gamma\left(\frac{2p-1}{p-1}\right)3^{\frac{p}{p-1}}cK^{1-\frac{1}{p}}T^{\frac{1}{p}}\alpha^{-\frac{1}{p-1}} \tag{F.135}$$

$$+(3\lambda)^{\frac{p}{p-1}}cK^{1-\frac{1}{p}+\frac{p^2}{\alpha(p-1)^2}}T^{\frac{1}{p}}\alpha^{\frac{p^2}{\alpha(p-1)^2}-\frac{1}{p-1}} \tag{F.136}$$

$$+c\alpha^{\frac{1}{p-1}}K^{1-\frac{1}{p}}T^{\frac{1}{p}}+c\alpha K^{1-\frac{1}{p}}T^{\frac{1}{p}} \tag{F.137}$$

$$\leq \exp\left(\frac{b_p \nu_p}{2c^p}\right)\left[2^{\frac{2p-1}{p-1}}+2\right]\Gamma\left(\frac{2p-1}{p-1}\right)3^{\frac{p}{p-1}}cK^{1-\frac{1}{p}}T^{\frac{1}{p}}\alpha^{-\frac{1}{p-1}} \tag{F.138}$$

$$+3^{\frac{p}{p-1}}cK^{1-\frac{1}{p}+\frac{p^2}{\alpha(p-1)^2}}T^{\frac{1}{p}}\alpha^{1+\frac{p^2}{\alpha(p-1)^2}}+c\alpha^{\frac{1}{p-1}}K^{1-\frac{1}{p}}T^{\frac{1}{p}}+c\alpha K^{1-\frac{1}{p}}T^{\frac{1}{p}} \tag{F.139}$$

$$\because\ \lambda=\alpha \tag{F.140}$$

$$\leq O\left(c\alpha^{1+\frac{p^2}{\alpha(p-1)^2}}K^{1-\frac{1}{p}+\frac{2p}{\alpha(p-1)}}T^{\frac{1}{p}}\right). \tag{F.141}$$

For the minimum rate, we set $\alpha = \ln(K)$, then,

$$O\left(\ln(K)^{1+\frac{p^2}{\ln(K)(p-1)^2}}K^{1-\frac{1}{p}+\frac{2p}{\ln(K)(p-1)}}T^{\frac{1}{p}}\right)\leq O\left(K^{1-\frac{1}{p}}T^{\frac{1}{p}}\ln(K)\right)$$

where $\ln(K)^{1+\frac{p^2}{\ln(K)(p-1)^2}} \leq e^{\frac{p^2}{e(p-1)^2}}\ln(K)$. For the lower bound,

$$\Omega\left(K^{1-\frac{1}{p}}T^{\frac{1}{p}}F^{-1}\left(1-\frac{1}{K}\right)\right)=\Omega\left(\lambda K^{1-\frac{1}{p}+\frac{1}{\alpha}}T^{\frac{1}{p}}\right)\geq\Omega\left(\alpha K^{1-\frac{1}{p}+\frac{1}{\alpha}}T^{\frac{1}{p}}\right)$$

The corollary is proved. □

Proof of Corollary 10. The CDF of a Fréchet distribution is given as

$$F(x)=\exp\left(-\left(\frac{x}{\lambda}\right)^{-\alpha}\right)$$

Then, its inverse is

$$F^{-1}(y)=\lambda\ln(1/y)^{-1/\alpha}\leq(1-y)^{-1/\alpha}$$

and

$$\lambda\ln(1/y)^{-1/\alpha}\geq\lambda\left(\frac{y}{1-y}\right)^{\frac{1}{\alpha}}$$

where $\ln(x) \leq x-1$ is used. Then,

$$\left[F^{-1}\left(1-\frac{c^2}{T\Delta_a^2}\right)\right]^2 \leq \lambda^2\left[\frac{T\Delta_a^2}{c^2}\right]^{2/\alpha}.$$

In [5], we have $\sup h \leq 2\frac{\alpha}{\lambda} \leq 2$ due to $\lambda \geq \alpha$, and M can be obtained,

$$\begin{aligned}
\int_0^\infty \frac{\exp(-z)}{\left(1-\exp\left(-\left(\frac{z}{\lambda}\right)^{-\alpha}\right)\right)}dz \leq & \int_0^\infty \left(1+\left(\frac{z}{\lambda}\right)^\alpha\right)\exp(-z)\,dz \\
& \because\ 1/(1-\exp(-x^{-1})) \leq 1+x \\
=& 1+\int_0^\infty \left(\frac{z}{\lambda}\right)^\alpha \exp(-z)\,dz \\
=& 1+\frac{\Gamma(\alpha+1)}{\lambda^\alpha} \\
\leq& 1+\frac{\Gamma(\alpha+1)}{\lambda^\alpha} \leq 2.
\end{aligned}$$

Thus,

$$(\sup h)M \leq 4.$$

For $\frac{(6c)^{\frac{p}{p-1}}}{\Delta_a^{\frac{1}{p-1}}}\left[-F^{-1}\left(\frac{c^{\frac{p}{p-1}}}{T\Delta_a^{\frac{p}{p-1}}}\right)\right]_+^{\frac{p}{p-1}}$, the summation is zero,

$$\frac{(6c)^{\frac{p}{p-1}}}{\Delta_a^{\frac{1}{p-1}}}\left[-F^{-1}\left(\frac{c^{\frac{p}{p-1}}}{T\Delta_a^{\frac{p}{p-1}}}\right)\right]_+^{\frac{p}{p-1}} = 0,$$

since its support is $(0,\infty)$. Then, the problem dependent regret bound becomes,

$$\begin{aligned}
\mathbb{E}\left[\mathcal{R}_T\right] \leq \sum_{a\neq a^\star} &\left[\exp\left(\frac{b_p\nu_p}{2c^p}\right)\left\{\|h\|_\infty M_1 + \frac{F(0)}{1-F(0)} + 2^{\frac{2p-1}{p-1}} + 1\right\}\right. \qquad \text{(F.142)}\\
&\times\Gamma\left(\frac{2p-1}{p-1}\right)\left(\frac{(3c)^p}{\Delta_a}\right)^{\frac{1}{p-1}} + \frac{(6c)^{\frac{p}{p-1}}}{\Delta_a^{\frac{1}{p-1}}}\left[-F^{-1}\left(\frac{c^{\frac{p}{p-1}}}{T\Delta_a^{\frac{p}{p-1}}}\right)\right]_+^{\frac{p}{p-1}} \qquad \text{(F.143)}\\
&+\left(\frac{(3c)^p}{\Delta_a}\right)^{\frac{1}{p-1}}\left\{F^{-1}\left(1-\frac{1}{T}\left(\frac{c}{\Delta_a}\right)^{\frac{p}{p-1}}\right)\right\}^{\frac{p}{p-1}} \qquad \text{(F.144)}\\
&\left.+\left(\frac{c^p}{\Delta_a}\right)^{\frac{1}{p-1}} + \Delta_a\right] \qquad \text{(F.145)}\\
\leq \sum_{a\neq a^\star} &\left[\exp\left(\frac{b_p\nu_p}{2c^p}\right)\left\{4 + 2^{\frac{2p-1}{p-1}} + 1\right\}\Gamma\left(\frac{2p-1}{p-1}\right)\left(\frac{(3c)^p}{\Delta_a}\right)^{\frac{1}{p-1}}\right. \qquad \text{(F.146)}\\
&\left.+\left(\frac{(3c\lambda)^p}{\Delta_a}\right)^{\frac{1}{p-1}}\left[\frac{T\Delta_a^{\frac{p}{p-1}}}{c^{\frac{p}{p-1}}}\right]^{\frac{p}{\alpha(p-1)}} + \left(\frac{c^p}{\Delta_a}\right)^{\frac{1}{p-1}} + \Delta_a\right] \qquad \text{(F.147)}\\
\leq O&\left(\sum_{a\neq a^\star}\frac{C_{c,p,\nu_p,F}}{\Delta_a^{\frac{1}{p-1}}} + \left(\frac{(3c\lambda)^p}{\Delta_a}\right)^{\frac{1}{p-1}}\left[\frac{T\Delta_a^{\frac{p}{p-1}}}{c^{\frac{p}{p-1}}}\right]^{\frac{p}{\alpha(p-1)}} + \Delta_a\right). \qquad \text{(F.148)}
\end{aligned}$$

