Towards Adversarial Robustness of Sequential Decision Making Algorithms

By

GUANLIN LIU

DISSERTATION

Submitted in partial satisfaction of the requirements for the degree of

DOCTOR OF PHILOSOPHY

in

Electrical and Computer Engineering

in the

OFFICE OF GRADUATE STUDIES

of the

UNIVERSITY OF CALIFORNIA

DAVIS

Approved:

Lifeng Lai, Chair

Junshan Zhang

Khaled Abdel-Ghaffar

Committee in Charge

2024

# Abstract

Reinforcement Learning (RL) is a framework for control-theoretic problems that make decisions over time under uncertain environments. RL has many applications in a variety of scenarios such as displaying advertisements, articles recommendation, cognitive radios, and search engines, to name a few. The growing applications of RL in security and safety-critical areas, such as large language models and autonomous driving, highlight the need for adversarially robust RL and motivate this work. In order to develop trustworthy machine learning systems, we make progress in understanding adversarial attacks on learning systems and correspondingly building robust defense mechanisms.

In this dissertation, we discuss our work in mainly five increasingly complex scenarios.

Firstly, we introduce a new class of attacks named action manipulation attacks on stochastic multi-armed bandits, which is special class of RL problem with only one state in the state space. In this class of attacks, an adversary can change the action signal selected by the user. We show that without knowledge of mean rewards of arms, our proposed attack can manipulate Upper Confidence Bound (UCB) algorithm into pulling a target arm very frequently by spending only logarithmic cost. To defend against this class of attacks, we introduce a novel algorithm that is robust to action-manipulation attacks when an upper bound for the total attack cost is given. We prove that our algorithm has a pseudo-regret upper bounded by $\mathcal{O}(\max\{\log(T), A\})$ with a high probability, where $T$ is the total number of rounds and $A$ is the upper bound of the total attack cost.

Secondly, we design action poisoning attack schemes against linear contextual bandit algorithms in both white-box and black-box settings. Contextual bandits are a class of problems that sit between the stochastic multi-armed bandits and the general RL. In contextual bandits, one learns in different states, but the state transition is independent on the agent's action and the state. We analyze the cost of the proposed attack strategies for a very popular and widely used bandit algorithm: LinUCB. We further extend our proposed attack strategy to generalized linear models.

Thirdly, building on the work on multi-arm bandits and contextual bandits, we extend the

study to the general RL. We study the action poisoning attack in both white-box and black-box settings. We introduce an adaptive attack scheme called LCB-H, which works for most RL agents in the black-box setting. We prove that the LCB-H attack can force any efficient RL agent, whose dynamic regret scales sublinearly with the total number of steps taken, to choose actions by following a target policy. In addition, we apply LCB-H attack against a popular model-free RL algorithm: UCB-H. We show that, even in the black-box setting, by spending only logarithm cost, the proposed LCB-H attack scheme can force the UCB-H agent to choose actions according to the policy selected by the attacker very frequently.

Fourthly, we broaden the study to the multi-agent RL (MARL) problem. We investigate the impact of adversarial attacks on MARL. In the considered setup, there is an exogenous attacker who is able to modify the rewards before the agents receive them or manipulate the actions before the environment receives them. The attacker aims to guide each agent into a target policy or maximize the cumulative rewards under some specific reward function chosen by the attacker, while minimizing the amount of manipulation on feedback and action. We first show the limitations of the action poisoning only attacks and the reward poisoning only attacks. We then introduce a mixed attack strategy with both the action poisoning and the reward poisoning. We show that the mixed attack strategy can efficiently attack MARL agents even if the attacker has no prior information about the underlying environment and the agents' algorithms.

Finally, building on the insights from the adversarial attacks on RL, we design a robust RL algorithm, which aims to find a policy that optimizes the worst-case performance in the face of uncertainties. we focus on action robust RL with the probabilistic policy execution uncertainty, in which, instead of always carrying out the action specified by the policy, the agent will take the action specified by the policy with probability $1 - \rho$ and an alternative adversarial action with probability $\rho$. We show the existence of an optimal policy on the action robust MDPs with probabilistic policy execution uncertainty and provide the action robust Bellman optimality equation for its solution. Based on that, we develop Action Robust Reinforcement Learning with Certificates (ARRLC) algorithm that achieves minimax optimal regret and sample complexity.

# Acknowledgement

I am incredibly grateful to all those who have supported and encouraged me throughout my doctoral journey.

Firstly, I express my deepest gratitude to my advisor, Professor Lifeng Lai. His guidance, mentorship, and unwavering support have been instrumental in shaping my research trajectory and academic growth. I am particularly grateful to him for introducing me to the field of reinforcement learning. He not only opened my eyes to the potential of reinforcement learning, but also instilled in me a strong foundation for further exploration. His research passion truly sparked my interest and motivated me to dive deeper into this field. He also worked on my papers a lot, to improve my writing styles and the organization of papers. I am incredibly grateful for the opportunity to have learned under your mentorship.

Apart from my advisor, I would like to thank other committee members, Prof. Junshan Zhang and Prof. Khaled Abdel Ghaffar, for their effort spent in my research. They provided fruitful comments and insightful feedback, which helped me to improve my results.

I wish to show my appreciation to everyone in my research group, Yulu Jin, Xinyi Ni, Parisa Oftadeh, Puning Zhao, Fuwei Li, Minhui Huang, Xinyang Cao, Xiaochuan Ma, Chenye Yang and Haodong Liang, for their great support and technical advice. I am also deeply grateful to my co-authors, Zhihan Zhou, Ziqing Lu, and Weiyu Xu. Thank all my friends for their support and encouragement throughout this journey.