The problem independent regret bound can be obtained by choosing the threshold of the minimum gap as $\Delta = c\,(K/T)^{1-\frac{1}{p}}\,\alpha$.

$$\begin{aligned}
\mathbb{E}\left[\mathcal{R}_T\right] \leq \sum_{\Delta_a>\Delta} &\left[\exp\left(\frac{b_p\nu_p}{2c^p}\right)\left[5 + 2^{\frac{2p-1}{p-1}}\right]\Gamma\left(\frac{2p-1}{p-1}\right)\left(\frac{(3c)^p}{\Delta_a}\right)^{\frac{1}{p-1}}\right. \qquad \text{(F.149)}\\
&\left.+\left(\frac{(3c\lambda)^p}{\Delta_a}\right)^{\frac{1}{p-1}}\left[\frac{T\Delta_a^{\frac{p}{p-1}}}{c^{\frac{p}{p-1}}}\right]^{\frac{p}{\alpha(p-1)}} + \left(\frac{c^p}{\Delta_a}\right)^{\frac{1}{p-1}}\right] + \Delta T \qquad \text{(F.150)}\\
\leq K&\left[\exp\left(\frac{b_p\nu_p}{2c^p}\right)\left[5 + 2^{\frac{2p-1}{p-1}}\right]\Gamma\left(\frac{2p-1}{p-1}\right)\left(\frac{(3c)^p}{\Delta}\right)^{\frac{1}{p-1}}\right. \qquad \text{(F.151)}\\
&\left.+\left(\frac{(3c\lambda)^p}{\Delta}\right)^{\frac{1}{p-1}}\left[\frac{T\Delta^{\frac{p}{p-1}}}{c^{\frac{p}{p-1}}}\right]^{\frac{p}{\alpha(p-1)}} + \left(\frac{c^p}{\Delta}\right)^{\frac{1}{p-1}}\right] + \Delta T \qquad \text{(F.152)}\\
\leq \exp&\left(\frac{b_p\nu_p}{2c^p}\right)\left[5 + 2^{\frac{2p-1}{p-1}}\right]\Gamma\left(\frac{2p-1}{p-1}\right)3^{\frac{p}{p-1}}cK^{1-\frac{1}{p}}T^{\frac{1}{p}}\alpha^{-\frac{1}{p-1}} \qquad \text{(F.153)}\\
&+3^{\frac{p}{p-1}}c\lambda^{\frac{p}{p-1}}K^{1-\frac{1}{p}+\frac{p^2}{\alpha(p-1)^2}}T^{\frac{1}{p}}\alpha^{\frac{p^2}{\alpha(p-1)^2}-\frac{1}{p-1}} \qquad \text{(F.154)}
\end{aligned}$$

$$
\begin{aligned}
&+ c\alpha^{\frac{1}{p-1}} K^{1-\frac{1}{p}} T^{\frac{1}{p}} + c\alpha K^{1-\frac{1}{p}} T^{\frac{1}{p}} && \text{(F.155)}\\
\leq &\exp\left(\frac{b_p \nu_p}{2c^p}\right)\left[5 + 2^{\frac{2p-1}{p-1}}\right]\Gamma\left(\frac{2p-1}{p-1}\right) 3^{\frac{p}{p-1}} c K^{1-\frac{1}{p}} T^{\frac{1}{p}} \alpha^{-\frac{1}{p-1}} && \text{(F.156)}\\
&+ 3^{\frac{p}{p-1}} c K^{1-\frac{1}{p}+\frac{p^2}{\alpha(p-1)^2}} T^{\frac{1}{p}} \alpha^{1+\frac{p^2}{\alpha(p-1)^2}-\frac{1}{p-1}} && \text{(F.157)}\\
&+ c\alpha^{\frac{1}{p-1}} K^{1-\frac{1}{p}} T^{\frac{1}{p}} + c\alpha K^{1-\frac{1}{p}} T^{\frac{1}{p}} && \text{(F.158)}\\
\leq &O\left(\alpha^{1+\frac{p^2}{\alpha(p-1)^2}} K^{1-\frac{1}{p}+\frac{p^2}{\alpha(p-1)^2}} T^{\frac{1}{p}}\right). && \text{(F.159)}
\end{aligned}
$$

The optimal rate is obtained by setting $\alpha = \ln(K)$,

$$
O\left(\ln(K)^{1+\frac{p^2}{\ln(K)(p-1)^2}} K^{1-\frac{1}{p}+\frac{2p}{\ln(K)(p-1)}} T^{\frac{1}{p}}\right) \leq O\left(K^{1-\frac{1}{p}} T^{\frac{1}{p}} \ln(K)\right),
$$

where $\ln(K)^{\frac{p^2}{\ln(K)(p-1)^2}} \leq e^{\frac{p^2}{e(p-1)^2}}$. Before proving the lower bound, note that

$$
F^{-1}\left(1-\frac{1}{K}\right) = \lambda \ln\left(\frac{1}{1-\frac{1}{K}}\right)^{-1/\alpha} \geq \alpha\left(K-1\right)^{1/\alpha}
$$

Consequently, the lower bound is simply obtained by the general lower bound. The corollary is proved. □

Bibliography

[1] Pieter Abbeel and Andrew Y. Ng. Apprenticeship learning via inverse reinforcement learning. In *Proceedings of the 21st International Conference of Machine Learning*, July 2004.

[2] Pieter Abbeel, Adam Coates, and Andrew Y. Ng. Autonomous helicopter aerobatics through apprenticeship learning. *International Journal of Robotics Research*, 29(13):1608–1639, 2010.

[3] Jacob D. Abernethy, Elad Hazan, and Alexander Rakhlin. Competing in the dark: An efficient algorithm for bandit linear optimization. In *21st Annual Conference on Learning Theory*, pages 263–274, July 2008.