My heart overflows with gratitude for my parents, Fanyun Liu and Dong bing Xie. I am eternally grateful for the sacrifices they made to ensure I received the best education possible. This accomplishment would not have been possible without them.

To my wife, Perry Yang, my deepest thanks and affection. She has been my rock throughout this entire journey. Her unwavering support, love, and understanding have provided me with the strength and motivation to persevere during challenging times. She is an essential part of my success.

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

In order to develop trustworthy machine learning systems, understanding adversarial attacks on learning systems and correspondingly building robust defense mechanisms have attracted significant recent research interests [4, 9, 29, 37, 49, 55, 74, 108]. Reinforcement learning (RL), a framework for control-theoretic problems that make decisions over time under uncertain environment, has many applications in a variety of scenarios. As RL models are being increasingly used in safety critical and security related applications, it is critical to understand the effects of adversarial attacks on RL systems in order to develop trust-worthy RL systems. While there are many existing works addressing adversarial attacks on supervised learning models [3,13,16,17,20,22,29,46,50,75,80,95,101,102,114], the understanding of adversarial attacks on RL models is less complete. The goal of this dissertation is to fill in the gap and develop robust RL algorithms that can tolerate adversarial attacks.

In this chapter, we introduce the background of this dissertation. In Chapter 1.1, we introduce basic concepts used in this dissertation. In Chapter 1.2, we introduce the adversarial attacks on stochastic bandits and a robust stochastic multi-armed bandits algorithm that can defend the action poisoning attacks. In Chapter 1.3, we introduce the adversarial attacks on linear contextual bandits and generalized linear contextual bandits. In Chapter 1.4, we introduce the adversarial attacks on RL. In Chapter 1.5, we introduce the adversarial attacks on multi-agent reinforcement learning

(MARL). In Chapter 1.6, we introduce the action robust RL with probabilistic policy execution uncertainty.

## 1.1 Preliminaries

RL is a framework for control-theoretic problems that make decisions over time under uncertain environment. RL problems aim to directly construct algorithms that learn from interactions to achieve a goal. The learner or decision maker is called an agent. The thing it interacts with is called the environment, which includes everything outside the agent. The agent and the environment continually interact. The agent chooses actions and the environment responds to these actions and presents the agent with new situations. The environment also generates rewards. The agent aims to maximize the total rewards over time. A task, an instance of a RL problem, is defined by a complete specification of an environment.

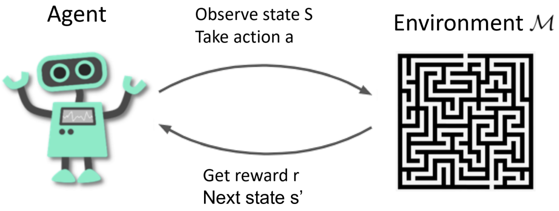


Figure 1.1: The agent–environment interaction in Reinforcement Learning.

More specifically, we denote a tabular episodic Markov decision process (MDP) as a tuple $\mathcal{M} = (\mathcal{S}, \mathcal{A}, H, P, R)$, where $\mathcal{S}$ is the state space with $|\mathcal{S}| = S$, $\mathcal{A}$ is the action space with $|\mathcal{A}| = A$, $H \in \mathbb{Z}^+$ is the number of steps in each episode (planning horizon), $P_h : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ is the probability transition function which maps state-action-state pair to a probability, $R_h : \mathcal{S} \times \mathcal{A} \to [0, 1]$ represents the reward function in the step $h$. In general, the probability transition functions and the reward functions can be different over steps.

The agent interacts with environment in a sequence of episodes. In each episode $k$ of this MDP, $s_1$ is generated randomly by a distribution or chosen by the environment. Initial states may be different between episodes. We define $[H] := \{1, \cdots, H\}$ to denote the set of integers from $1$ to $H$. At each step $h \in [H]$ in an episode,, the agent observes the state $s_h$ and chooses an action $a_h$. After receiving the action, the environment generates a random reward $r_h \in [0, 1]$ derived from a random distribution with mean $R_h(s_h, a_h)$ and next state $s_{h+1}$ which is drawn from the distribution $P_h(\cdot|s, a)$. $P_h(\cdot|s, a)$ represents the probability distribution over states if action $a$ is taken for state $s$. The agent stops interacting with environment after $H$ steps and start another episode.

The policy $\pi$ of agent is expressed as a mapping $\pi : \mathcal{S} \times [H] \to \mathcal{A}$. For notational convenience, we use $\pi_h(s)$ to denote $\pi(s, h)$. Interacting with the environment $\mathcal{M}$, the policy induces a random trajectory $\{s_1, a_1, r_1, s_2, a_2, r_2, \cdots, s_H, a_H, r_H, s_{H+1}\}$, where $s_1$ is the initial state, $a_h = \pi_h(s_h)$, $r_h$ is derived from a random distribution with mean $R_h(s_h, a_h)$, and $s_{h+1} \sim P_h(\cdot|s_h, a_h)$ for each $h$. RL agents learn to maximize the expected cumulative reward $\mathbb{E}[\sum_{h=1}^{H} r_h]$.

We use $V_h^\pi : \mathcal{S} \to \mathbb{R}$ to denote the value function at step $h$ under policy $\pi$, so that $V_h^\pi(s)$ gives the expected sum of remaining rewards received under policy $\pi$, starting from $s_h = s$, until the end of the episode. Accordingly, we also use $Q_h^\pi : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ to denote the $Q$-function at step $h$ under policy $\pi$, so that $Q_h^\pi(s, a)$ gives the expected sum of remaining rewards received under policy $\pi$, starting from $s_h = s$, $a_h = a$, until the end of the episode. In symbols:

$$V_h^\pi(s) = \mathbb{E}\left[\sum_{h'=h}^{H} r_{h'} \middle| s_h = s, \pi\right],\ Q_h^\pi(s, a) = \mathbb{E}\left[\sum_{h'=h}^{H} r_{h'} \middle| s_h = s, a_h = a, \pi\right]. \tag{1.1}$$

These functions represent the expected total rewards received from step $h$ to $H$, under policy $\pi$, starting from state $s$ and state-action pair $(s, a)$ respectively.

The value function and $Q$-function satisfy the Bellman consistency equations [41]:

$$\begin{aligned} V_h^\pi(s) &= Q_h^\pi(s, \pi(s)), \\ Q_h^\pi(s, a) &= R_h(s, a) + \mathbb{E}_{s' \sim P_h(\cdot|s,a)}[V_{h+1}^\pi(s')]. \end{aligned} \tag{1.2}$$

For simplicity, we denote $V_{H+1}^\pi = \mathbf{0}$, $Q_{H+1}^\pi = \mathbf{0}$ and $P_h V_{h+1}^\pi(s, a) = \mathbb{E}_{s' \sim P_h(\cdot|s,a)}[V_{h+1}^\pi(s')]$.

Under mild technical assumptions, there exists an optimal policy $\pi^*$ such that $\pi^*$ maximizes the value function and $Q$-function:

$$\begin{aligned} V_h^{\pi^*}(s) &= V_h^*(s) = \sup_\pi V_h^\pi(s), \\ Q_h^{\pi^*}(s,a) &= Q_h^*(s,a) = \sup_\pi Q_h^\pi(s,a), \end{aligned} \tag{1.3}$$

for all $s$, $a$ and $h$.

$$\text{Regret}(K) = \sum_{k=1}^{K} [V_1^*(s_1^k) - V_1^{\pi_h^k}(s_1^k)], \tag{1.4}$$

where $s_1^k$ is the initial state for each episode $k$ and $\pi^k$ is the control policy followed by the agent at episode $k$.

Among the general RL problems, there are classes of special cases that are simpler but are still useful in practice. The first class of such special cases is multi-armed bandits that involve learning to act in only one situation. In other words, stochastic multi-armed bandits are RL problem with only one state in the state space. More specifically, in each round $t = 1, 2, 3, \ldots, T$, the user pulls an arm (or action) $I_t \in \{1, \ldots, K\}$ and receives a random reward $r_t$ drawn from the reward distribution of arm $I_t$. The user aims to maximize the cumulative rewards over $T$ rounds.

Another special case is contextual bandit problems that sit somewhere in between the stochastic multi-armed bandits and general RL. It learns in different states, but the state transition is independent on the agent's action and the state. More specifically, in each round $t = 1, 2, 3, \ldots, T$, the agent observes a context $x_t \in \mathcal{D}$ where $\mathcal{D} \subset \mathbb{R}^d$, pulls an arm $I_t$ and receives a reward $r_{t,I_t}$. Each arm $i$ is associated with an unknown but fixed coefficient vector $\theta_i \in \Theta$ where $\Theta \subset \mathbb{R}^d$. In each round $t$, the reward depends on both $x_t$ and $\theta_i$.

RL has many applications in a variety of scenarios such as recommendation systems [119], autonomous driving [78], finance [64] and business management [76], to name a few. As RL models are being increasingly used in safety critical and security related applications, it is critical to develop trustworthy RL systems. For example, in recommendation systems, the transitions of the decisions and the reward signal rely on a feedback loop between the recommendation system and the user. A restaurant may attack the recommendation systems to force the systems into increasing the restaurant's exposure. Such attacks can disrupt the users' experience and cause damage to the recommendation company's interests. Another example is self-driving car. If a car's self-driving system is built on RL, the attacker may be able to implement destabilizing forces or manipulate the action signal, so as to change the brake force. This can cause a car accident and bring a serious threat to life and property safety. In this project, we focus on the following research question:

- Should we trust the decision made by an RL agent?
- Can an adversary mislead the RL agent?
- Is there any powerful adversary which can efficiently mislead the RL agent even in the black-box setting?
- Could we design algorithms that archive robustness to adversarial attacks?

Understanding the effects of adversarial attacks on RL systems is the first step towards the goal of safe applications of RL models. In the modern industry-scale applications of RL models, action decisions, reward and state signal collection, and policy iterations are normally implemented in a distributed network. When data packets containing the reward signals and action decisions etc are transmitted through the network, an attacker can intercept and modify these data packets to implement adversarial attacks.

There are some recent interesting work on adversarial attacks against RL algorithms under various setting [8, 38, 60, 69, 83, 84, 92, 116]. Adversarial attacks in online RL differ significantly from adversarial attacks in classical supervised learning and are more difficult due to the following challenges.

Challenge I: Consideration of long-term rewards. In reinforcement learning, the agent aims to maximize the expected cumulative reward instead of the immediate reward. However, actions may affect not only the immediate reward but also the next situation and, through that, all subsequent rewards. Unlike classical supervised learning, there is no examples of correct actions. The agent needs to learn the correct action by considering long-term rewards. The adversary also has to consider long-term rewards to decide the attack strategy, which makes adversarial attacks in online RL challenging.