[4] Jacob D. Abernethy, Chansoo Lee, Abhinav Sinha, and Ambuj Tewari. Online linear optimization via smoothing. In *Proceedings of The 27th Conference on Learning Theory*, pages 807–823, June 2014.

[5] Jacob D. Abernethy, Chansoo Lee, and Ambuj Tewari. Fighting bandits with a new kind of smoothness. In *Advances in Neural Information Processing Systems (NeurIPS)*, December 2015.

[6] Navid Aghasadeghi and Timothy Bretl. Maximum entropy inverse rein-

forcement learning in continuous state spaces with path integrals. In *International Conference on Intelligent Robots and Systems*, September 2011.

[7] Shipra Agrawal and Navin Goyal. Further optimal regret bounds for thompson sampling. In *Proc. of the 16th International Conference on Artificial Intelligence and Statistics (AISTATS)*, April 2013.

[8] Horst Alzer. On some inequalities for the incomplete gamma function. *Math. Comput.*, 66(218):771–778, 1997.

[9] Shunichi Amari and Atsumi Ohara. Geometry of q-exponential family of probability distributions. *Entropy*, 13(6):1170–1185, 2011.

[10] Brenna Argall, Sonia Chernova, Manuela M. Veloso, and Brett Browning. A survey of robot learning from demonstration. *Robotics and Autonomous Systems*, 57(5):469–483, 2009.

[11] Julien Audiffren, Michal Valko, Alessandro Lazaric, and Mohammad Ghavamzadeh. Maximum entropy semi-supervised inverse reinforcement learning. In *Proc. of the 24th International Joint Conference on Artificial Intelligence*. AAAI Press, July 2015.

[12] Peter Auer. Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research*, 3:397–422, 2002.

[13] Peter Auer, Nicolò Cesa-Bianchi, Yoav Freund, and Robert E. Schapire. The nonstochastic multiarmed bandit problem. *SIAM J. Comput.*, 32(1): 48–77, 2002.

[14] Mohammad Gheshlaghi Azar, Vicenç Gómez, and Hilbert J. Kappen. Dynamic policy programming. *Journal of Machine Learning Research*, 13: 3207–3245, 2012.

[15] Andrew R. Barron. Approximation and estimation bounds for artificial neural networks. In *Proceedings of the Fourth Annual Workshop on Computational Learning Theory, COLT 1991, Santa Cruz, California, USA, August 5-7, 1991*, pages 243–249, 1991.

[16] Boris Belousov and Jan Peters. Entropic regularization of markov decision processes. *Entropy*, 21(7):674, 2019.

[17] Felix Berkenkamp, Angela P. Schoellig, and Andreas Krause. Safe controller optimization for quadrotors with gaussian processes. In *2016 IEEE International Conference on Robotics and Automation, ICRA 2016, Stockholm, Sweden, May 16-21, 2016*, pages 491–496, 2016.

[18] Lars Berscheid, Thomas Rühr, and Torsten Kröger. Improving data efficiency of self-supervised learning for robotic grasping. In *International Conference on Robotics and Automation, ICRA 2019, Montreal, QC, Canada, May 20-24, 2019*, pages 2125–2131, 2019.

[19] Michael Bloem and Nicholas Bambos. Infinite time horizon maximum causal entropy inverse reinforcement learning. In *Proc. of the IEEE Conference on Decision and Control*, Dec 2014.

[20] Salomon Bochner. *Harmonic analysis and the theory of probability.* Courier Dover Publications, 2012.

[21] Abdeslam Boularias, Jens Kober, and Jan Peters. Relative entropy inverse reinforcement learning. In *Proc. of the 14th International Conference on Artificial Intelligence and Statistics.* JMLR.org, April 2011.

[22] Konstantinos Bousmalis, Alex Irpan, Paul Wohlhart, Yunfei Bai, Matthew Kelcey, Mrinal Kalakrishnan, Laura Downs, Julian Ibarz, Peter Pastor,

Kurt Konolige, Sergey Levine, and Vincent Vanhoucke. Using simulation and domain adaptation to improve efficiency of deep robotic grasping. In *2018 IEEE International Conference on Robotics and Automation, ICRA 2018, Brisbane, Australia, May 21-25, 2018*, pages 4243–4250, 2018.

[23] Sebastian Brechtel, Tobias Gindele, and Rüdiger Dillmann. Probabilistic decision-making under uncertainty for autonomous driving using continuous pomdps. In *Proc. of the International Conference on Intelligent Transportation Systems*, Oct 2014.

[24] Glenn W Brier. Verification of forecasts expressed in terms of probability. *Monthey Weather Review*, 78(1):1–3, 1950.

[25] Sébastien Bubeck, Rémi Munos, and Gilles Stoltz. Pure exploration in multi-armed bandits problems. In Ricard Gavaldà, Gábor Lugosi, Thomas Zeugmann, and Sandra Zilles, editors, *Proc. of the 20th International Conference on Algorithmic Learning Theory (ALT)*, October 2009.

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

[27] Junhao Cai, Hui Cheng, Zhanpeng Zhang, and Jingcheng Su. Metagrasp: Data efficient grasping by affordance interpreter network. In *International Conference on Robotics and Automation, ICRA 2019, Montreal, QC, Canada, May 20-24, 2019*, pages 4960–4966, 2019.

[28] 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.

[29] Nicolò Cesa-Bianchi, Claudio Gentile, Gergely Neu, and Gábor Lugosi. Boltzmann exploration done right. In *Advances in Neural Information Processing Systems (NeurIPS)*, December 2017.

[30] Jaedeug Choi and Kee-Eung Kim. Bayesian nonparametric feature construction for inverse reinforcement learning. In *Proceedings of the 23rd International Joint Conference on Artificial Intelligence*. IJCAI/AAAI, August 2013.

[31] Jaedeug Choi and Kee-Eung Kim. Hierarchical bayesian inverse reinforcement learning. *Cybernetics, IEEE Transactions on*, 45(4):793–805, 2015.

[32] Sungjoon Choi, Eunwoo Kim, Kyungjae Lee, and Songhwai Oh. Leveraged non-stationary gaussian process regression for autonomous robot navigation. In *Proc. of the IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, May 2015.

[33] Sungjoon Choi, Kyungjae Lee, and Songhwai Oh. Robust learning from demonstration using leveraged gaussian processes and sparse constrained opimization. In *Proc. of the IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, May 2016.

[34] Yinlam Chow, Ofir Nachum, and Mohammad Ghavamzadeh. Path consistency learning in tsallis entropy regularized mdps. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018*, pages 978–987, Stockholmsmässan, Stockholm, Sweden, 2018.

[35] Bo Dai, Albert Shaw, Lihong Li, Lin Xiao, Niao He, Zhen Liu, Jianshu Chen, and Le Song. SBEED: convergent reinforcement learning with nonlinear function approximation. In *Proceedings of the 35th International*

Conference on Machine Learning, (ICML 2018), pages 1133–1142, Stockholmsmässan, Stockholm, Sweden, 2018.

[36] Guy Van den Broeck, Kurt Driessens, and Jan Ramon. Monte-carlo tree search in poker using expected reward distributions. In *Advances in Machine Learning, First Asian Conference on Machine Learning*, pages 367–381, November 2009.

[37] Yan Duan, Xi Chen, Rein Houthooft, John Schulman, and Pieter Abbeel. Benchmarking deep reinforcement learning for continuous control. In *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016*, pages 1329–1338, New York City, NY, USA, 2016. JMLR.org.

[38] Krishnamurthy Dvijotham and Emanuel Todorov. Inverse optimal control with linearly-solvable mdps. In *Proc. of the 27th International Conference on Machine Learning*. Omnipress, June 2010.

[39] Benjamin Eysenbach, Abhishek Gupta, Julian Ibarz, and Sergey Levine. Diversity is all you need: Learning skills without a reward function. In *International Conference on Learning Representations ICLR*, 2019. URL `https://openreview.net/forum?id=SJx63jRqFm`.

[40] Scott Fujimoto, Herke van Hoof, and David Meger. Addressing function approximation error in actor-critic methods. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018*, pages 1582–1591, Stockholmsmässan, Stockholm, Sweden, 2018.

[41] Matthieu Geist, Bruno Scherrer, and Olivier Pietquin. A theory of regularized markov decision processes. *arXiv preprint arXiv:1901.11275*, 2019.

[42] Ghazal Ghazaei, Iro Laina, Christian Rupprecht, Federico Tombari, Nassir Navab, and Kianoush Nazarpour. Dealing with ambiguity in robotic grasping via multiple predictions. In *Computer Vision - ACCV 2018 - 14th Asian Conference on Computer Vision, Perth, Australia, December 2-6, 2018, Revised Selected Papers, Part IV*, pages 38–55, 2018.

[43] John Gittins. Quantitative methods in the planning of pharmaceutical research. *Drug Information Journal*, 30(2):479–487, 1996.

[44] Jordi Grau-Moya, Felix Leibfried, and Peter Vrancx. Soft q-learning with mutual-information regularization. In *International Conference on Learning Representations*, 2019. URL `https://openreview.net/forum?id=HyEtjoCqFX`.

[45] Peter D Grünwald and A Philip Dawid. Game theory, maximum entropy, minimum discrepancy and robust Bayesian decision theory. *Annals of Statistics*, pages 1367–1433, 2004.

[46] Shixiang Gu, Timothy P. Lillicrap, Ilya Sutskever, and Sergey Levine. Continuous deep q-learning with model-based acceleration. In *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016*, pages 2829–2838, New York City, NY, USA, 2016.

[47] Marcus Gualtieri, Andreas ten Pas, Kate Saenko, and Robert Platt Jr. High precision grasp pose detection in dense clutter. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2016, Daejeon, South Korea, October 9-14, 2016*, pages 598–605, 2016.

[48] Abhinav Gupta, Adithyavairavan Murali, Dhiraj Gandhi, and Lerrel Pinto. Robot learning in homes: Improving generalization and reducing dataset

bias. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada.*, pages 9112–9122, 2018.

[49] Tuomas Haarnoja, Aurick Zhou, Sehoon Ha, Jie Tan, George Tucker, and Sergey Levine. Learning to walk via deep reinforcement learning. In *Proceedings of the 15th Robotics: Science and Systems, RSS 2019.*

[50] Tuomas Haarnoja, Haoran Tang, Pieter Abbeel, and Sergey Levine. Reinforcement learning with deep energy-based policies. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017*, pages 1352–1361, Sydney, NSW, Australia, 2017.

[51] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018*, pages 1856–1865, Stockholmsmässan, Stockholm, Sweden, 2018.

[52] Karol Hausman, Yevgen Chebotar, Stefan Schaal, Gaurav S. Sukhatme, and Joseph J. Lim. Multi-modal imitation learning from unstructured demonstrations using generative adversarial nets. In *Advances in Neural Information Processing Systems*, pages 1235–1245, December 2017.

[53] Nicolas Heess, David Silver, and Yee Whye Teh. Actor-critic reinforcement learning with energy-based policies. In *Proc. of the Tenth European Workshop on Reinforcement Learning*, Jun 2012.

[54] Dave Higdon, J Swall, and J Kern. Non-stationary spatial modeling. *Bayesian statistics*, 6(1):761–768, 1999.

[55] Jonathan Ho and Stefano Ermon. Generative adversarial imitation learning. In *Advances in Neural Information Processing Systems*, pages 4565–4573, December 2016.

[56] Tomas Hodan, Pavel Haluza, Stepán Obdrzálek, Jiri Matas, Manolis I. A. Lourakis, and Xenophon Zabulis. T-LESS: an RGB-D dataset for 6d pose estimation of texture-less objects. In *2017 IEEE Winter Conference on Applications of Computer Vision, WACV 2017, Santa Rosa, CA, USA, March 24-31, 2017*, pages 880–888, 2017.

[57] Jemin Hwangbo, Inkyu Sa, Roland Siegwart, and Marco Hutter. Control of a quadrotor with reinforcement learning. *IEEE Robotics and Automation Letters*, 2(4):2096–2103, Jun 2017.

[58] Stephen James, Andrew J. Davison, and Edward Johns. Transferring end-to-end visuomotor control from simulation to real world for a multi-stage task. In *1st Annual Conference on Robot Learning, CoRL 2017, Mountain View, California, USA, November 13-15, 2017, Proceedings*, pages 334–343, 2017.

[59] Stephen James, Paul Wohlhart, Mrinal Kalakrishnan, Dmitry Kalashnikov, Alex Irpan, Julian Ibarz, Sergey Levine, Raia Hadsell, and Konstantinos Bousmalis. Sim-to-real via sim-to-sim: Data-efficient robotic grasping via randomized-to-canonical adaptation networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 12627–12637, 2019.

[60] Anmol Kagrecha, Jayakrishnan Nair, and Krishna P. Jagannathan. Distribution oblivious, risk-aware algorithms for multi-armed bandits with un-

bounded rewards. In *Advances in Neural Information Processing Systems (NeurIPS)*, December 2019.

[61] Adam Tauman Kalai and Santosh S. Vempala. Efficient algorithms for online decision problems. *Journal of Computer and System Sciences*, 71 (3):291–307, 2005.

[62] Dmitry Kalashnikov, Alex Irpan, Peter Pastor, Julian Ibarz, Alexander Herzog, Eric Jang, Deirdre Quillen, Ethan Holly, Mrinal Kalakrishnan, Vincent Vanhoucke, and Sergey Levine. Scalable deep reinforcement learning for vision-based robotic manipulation. In *2nd Annual Conference on Robot Learning, CoRL 2018, Zürich, Switzerland, 29-31 October 2018, Proceedings*, pages 651–673, 2018.

[63] Alexander Kasper, Zhixing Xue, and Rüdiger Dillmann. The kit object models database: An object model database for object recognition, localization and manipulation in service robotics. *The International Journal of Robotics Research*, 31(8):927–934, 2012.

[64] Baekjin Kim and Ambuj Tewari. On the optimality of perturbations in stochastic and adversarial multi-armed bandit problems. In *Advances in Neural Information Processing Systems (NeurIPS)*, December 2019.