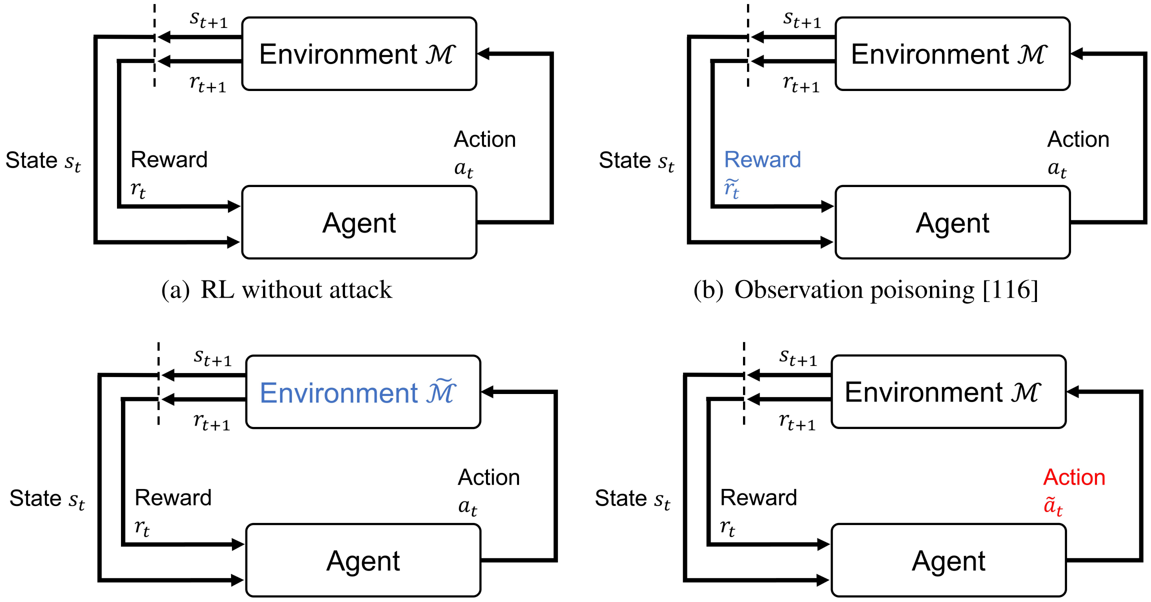
Challenge II: No access to future data. Adversarial attacks in classical supervised learning [46, 51] often require access to the entire training dataset, so the attacker can decide on the optimal attack strategy before learning begins. In online RL, the training data (trajectories) is collected while the agent is learning. The adversary can only access and change the data in the history. The adversary does not know the future data. Since the adversary has to consider long-term rewards, he needs to predict the future data, e.g. the next state, which makes adversarial attacks in online RL challenging.

Challenge III: Unknown dynamics of environment. While challenges I and II can be partially addressed by predicting future trajectories, it requires prior knowledge of the dynamics of the underlying MDP. However, knowing the underlying dynamics of environment is impractical. More generally, the attacker learns the environment only based on the agent's observations. We called this case as black-box attack. In black-box attack, the adversary needs to estimate the underlying dynamics of environment, which makes adversarial attacks in online RL challenging.

As shown in Figure 1.2, there are several different types attacks against RL: observation poisoning attack, environment poisoning attacks and action poisoning attacks. The observation attacks can change the observations of the agent from the environment including the reward signal or the state signal [8, 116]. At the time step $t$, the adversary can replace the true reward $r_t$ by an arbitrary reward $\tilde{r}_t$. By changing the reward, the agent can manipulate the agent's estimation of the environment and then could impact the agent's behavior.

In the environment poisoning setting, the adversary can arbitrarily change both the rewards and the state transition functions [69]. When no corruption happens, the agent faces a nominal MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, H, P, R)$. If the adversary decides to corrupt, the adversary can change $\mathcal{M}$ to $\tilde{\mathcal{M}} = (\mathcal{S}, \mathcal{A}, H, \tilde{P}, \tilde{R})$.

An attacker may introduce action poisoning attacks on RL agent. In particular, at the time step



(a) RL without attack

(b) Observation poisoning [116]

(c) Environment poisoning [83]

(d) Action poisoning [61]

Figure 1.2: Adversarial attacks against reinforcement learning.

$t$, after the agent chooses an action $a_t$, the attacker can change it to another action $\tilde{a}_t \in \mathcal{A}$. Then the environment receives $\tilde{a}_t$ instead of $a_t$, and generates a random reward $r_t$ and the next state $s_{t+1}$ corresponding to the action $\tilde{a}_t$. Note that the agent may not know the attacker's manipulations and the presence of the attacker and hence will still view $r_t$ as the reward and $s_{t+1}$ as the next state generated from state-action pair $(s_t, a_t)$.

In this research dissertation, we study the following problems: 1) adversarial attacks against stochastic bandits and defense strategies; 2) adversarial attacks against contextual bandits; 3) adversarial attacks against reinforcement learning; 4) adversarial attacks against multi-agent reinforcement learning; 5) efficient action robust reinforcement learning with probabilistic policy execution uncertainty.

## 1.2 Attacks on Stochastic Bandits

In this section, we focus on multiple armed bandits (MABs), a simple but very powerful framework of online learning that makes decisions over time under uncertainty. Stochastic multi-armed bandits a special case of the general RL, in which there is only one state in the state space.