[65] Beomjoon Kim and Joelle Pineau. Socially adaptive path planning in human environments using inverse reinforcement learning. *International Journal of Social Robotics*, 8(1):51–66, January 2015.

[66] DongWook Kim, Jae In Kim, and Yong-Lae Park. A simple tripod mobile robot using soft membrane vibration actuators. *IEEE Robotics and Automation Letters*, 4(3):2289–2295, 2019.

[67] Sangbae Kim, Cecilia Laschi, and Barry Trimmer. Soft robotics: a bioinspired evolution in robotics. *Trends in biotechnology*, 31(5):287–294, 2013.

[68] Jens Kober, J. Andrew Bagnell, and Jan Peters. Reinforcement learning in robotics: A survey. *International Journal of Robotics Research*, 32(11): 1238–1274, Aug 2013.

[69] N Koenig and J Hsu. The many faces of simulation: Use cases for a general purpose simulator. In *International Conference on Robotics and Automation, ICRA 2013*, volume 13, pages 10–11, 2013.

[70] Torsten Koller, Felix Berkenkamp, Matteo Turchetta, and Andreas Krause. Learning-based model predictive control for safe exploration. In *57th IEEE Conference on Decision and Control, CDC 2018, Miami, FL, USA, December 17-19, 2018*, pages 6059–6066, 2018.

[71] Jussi Kujala and Tapio Elomaa. On following the perturbed leader in the bandit setting. In *Algorithmic Learning Theory, 16th International Conference, Singapore*, pages 371–385, October 2005.

[72] Sulabh Kumra and Christopher Kanan. Robotic grasp detection using deep convolutional neural networks. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2017, Vancouver, BC, Canada, September 24-28, 2017*, pages 769–776, 2017.

[73] Tobias Lang, Christian Plagemann, and Wolfram Burgard. Adaptive nonstationary kernel regression for terrain modeling. In *Robotics: Science and Systems*, 2007.

[74] Kyungjae Lee, Sungjoon Choi, and Songhwai Oh. Sparse Markov decision

processes with causal sparse Tsallis entropy regularization for reinforcement learning. *IEEE Robotics and Automation Letters*, 3(3):1466–1473, 2018.

[75] Sergey Levine, Zoran Popovic, and Vladlen Koltun. Feature construction for inverse reinforcement learning. In *Advances in Neural Information Processing Systems 23*, December 2010.

[76] Sergey Levine, Zoran Popovic, and Vladlen Koltun. Nonlinear inverse reinforcement learning with gaussian processes. In *Advances in Neural Information Processing Systems*, pages 19–27, 2011.

[77] Yunzhu Li, Jiaming Song, and Stefano Ermon. Infogail: Interpretable imitation learning from visual demonstrations. In *Advances in Neural Information Processing Systems*, pages 3815–3825, December 2017.

[78] Hongzhuo Liang, Xiaojian Ma, Shuang Li, Michael Görner, Song Tang, Bin Fang, Fuchun Sun, and Jianwei Zhang. Pointnetgpd: Detecting grasp configurations from point sets. In *International Conference on Robotics and Automation, ICRA 2019, Montreal, QC, Canada, May 20-24, 2019*, pages 3629–3635, 2019.

[79] Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*, 2015.

[80] Timothy P. Lillicrap, Jonathan J. Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. *CoRR*, abs/1509.02971, 2015. URL `http://arxiv.org/abs/1509.02971`.

[81] Shiyin Lu, Guanghui Wang, Yao Hu, and Lijun Zhang. Optimal algorithms for lipschitz bandits with heavy-tailed rewards. In *Proc. of the 36th International Conference on Machine Learning (ICML)*, July 2019.

[82] Jeffrey Mahler and Ken Goldberg. Learning deep policies for robot bin picking by simulating robust grasping sequences. In Sergey Levine, Vincent Vanhoucke, and Ken Goldberg, editors, *Proceedings of the 1st Annual Conference on Robot Learning*, volume 78 of *Proceedings of Machine Learning Research*, pages 515–524. PMLR, 13–15 Nov 2017.

[83] Jeffrey Mahler, Florian T Pokorny, Brian Hou, Melrose Roderick, Michael Laskey, Mathieu Aubry, Kai Kohlhoff, Torsten Kröger, James Kuffner, and Ken Goldberg. Dex-net 1.0: A cloud-based network of 3d objects for robust grasp planning using a multi-armed bandit model with correlated rewards. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1957–1964. IEEE, 2016.

[84] Jeffrey Mahler, Jacky Liang, Sherdil Niyaz, Michael Laskey, Richard Doan, Xinyu Liu, Juan Aparicio Ojea, and Ken Goldberg. Dex-net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics. In *Robotics: Science and Systems XIII, Massachusetts Institute of Technology, Cambridge, Massachusetts, USA, July 12-16, 2017*, 2017.

[85] Andre Martins and Ramon Astudillo. From softmax to sparsemax: A sparse model of attention and multi-label classification. In *Proc. of the International Conference on Machine Learning*, Jun 2016.

[86] P Warwick Millar. The minimax principle in asymptotic statistical theory. In *Ecole d'Eté de Probabilités de Saint-Flour XI—1981*, pages 75–265. Springer, 1983.

[87] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, Martin A. Riedmiller, Andreas Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Daan Wierstra, Shane Legg, and Demis Hassabis. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015.

[88] Volodymyr Mnih, Adrià Puigdomènech Badia, Mehdi Mirza, Alex Graves, Timothy P. Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016*, pages 1928–1937, New York City, NY, USA, 2016.

[89] Arsalan Mousavian, Clemens Eppner, and Dieter Fox. 6-dof graspnet: Variational grasp generation for object manipulation. *CoRR*, abs/1905.10520, 2019. URL `http://arxiv.org/abs/1905.10520`.

[90] Ofir Nachum, Mohammad Norouzi, Kelvin Xu, and Dale Schuurmans. Bridging the gap between value and policy based reinforcement learning. In *Advances in Neural Information Processing Systems 30 NeurIPS 2017*, pages 2772–2782, Long Beach, CA, USA, 2017.

[91] Gergely Neu, Anders Jonsson, and Vicenç Gómez. A unified view of entropy-regularized markov decision processes. *arXiv preprint arXiv:1705.07798*, 2017.

[92] Andrew Y. Ng and Stuart J. Russell. Algorithms for inverse reinforcement learning. In *Proc. of the International Conference on Machine Learning*, Jun 2000.

[93] Laura Niss and Ambuj Tewari. What you see may not be what you get: UCB bandit algorithms robust to ϵ-contamination. *CoRR*, abs/1910.05625, 2019. URL `http://arxiv.org/abs/1910.05625`.

[94] Brendan O'Donoghue, Rémi Munos, Koray Kavukcuoglu, and Volodymyr Mnih. PGQ: combining policy gradient and q-learning. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2017. URL `https://openreview.net/forum?id=B1kJ6H9ex`.

[95] C Paciorek and M Schervish. Nonstationary covariance functions for Gaussian process regression. In *Proc. of the Advances in Neural Information Processing Systems*, volume 16, pages 273–280, 2004.

[96] Xue Bin Peng, Glen Berseth, KangKang Yin, and Michiel van de Panne. Deeploco: dynamic locomotion skills using hierarchical deep reinforcement learning. *ACM Trans. Graph.*, 36(4):41:1–41:13, 2017.

[97] Daniel J Preston, Haihui Joy Jiang, Vanessa Sanchez, Philipp Rothemund, Jeff Rawson, Markus P Nemitz, Won-Kyu Lee, Zhigang Suo, Conor J Walsh, and George M Whitesides. A soft ring oscillator. *Science Robotics*, 4(31):eaaw5496, 2019.

[98] Martin L Puterman. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.

[99] Charles Ruizhongtai Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 77–85, 2017.

[100] Deirdre Quillen, Eric Jang, Ofir Nachum, Chelsea Finn, Julian Ibarz, and Sergey Levine. Deep reinforcement learning for vision-based robotic grasping: A simulated comparative evaluation of off-policy methods. In *2018 IEEE International Conference on Robotics and Automation, ICRA 2018, Brisbane, Australia, May 21-25, 2018*, pages 6284–6291, 2018.

[101] Joaquin Quiñonero-Candela and Carl Edward Rasmussen. A unifying view of sparse approximate gaussian process regression. *Journal of Machine Learning Research*, 6(13):1939–1959, 2005.

[102] Shankarachary Ragi and Edwin K. P. Chong. UAV path planning in a dynamic environment via partially observable markov decision process. *IEEE Trans. Aerospace and Electronic Systems*, 49(4):2397–2412, Oct 2013.

[103] Deepak Ramachandran and Eyal Amir. Bayesian inverse reinforcement learning. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence*, January 2007.

[104] C.E. Rasmussen and C.K.I. Williams. *Gaussian processes for machine learning*, volume 1. MIT press Cambridge, MA, 2006.

[105] Nathan D. Ratliff, J. Andrew Bagnell, and Martin Zinkevich. Maximum margin planning. In *Proc. of the 23rd International Conference on Machine learning*, June 2006.

[106] Nathan D. Ratliff, David M. Bradley, J. Andrew Bagnell, and Joel E. Chestnutt. Boosting structured prediction for imitation learning. In *Advances in Neural Information Processing Systems 19*. MIT Press, December 2007.

[107] Nathan D Ratliff, David Silver, and J Andrew Bagnell. Learning to search:

Functional gradient techniques for imitation learning. *Autonomous Robots*, 27(1):25–53, 2009.

[108] Stéphane Ross. *Interactive Learning for Sequential Decisions and Predictions*. PhD thesis, Carnegie Mellon University, 2013.

[109] Stéphane Ross and Drew Bagnell. Efficient reductions for imitation learning. In *Proc. of the 13rd International Conference on Artificial Intelligence and Statistics*. JMLR.org, may 2010.

[110] Vishal Satish, Jeffrey Mahler, and Ken Goldberg. On-policy dataset synthesis for learning robot grasping policies using fully convolutional deep networks. *IEEE Robotics and Automation Letters*, 4(2):1357–1364, 2019.

[111] Craig Saunders, Alexander Gammerman, and Volodya Vovk. Ridge regression learning algorithm in dual variables. In *Proc. of the 15th International Conference on Machine Learning*, July 1998.

[112] Tom Schaul, John Quan, Ioannis Antonoglou, and David Silver. Prioritized experience replay. *arXiv preprint arXiv:1511.05952*, 2015.

[113] John Schulman, Sergey Levine, Pieter Abbeel, Michael I. Jordan, and Philipp Moritz. Trust region policy optimization. In *Proceedings of the 32nd International Conference on Machine Learning*, pages 1889–1897, July 2015.

[114] John Schulman, Philipp Moritz, Sergey Levine, Michael I. Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. *CoRR*, abs/1506.02438, 2015. URL `http://arxiv.org/abs/1506.02438`.

[115] John Schulman, Pieter Abbeel, and Xi Chen. Equivalence between policy gradients and soft q-learning. *arXiv preprint arXiv:1704.06440*, 2017.

[116] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *CoRR*, abs/1707.06347, 2017. URL `http://arxiv.org/abs/1707.06347`.

[117] 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 (NeurIPS)*, December 2018.

[118] Kyriacos Shiarlis, Joao Messias, Maarten van Someren, and Shimon Whiteson. Inverse reinforcement learning from failure. In *RSS 2015: Proc. of the 2015 Robotics: Science and Systems Conference, Workshop on Learning from Demonstration*, July 2015.

[119] David Silver, Aja Huang, Chris J. Maddison, Arthur Guez, Laurent Sifre, George van den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Vedavyas Panneershelvam, Marc Lanctot, Sander Dieleman, Dominik Grewe, John Nham, Nal Kalchbrenner, Ilya Sutskever, Timothy P. Lillicrap, Madeleine Leach, Koray Kavukcuoglu, Thore Graepel, and Demis Hassabis. Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016.

[120] David Roger Smart. *Fixed point theorems*, volume 66. CUP Archive, 1980.

[121] Russell Smith et al. *Open dynamics engine.* 2005.

[122] Richard S. Sutton and Andrew G. Barto. *Reinforcement learning - an introduction.* Adaptive computation and machine learning. MIT Press, 1998.

[123] Taiji Suzuki. Fast generalization error bound of deep learning from a kernel perspective. In *International Conference on Artificial Intelligence and Statistics, AISTATS 2018, 9-11 April 2018, Playa Blanca, Lanzarote, Canary Islands, Spain*, pages 1397–1406, 2018.

[124] Umar Syed and Robert E Schapire. A game-theoretic approach to apprenticeship learning. In *Advances in Neural Information Processing Systems*, pages 1449–1456, 2008.

[125] Umar Syed, Michael Bowling, and Robert E Schapire. Apprenticeship learning using linear programming. In *Proceedings of the 25th international conference on Machine learning*, pages 1032–1039. ACM, 2008.

[126] Umar Syed, Michael H. Bowling, and Robert E. Schapire. Apprenticeship learning using linear programming. In *Machine Learning, Proceedings of the Twenty-Fifth International Conference (ICML 2008)*, pages 1032–1039, Helsinki, Finland, 2008.

[127] Csaba Szepesvári. *Algorithms for Reinforcement Learning.* Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers, 2010.

[128] Ben Taskar, Vassil Chatalbashev, Daphne Koller, and Carlos Guestrin. Learning structured prediction models: A large margin approach. In *Proc. of the 22nd International Conference on Machine learning*, August 2005.

[129] Andreas ten Pas, Marcus Gualtieri, Kate Saenko, and Robert Platt Jr. Grasp pose detection in point clouds. *I. J. Robotics Res.*, 36(13-14):1455–1473, 2017.

[130] 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.

[131] Thomas George Thuruthel, Egidio Falotico, Federico Renda, and Cecilia Laschi. Model-based reinforcement learning for closed-loop dynamic control of soft robotic manipulators. *IEEE Transactions on Robotics*, 35(1):124–134, 2018.