In Chapter 2, we will introduce a new class of attacks on MABs named action-manipulation attack. In the action-manipulation attack, an attacker, sitting between the environment and the user, can change the action selected by the user to another action. The user will then receive a reward from the environment corresponding to the action chosen by the attacker. Compared with the reward-manipulation attacks discussed above, the action-manipulation attack is more difficult to carry out. In particular, as the action-manipulation attack only changes the action, it can impact but does not have direct control of the reward signal, because the reward signal will be a random variable drawn from a distribution depending on the action chosen by the attacker. This is in contrast to reward-manipulation attacks where an attacker has direct control and can change the reward signal to any value.

In order to demonstrate the significant security threat of action-manipulation attacks to stochastic bandits, we propose an action-manipulation attack strategy against the widely used UCB algorithm. We choose to attack the UCB algorithm as it is widely used in practice and has been extensively studied in the literature. The proposed attack strategy aims to force the user to frequently pull a target arm chosen by the attacker. We assume that the attacker does not know the true mean reward of each arm. The assumption that the attacker does not know the mean rewards of arms is necessary for the design of attack strategies, as otherwise the attacker can perform the attack trivially. To see this, with the knowledge of the mean rewards, the attacker knows which arm has the worst mean reward and can perform the following oracle attack: when the user pulls a non-target arm, the attacker changes the arm to the worst arm. This oracle attack makes all non-target arms have expected rewards less than that of the target arm, if the target arm selected by the attacker is not the worst arm. In addition, under this attack, all sublinear-regret bandit algorithms will pull the target arm $\mathcal{O}(T)$ times. However, the oracle attack is not practical. The goal of our work is to develop an attack strategy that has similar performance of the oracle attack strategy without requiring the knowledge of the true mean rewards. When the user pulls a non-target arm, the attacker could decide to attack by changing the action to the possible worst arm. As the attacker does not know the true value of arms, our attack scheme relies on lower confidence bounds (LCB) of the value of each arm in making attack decisions. Correspondingly, we name our attack scheme as LCB attack strategy. On the other hand, we also show that, if the target arm is the worst arm and the attacker can only incur logarithmic costs, no attack algorithm can force the user to pull the worst arm more than $T - \mathcal{O}(T^{\alpha})$ times with $0 < \alpha < 1$. In addition, we study an oracle attack to illustrate the challenges arise for the case where the target arm is the worst arm.

Motivated by the analysis of the action-manipulation attacks and the significant security threat to MABs, we then design a bandit algorithm which can defend against the action-manipulation attacks and still is able to achieve a small regret. The main idea of the proposed algorithm is to bound the maximum amount of offset, in terms of user's estimate of the mean rewards, that can be introduced by the action-manipulation attacks. We then use this estimate of maximum offset to properly modify the UCB algorithm and build specially designed high-probability upper bounds of the mean rewards so as to decide which arm to pull. We name our bandit algorithm as maximum offset upper confidence bound (MOUCB). In particular, our algorithm first pulls every arm a certain of times and then pulls the arm whose modified upper confidence bound is the largest. Furthermore, we prove that MOUCB bandit algorithm has a pseudo-regret upper bounded by $\mathcal{O}(\max\{\log T, A\})$, where $T$ is the total number of rounds and $A$ is an upper bound for the total attack cost. In particular, if $A$ scales as $\log(T)$, MOUCB archives a logarithm pseudo-regret which is same as the regret of UCB algorithm.

**Related work:** In this paragraph, we discuss related works. There is a line of interesting recent work on online reward-manipulation attacks on stochastic MABs [43, 56, 65]. In the reward-manipulation attacks, there is an adversary who can change the reward signal from the environment, and hence the reward signal received by the user is not the true reward signal from the environment. In particular, [43] proposes an interesting attack strategy that can force a user, who runs either $\epsilon$-Greedy or Upper Confidence Bound (UCB) algorithm, to select a target arm while only spending effort that grows in logarithmic order. [56] proposes an optimization based framework for offline reward-manipulation attacks. Furthermore, it studies a form of online attack strategy that is effective in attacking any bandit algorithm that has a regret scaling in logarithm order, without knowing what particular algorithm the user is using. [30] considers an attack model where an adversary attacks with a certain probability at each round but its attack value can be arbitrary and unbounded. The paper proposes algorithms that are robust to these types of attacks. [65] considers how to defend against reward-manipulation attacks, a complementary problem to [43, 56]. In particular, [65] introduces a bandit algorithm that is robust to reward-manipulation attacks under certain attack cost, by using a multi-layer approach. [33] presents an algorithm named BARBAR that is robust to reward poisoning attacks and the regret of the proposed algorithm is nearly optimal. [24] introduces another model of adversary setting where each arm is able to manipulate its own reward and seeks to maximize its own expected number of pull count. Under this setting, [24] analyzes the robustness of Thompson Sampling, UCB, and $\epsilon$-greedy under attacks, and proves that all three algorithms achieve a regret upper bound that increases over rounds in a logarithmic order or increases with attack cost in a linear order. This line of reward-manipulation attack has also recently been investigated for contextual bandits in [67], which develops an attack algorithm that can force the bandit algorithm to pull a target arm for a target contextual vector by slightly manipulating rewards in the data.