[132] Josh Tobin, Lukas Biewald, Rocky Duan, Marcin Andrychowicz, Ankur Handa, Vikash Kumar, Bob McGrew, Alex Ray, Jonas Schneider, Peter Welinder, Wojciech Zaremba, and Pieter Abbeel. Domain randomization and generative models for robotic grasping. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2018, Madrid, Spain, October 1-5, 2018*, pages 3482–3489, 2018.

[133] Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *Proc. of the International Conference on Intelligent Robots and Systems*, Oct 2012.

[134] Michel Tokic and Günther Palm. Value-difference based exploration: Adaptive control between epsilon-greedy and softmax. In *KI 2011: Advances in Artificial Intelligence, 34th Annual German Conference on AI*, Oct 2011.

[135] Constantino Tsallis. Possible generalization of boltzmann-gibbs statistics. *Journal of statistical physics*, 52(1-2):479–487, 1988.

[136] 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.

[137] Michal Valko, Mohammad Ghavamzadeh, and Alessandro Lazaric. Semi-supervised apprenticeship learning. In *Proc. of the Tenth European Workshop on Reinforcement Learning*. JMLR.org, June 2012.

[138] Peter Vamplew, Richard Dazeley, and Cameron Foale. Softmax exploration strategies for multiobjective reinforcement learning. *Neurocomputing*, 263: 74–86, Jun 2017.

[139] Hado van Hasselt, Arthur Guez, and David Silver. Deep reinforcement learning with double q-learning. In *Proc. of the Thirtieth AAAI Conference on Artificial Intelligence*, Feb 2016.

[140] Weiran Wang and Miguel A Carreira-Perpinán. Projection onto the probability simplex: An efficient algorithm with a simple proof, and an application. *arXiv preprint arXiv:1309.1541*, 2013.

[141] Ziyu Wang, Josh S. Merel, Scott E. Reed, Nando de Freitas, Gregory Wayne, and Nicolas Heess. Robust imitation of diverse behaviors. In *Advances in Neural Information Processing Systems*, pages 5326–5335, December 2017.

[142] Christopher JCH Watkins and Peter Dayan. Q-learning. *Machine Learning*, 8(3-4):279–292, May 1992.

[143] Linda Wright, G. Muraleedharan, Carlos Guedes Soares, and Cláudia Lucas. *Characteristic and Moment Generating Functions of Generalised Extreme Value Distribution (GEV)*, pages 269–276. 01 2010. ISBN 978-1-61728-655-1.

[144] Markus Wulfmeier, Peter Ondruska, and Ingmar Posner. Maximum entropy deep inverse reinforcement learning. *arXiv preprint arXiv:1507.04888*, 2015.

[145] Xinchen Yan, Mohi Khansari, Jasmine Hsu, Yuanzheng Gong, Yunfei Bai, Sören Pirk, and Honglak Lee. Data-efficient learning for sim-to-real robotic grasping using deep point cloud prediction networks. *CoRR*, abs/1906.08989, 2019. URL `http://arxiv.org/abs/1906.08989`.

[146] JJ Ye. Constraint qualifications and necessary optimality conditions for optimization problems with variational inequality constraints. *SIAM Journal on Optimization*, 10(4):943–962, 2000.

[147] Brayan S. Zapata-Impata. Using geometry to detect grasping points on 3d unknown point cloud. In *Proceedings of the 14th International Conference on Informatics in Control, Automation and Robotics, ICINCO 2017, Madrid, Spain, July 26-28, 2017, Volume 2.*, pages 154–161, 2017.

[148] Andy Zeng, Shuran Song, Kuan-Ting Yu, Elliott Donlon, Francois Robert Hogan, Maria Bauzá, Daolin Ma, Orion Taylor, Melody Liu, Eudald Romo, Nima Fazeli, Ferran Alet, Nikhil Chavan Dafle, Rachel Holladay, Isabella Morona, Prem Qu Nair, Druck Green, Ian Taylor, Weber Liu, Thomas A. Funkhouser, and Alberto Rodriguez. Robotic pick-and-place of novel objects in clutter with multi-affordance grasping and cross-domain image matching. In *2018 IEEE International Conference on Robotics and Automation, ICRA 2018, Brisbane, Australia, May 21-25, 2018*, pages 1–8, 2018.

[149] Tianhao Zhang, Zoe McCarthy, Owen Jow, Dennis Lee, Xi Chen, Ken Goldberg, and Pieter Abbeel. Deep imitation learning for complex manipulation tasks from virtual reality teleoperation. In *2018 IEEE International Conference on Robotics and Automation, ICRA 2018, Brisbane, Australia, May 21-25, 2018*, pages 1–8, 2018.

[150] Jiangchuan Zheng, Siyuan Liu, and Lionel M. Ni. Robust bayesian inverse reinforcement learning with sparse behavior noise. In *Proc. of the 28th AAAI Conference on Artificial Intelligence*. AAAI Press, July 2014.

[151] Brian D Ziebart. *Modeling purposeful adaptive behavior with the principle of maximum causal entropy*. PhD thesis, Carnegie Mellon University, 2010.

[152] Brian D Ziebart, Andrew L Maas, J Andrew Bagnell, and Anind K Dey. Maximum entropy inverse reinforcement learning. In *AAAI*, volume 8, pages 1433–1438. Chicago, IL, USA, 2008.

[153] Christoph Zimmer, Mona Meister, and Duy Nguyen-Tuong. Safe active learning for time-series modeling with gaussian processes. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada.*, pages 2735–2744, 2018.

초 록

본 학위 논문에서는 시범과 보상함수를 기반으로한 로봇 학습 문제를 다룬다. 로봇 학습 방법은 불확실하고 복잡 업무를 잘 수행 할 수 있는 최적의 정책 함수를 찾는 것을 목표로 한다. 로봇 학습 분야의 다양한 문제 중에, 샘플 복잡도를 줄이는 것에 집중한다. 특히, 효율적인 탐색 방법과 혼합 시범으로 부터의 학습 기법을 개발하여 적은 수의 샘플로도 높은 효율을 갖는 정책 함수를 학습하는 것이 목표이다.

효율적인 탐색 방법을 개발하기 위해서, 우리는 일반화된 쌀리스 엔트로피를 사용한다. 쌀리스 엔트로피는 샤논-깁스 엔트로피를 일반화한 개념으로 엔트로픽 인덱스라는 새로운 파라미터를 도입한다. 엔트로픽 인덱스를 조절함에 따라 다양한 형태의 엔트로피를 만들어 낼 수 있고 각 엔트로피는 서로 다른 레귤러라이제이션 효과를 보인다. 이 성질을 기반으로, 스파스 마르코프 결정과정을 제안한다. 스파스 마르코프 결정과정은 스파스 쌀리스 엔트로피를 이용하여 희소하면서 동시에 다모드의 정책 분포를 표현하는데 효과적이다. 이를 통해서 샤논-깁스 엔트로피를 사용하였을때에 비해 더 좋은 성능을 갖음을 수학적으로 증명하였다. 또한 스파스 쌀리스 엔트로피로 인한 성능 저하를 이론적으로 계산하였다. 스파스 마르코프 결정과정을 더욱 일반화시켜 일반화된 쌀리스 엔트로피 결정과정을 제안하였다. 마찬가지로 쌀리스 엔트로피를 마르코프 결정과정에 추가함으로써 생기는 최적 정책함수의 변화와 성능 저하를 수학적으로 증명하였다. 나아가, 성능저하를 없앨 수 있는 방법인 엔트로픽 인덱스 스케쥴링을 제안하였고 실험적으로 최적의 성능을 갖음을 보였다.