**Contributions:** The contributions of this work are: (1) we introduce a new class of attacks on MABs named action-manipulation attack. We propose an action-manipulation attack strategy, LCB attack strategy. Our analysis shows that, if the target arm selected by the attacker is not the worst arm, the LCB attack strategy can successfully manipulate the user to select the target arm almost all the time with an only logarithmic cost. In particular, LCB attack strategy can force the user to pull the target arm $T - \mathcal{O}(\log(T))$ times over $T$ rounds, with the total attack cost being only $\mathcal{O}(\log(T))$. (2) We show the necessity of the assumption that the target arm selected by the attacker is not the worst arm. We show that, if the target arm is the worst arm and the attacker can only incur logarithmic costs, no attack algorithm can force the user to pull the worst arm more than $T -$

$\mathcal{O}(T^{\alpha})$ times. (3) We design a bandit algorithm which can defend against the action-manipulation attacks and still is able to achieve a small regret. We prove that MOUCB bandit algorithm has a pseudo-regret upper bounded by $\mathcal{O}(\max\{\log T, A\})$, where $T$ is the total number of rounds and $A$ is an upper bound for the total attack cost. (4) We evaluate our attack strategies and the MOUCB algorithms using synthetic data. MOUCB algorithm archives logarithmic pseudo-regrets under both LCB attacks and the oracle attacks.

The results in this part have been published in [58, 59].

## 1.3 Attacks on Contextual Bandits

Contextual bandits are class of problems that are more complex than multi-armed bandit problems but are still simpler than the general RL. It learns in different states, but the state transition is independent on the agent's action and the state. Existing works on adversarial attacks against linear contextual bandits focus on the reward [26, 67] or context poisoning attacks [26]. In the reward poisoning attacks, the adversary can modify the reward. In the context poisoning attacks, the adversary can modify the context observed by the agent without changing the reward associated with the context.

In Chapter 3, we aim to investigate the impact of action poisoning attacks on contextual bandit models. More detailed comparisons of various types of attacks against contextual bandits will be provided in Chapter 3.1. We note that the goal of this work is not to promote any particular type of poisoning attack. Rather, our goal is to understand the potential risks of action poisoning attacks. We note that for the safe applications and design of robust contextual bandit algorithms, it is essential to address all possible weaknesses of the models and understanding the risks of different kinds of adversarial attacks. Since the action poisoning attack is an important aspect of poisoning attacks and may threaten the bandit systems, it is important to understand the potential risks of action poisoning attacks.

In Chapter 3, we study the action poisoning attack against linear contextual bandit in both white-box and black-box settings. In the white-box setting, we assume that the attacker knows the coefficient vectors associated with arms. Thus, at each round, the attacker knows the mean rewards of all arms. While it is often unrealistic to exactly know the coefficient vectors, the understanding of the white-box attacks could provide valuable insights on how to design the more practical black-box attacks. In the black-box setting, we assume that the attacker has no prior information about the arms and does not know the agent's algorithm. The limited information that the attacker has are the context information, the action signal chosen by the agent, and the reward signal generated from the environment. In both white-box and black-box settings, the attacker aims to manipulate the agent into frequently pulling a target arm chosen by the attacker with a minimum cost. The cost is measured by the number of rounds that the attacker changes the actions selected by the agent.

**Related work:** In this part, we discuss related works on two parts: adversarial attacks that cause standard bandit algorithms to fail and robust bandit algorithms that can defend against such attacks. **Attacks Models.** In linear contextual bandit setting, [67] studies offline reward poisoning attacks and investigates the feasibility and impacts of such attacks. The attacker in [67] aims to force the agent to pull a target arm on a particular context. [26] extends the attack idea of [43, 56] to linear contextual bandits. It proves that the proposed reward poisoning attack strategy can force any bandit algorithms to pull a specific set of arms when the rewards are bounded. It introduces an adaptive reward poisoning attack strategy and observes empirically that the total cost of the adaptive attack is sublinear. In addition, [26] analyzes the context poisoning attacks in white-box setting and shows that LinUCB is vulnerable to such attack.

The action poisoning attack on contextual linear bandit is not a simple extension of the case of MAB or RL. Firstly, in the MAB settings the rewards only depend on the arm (action), while in the contextual bandit setting, the rewards depend on both the arm (action) and the context (state). Secondly, [60] discusses the action poisoning attack in the tabular RL case where the number of states (contexts) is finite. In the linear contextual bandit problem, the number of contexts is infinite.

These factors make the design of attack strategies and performance analysis for the contextual linear bandit problems much more challenging.

**Robust algorithms.** Lots of efforts have been made to design robust bandit algorithms to defend adversarial attacks in the MABs setting [24, 30, 33, 58, 65]. In the linear contextual bandit setting, [10] proposes a stochastic linear bandit algorithm, called Robust Phased Elimination (RPE), that is robust to reward poisoning attacks. It provides two variants of RPE algorithm which separately work on known attack budget case and agnostic attack budget case. [21] provides a robust linear contextual bandit algorithm, called RobustBandit, that works under both the reward poisoning attacks and context poisoning attacks.

**Contributions:** The contributions of this work are: (1) We propose a new online action poisoning attack against contextual bandit in which the attacker aims to force the agent to frequently pull a target arm chosen by the attacker via strategically changing the agent's actions. (2) We introduce a white-box attack strategy that can manipulate any sublinear-regret linear contextual bandit agent into pulling a target arm $T - o(T)$ rounds over a horizon of $T$ rounds, while incurring a cost that is sublinear dependent on $T$. The proposed attack strategy can further be extended to generalized linear contextual bandit models. (3) We design a black-box attack strategy whose performance nearly matches that of the white-box attack strategy. We apply the black-box attack strategy against a very popular and widely used bandit algorithm: LinUCB. We show that our proposed attack scheme can force the LinUCB agent into pulling a target arm $T - O(\log^3 T)$ times with attack cost scaling as $O(\log^3 T)$. (4) We evaluate our attack strategies using both synthetic and real datasets. We observe empirically that the total cost of our black-box attack is sublinear for a variety of contextual bandit algorithms.

The results in this part have been published in [61].

## 1.4 Attacks on Reinforcement Learning

Building on the insights and techniques developed for multi-armed bandit and contextual bandit problems, we then focus on the general RL problems. While there is much existing work addressing adversarial attacks on supervised learning models [3,13,16,17,20,22,29,46,50,75,80,95,101,102, 114], the understanding of adversarial attacks on general RL models is less complete. Among the limited existing works on adversarial attacks against RL, they formally or experimentally considers different types of poisoning attack [8,38,69,83,84,92,116]. [92] discusses the differences between the poisoning attacks. In the observation poisoning attack setting, the attacker is able to manipulate the observations of the agent. Before the agent receives the reward signal or the state signal from the environment, the attacker is able to modify the data. In the environment poisoning setting, the attacker could directly change the underlying environment, i.e., the Markov decision process (MDP) model.

In Chapter 4, we introduce a suite of novel attacks on RL named action poisoning attacks. In the proposed action poisoning attacks models, an attacker sits between the agent and the environment and could change the agent's action. For example, in auto-driving systems, the attacker could implement destabilizing forces or manipulate the action signal, so as to change the brake force. Compared with the observation poisoning or environment poisoning attacks, the ability of the attacker in the action poisoning attack is more restricted, which brings some design challenges. In particular, compared with observation poisoning and environment poisoning attacks, the effects of the action poisoning attack on the change of observation is less direct. Furthermore, when the action space is discrete and finite, the ability of the action poisoning attacker is severely limited. We note that the goal of this work is not to promote action manipulation attacks. Rather our goal is to understand the potential risks of action manipulation attacks, as understanding the risks of different kinds of adversarial attacks on RL is essential for the safe applications of RL model and designing robust RL systems.

In Chapter 4, we investigate action poisoning attacks in both white-box and black-box settings. The white-box attack setting makes strong assumptions. In particular, the attacker has full information of the underlying MDP, the agent's algorithm or the agent's previous policy models, or all of them. While it is often unrealistic to exactly know the underlying environment or have the right to obtain the information of the agent's model, the understanding of the white-box attacks could provide insights on how to design black-box attack schemes. In the black-box setting, the attacker has no prior information of the underlying MDP and does not know the agent's algorithm. The only information the attacker has is observations generated from the environment when the agent interacts with the environment. The black-box setting is much more practical and is suitable for more realistic scenarios.

**Related work:** Existing works on poisoning attacks against RL have studied different types of adversarial manipulations. [69] studies reward poisoning attack against batch RL in which the attacker is able to gather and modify the collected batch data. [83] proposes a white-box environment poisoning model in which the attacker could manipulate the original MDP to a poisoned MDP. [8, 116] study online white-box reward poisoning attacks in which the attacker could manipulate the reward signal before the agent receives it. [92] proposes a practical black-box poisoning algorithm called VA2C-P. Their empirical results show that VA2C-P works for deep policy gradient RL agents without any prior knowledge of the environment. [84] develops a black-box reward poisoning attack strategy called U2, that can provably attack any efficient RL algorithms. There are also some interesting works that focus on attacking multi-arm bandit problems [30, 43, 56, 58, 59] and contextual bandit problems [26, 67]. Existing work on action poisoning attacks against RL is limited. There are some empirical studies in deep RL [48, 81, 92].

**Contributions:** Our main contributions are as follows: (1) We propose an action poisoning attack model in which the attacker aims to force the agent to learn a policy selected by the attacker (will be called target policy in the sequel) by changing the agent's actions to other actions. We use loss and cost functions to evaluate the effects of the action poisoning attack on a RL agent. The cost is the cumulative number of times when the attacker changes the agent's action, and the loss is the cumulative number of times when the agent does not follow the target policy. It is clearly of interest to minimize both the cost and loss functions. (2) In the white-box setting, we introduce an attack strategy named $\alpha$-portion attack. We show that the $\alpha$-portion attack strategy can force any sub-linear-regret RL agent to choose actions according to the target policy specified by the attacker with sub-linear cost and sub-linear loss. (3) We develop a black-box attack strategy, LCB-H, that nearly matches the performance of the white-box $\alpha$-portion attack. To the best of our knowledge, LCB-H is the first black-box action poisoning attack scheme that provably works against RL agents. (4) We investigate the impact of the LCH-B attack on UCB-H [41], a popular and efficient model-free $Q$-learning algorithm, and show that, by spending only logarithm cost, the LCB-H attack can force the UCB-H agent to choose actions according to the target policy with logarithm loss.

The results in this part have been published in [60].

## 1.5 Adversarial Attacks on Multi-Agent RL

Building on the insights and techniques developed for RL problems, we study the adversarial attacks on multi-agent RL (MARL). In MARL, at each state, each agent takes its own action, and these actions jointly determine the next state of the environment and the reward of each agent. The rewards may vary for different agents. For MARL setting, we focus on the model of Markov Games (MG) [86]. In this class of problems, researchers typically consider learning objectives such as Nash equilibrium (NE), correlated equilibrium (CE) and coarse correlated equilibrium (CCE) etc. A recent line of works provide non-asymptotic guarantees for learning NE, CCE or CE under different assumptions [6, 42, 63, 71, 89, 109, 113].

Existing work on adversarial attacks on MARL is limited. In this thrust, we aim to fill in this gap and systematically investigate the impact of adversarial attacks on online MARL. We consider a setting in which there is an attacker sits between the agents and the environment, and can monitor the states, the actions of the agents and the reward signals from the environment. The attacker is able to manipulate the feedback or action of the agents. The objective of the MARL learner is to learn an equilibrium. The attacker's goal is to force the agents to learn a target policy or to maximize the cumulative rewards under some specific reward function chosen by the attacker, while minimizing the amount of the manipulation on feedback and action.