또한, 헤비테일드 잡음이 있는 학습 문제를 해결하기 위해서 외란(Perturbation)을 이용한 탐색 기법을 개발하였다. 로봇 학습의 많은 문제는 잡음의 영향이 존재한다. 학습 신호안에 다양한 형태로 잡음이 들어있는 경우가 있고 이러한 경우에 잡음을 제거 하면서 최적의 행동을 찾는 문제는 효율적인 탐사 기법을 필요로 한다. 기존의 방법론들은 서브 가우시안(sub-Gaussian) 잡음에만 적용 가능했다면, 본 학위 논문에서 제안한 방식은 헤비테일드 잡음을 해결 할 수 있다는 점에서 기존의 방법론들보다 장점을 갖는다. 먼저, 일반적인 외란에 대해서 리그렛 바운드를

증명하였고 외란의 누적분포함수(CDF)와 리그렛 사이의 관계를 증명하였다. 이 관계를 이용하여 다양한 외란 분포의 리그렛 바운드를 계산 가능하게 하였고 다양한 분포들의 가장 효율적인 탐색 파라미터를 계산하였다.

혼합시범으로 부터의 학습 기법을 개발하기 위해서, 오시범을 다룰 수 있는 새로운 형태의 가우시안 프로세스 회귀분석 방식을 개발하였고, 이 방식을 확장하여 레버리지 가우시안 프로세스 역강화학습 기법을 개발하였다. 개발된 기법에서는 정시범으로부터 무엇을 해야 하는지와 오시범으로부터 무엇을 하면 안되는지를 모두 학습할 수 있다. 기존의 방법에서는 쓰일 수 없었던 오시범을 사용 할 수 있게 만듦으로써 샘플 복잡도를 줄일 수 있었고 정제된 데이터를 수집하지 않아도 된다는 점에서 큰 장점을 갖음을 실험적으로 보였다.

주요어: 로봇 학습, 실시간 학습, 강화 학습, 역강화 학습, 시범 기반 학습, 모방 학습

학 번: 2015-20957

감사의 글

6년간의 대학원 생활 끝에 공학박사로 졸업을 하게 되었습니다. 이제 더 이상 학생의 신분이 아니라는 생각에 설렘도 생기고 두려움도 생기며 관악을 떠날 준비를 하니 지난 대학원 생활을 돌이켜보는 시간을 갖게 되었습니다. 제가 박사 학위를 받기까지 그 동안 많은 분들께 도움을 받았다는 사실을 세삼 깨닫게 되었습니다. 그래서 그 동안 제가 한 사람의 연구자로써 성장 할 수 있도록 도움을 주신 분들께 감사의 말씀을 전하고 싶습니다.

우선 지난 대학원 생활 동안 저를 이끌어 주시고 지도해주신 오성회 교수님께 깊은 감사의 말씀 드리고 싶습니다. 교수님께 지도 받으면서 한 분야의 박사란 무엇인가, 연구란 무엇인가를 깊이 있게 고민해보고 배울 수 있었던 시간이었습니다. 특히, 주어진 문제를 해결하는 것이 아닌, 새로운 문제를 제시 할 수있는 능력이 필요하다는 것을 알았고 그러한 능력을 키우기 위해 부단히 노력했던 것 같습니다. 이런 고민을 함께 해주신 교수님께 감사의 말씀 드립니다. 또한, 영어가 부족하여 고생했던 저에게 충분한 시간을 주시고 기다려주셨던 교수님의 배려에 감사드리고 싶습니다. 덕분에 제가 이렇게 공학박사로써 졸업 할 수 있었습니다. 대학원 생활 동안 함께 세미나를 진행하며 지속적으로 연구 발표에 대해 지도 해주셨고 뿐만 아니라 더 나은 학위 논문을 위해서 지도해 주셨던 최진영 교수님께도 감사의 말씀 드립니다. 또한 바쁘신 와중에도 학위 논문 지도를 위해 시간을 내주신 심형보 교수님께도 감사의 말씀 드립니다. 교수님들의 지도 덕분에 학위 논문을 잘 마무리 지을 수 있었습니다.

그리고 대학원 생활을 즐겁게 할 수 있도록 도와준 연구실 동료들에게도 감사의 인사를 전합니다. 가장 먼저, 바쁘신 와중에도 저의 박사 학위 논문을 지도해주시고 심사해 주신 성준이형과 은우형께 감사의 말씀 드립니다. 성준이형에게는 대학원에 입학하여 졸업 할 때까지 많은 것을 보고 배운 것 같습니다. 연구를 대하는 자세와 열정은 후배로써 존경스러웠고 제가 공부하고 연구 할 때 좋은 본보기가 되었습니다. 그리고 항상 차분히 맡은 바를 열심히 해나가신 은우형께도 평정심과 차분한 마음가짐을 배울 수 있었습니다. 앞으로도 잘 부탁드립니다. 또한, 훌륭한 연구를

수행하시고 앞서 졸업하신 졸업생 선배님들께 감사합니다. 힘들때 마다 고민을 들어주시고 버팀목이 되어 주셨던 선배님들, 정훈이형, 정찬이형, 인환이형, 동훈이형, 윤선누나, 준식이형 감사드립니다. 그리고 박사 졸업 직전까지 함께 연구실 생활을 했던 건호형, 혜민누나, 경훈이형 감사드립니다. 힘든 일이나 고민이 있을 때 함께 걸으며 이야기를 들어주었던 동기 승규형 감사합니다. 항상 묵묵히 맡은 역할을 잘 하는 친구 찬호, 연구실의 방장을 도맡아한 누리, 전문연으로써 같은 고민을 나누던 휘연이, 윤호, 디모데, 연구실의 재간둥이 재구를 비롯하여 오빈, 민의, 건민, 민재까지 연구실 모두에게 감사의 말씀 드립니다.

무뚝뚝한 아들의 선택을 항상 믿고 응원해 주시는 부모님께도 감사의 말씀 드리고 싶습니다. 부모님의 지원과 믿음 그리고 사랑이 없었다면 힘든 박사과정 생활을 잘 이겨내지 못했을 것 입니다. 세상 물정 모르던 아들이 이제 학교를 떠나 사회로 나갑니다. 앞으로도 잘 할 수 있도록 노력하겠습니다. 지켜봐 주세요.

끝으로 학생이었던 저를 만나 결혼까지 하게된 저의 아내 은진이에게 감사의 말씀을 전하고 싶습니다. 힘들때마다 때로는 위로를 때로는 격려를 때로는 따끔한 충고를 해주었던 은진이가 있었기에 오늘의 제가 있지 않나 생각해봅니다. 은진이는 박사과정동안 제게 평안한 안식처같은 존재 였습니다. 이제는 길었던 공부를 마치고 제가 은진이의 안식처가 되고 싶습니다. 사랑한다.