**Related work:** **Attacks on Single Agent RL:** Adversarial attacks on single agent RL have been studied in various settings [8, 38, 69, 83, 84, 92, 116]. For example, [8, 85, 116] study online reward poisoning attacks in which the attacker could manipulate the reward signal before the agent receives it. [60] studies online action poisoning attacks in which the attacker could manipulate the action signal before the environment receives it. [85] studies the limitations of reward only manipulation or action only manipulation in single-agent RL.

**Attacks on MARL:** [68] considers a game redesign problem where the designer knows the full information of the game and can redesign the reward functions. The proposed redesign methods can incentivize players to take a specific target action profile frequently with a small cumulative design cost. [27, 32] study the poisoning attack on multi-agent reinforcement learners, assuming that the attacker controls one of the learners. [107] studies the reward poisoning attack on offline multi-agent reinforcement learners.

**Defense Against Attacks on RL:** There is also recent work on defending against adversarial attacks on RL [7, 14, 66, 104, 106, 115]. These works focus on the single-agent RL setting where an adversary can corrupt the reward and state transition.

**Contributions:** Our contributions are follows. 1) We propose an adversarial attack model in which the attacker aims to force the agent to learn a target policy selected by the attacker or to maximize the cumulative rewards under some specific reward function chosen by the attacker. We use loss and cost functions to evaluate the effectiveness of the adversarial attack on MARL agents. The cost is the cumulative sum of the action manipulations and the reward manipulations. If the attacker aims to force the agents to learn a target policy, the loss is the cumulative number of times when the agent does not follow the target policy. Otherwise, the loss is the regret to the policy that maximizes the attacker's rewards. It is clearly of interest to minimize both the loss and cost. 2) We study the attack problem in three different settings: the white-box, the gray-box and the black-box settings. In the white-box setting, the attacker has full information of the underlying environment. In the gray-box setting, the attacker has no prior information about the underlying environment and the agents' algorithm, but knows the target policy that maximizes its cumulative rewards. In the black-box setting, the target policy is also unknown for the attacker. 3) We show that the effectiveness of action poisoning only attacks and reward poisoning only attacks is limited. Even in the white-box setting, we show that there exist some MGs under which no action poisoning only Markov attack strategy or reward poisoning only Markov attack strategy can be efficient and successful. At the same time, we provide some sufficient conditions under which the action poisoning only attacks or the reward poisoning only attacks can efficiently attack MARL algorithms. Under such conditions, we introduce an efficient action poisoning attack strategy and an efficient reward poisoning attack strategy, and analyze their cost and loss. 4) We introduce a mixed attack strategy in the gray-box setting and an approximate mixed attack strategy in the black-box setting. We show that the mixed attack strategy can force any sub-linear-regret MARL agents to choose actions according to the target policy specified by the attacker with sub-linear cost and sub-linear loss. We further investigate the impact of the approximate mixed attack strategy attack on V-learning [42], a simple, efficient, decentralized algorithm for MARL.

The results in this part have been published in [62].

## 1.6 Action Robust Reinforcement Learning

The solutions to standard RL methods are not inherently robust to uncertainties, perturbations, or structural changes in the environment, which are frequently observed in real-world settings. A trustworthy reinforcement learning algorithm should be competent in solving challenging real-world problems with robustness against perturbations and uncertainties. Robust RL aims to improve the worst-case performance of algorithms deterministically or statistically in the face of uncertainties in different MDP components, including observations/states [93, 112], actions

[45, 97], transitions [39, 77, 96, 103], and rewards [38, 47].

In Chapter 6, we consider action uncertainties, also called policy execution uncertainties, and probabilistic uncertainty set proposed in [97]. Robust RL against action uncertainties focuses on the discrepancy between the actions generated by the RL agent and the conducted actions. Taking the robot control as an example, such policy execution uncertainty may come from the actuator noise, limited power range, or actuator failures in the real world. Taking the medication advice in healthcare as another example, such policy execution uncertainty may come from the patient's personal behaviors such as drug refusal, forgotten medication, or overdose etc.

To deal with the policy execution uncertainty, robust RL methods [81, 97] adopt the adversarial training framework [29, 70] and assume an adversary conducting adversarial attacks to mimic the naturalistic uncertainties. Training with an adversary can be formulated as a zero-sum game between the adversary and the RL agent. However, these interesting works do not provide theoretical guarantee on sample complexity or regret. In Chapter 6, we aim to fill this gap. The approaches in [81, 97] iteratively apply two stages: (i) given a fixed adversary policy, it calculates the agent's optimal policy; and (ii) update the adversary policy against the updated agent's policy. The repetition of stage (i) requires repeatedly solving MDP to find the optimal policy, which is sample inefficient. Motivated by the recent theoretical works on transition probability uncertainty that use the robust dynamic programming method [39] and achieve efficient sample complexity [79, 103, 110], we introduce the action robust Bellman equations and design sample efficient algorithms based on the action robust Bellman equations. Our methods simultaneously update the adversary policy and agent's policy instead of updating one after the other has converged.

**Related work:** We mostly focus on papers that are related to sample complexity bounds for the episodic RL and the two-player zero-sum Markov game, and action robust RL, that are closely related to our model. There are also some related settings, e.g., infinite-horizon discounted MDP [36, 52], robust RL with other uncertainties [39, 47, 103, 112], robust offline RL [31, 88], adversarial training with a generative RL model [79, 110], adversarial attacks on RL [60, 92, 116], etc, whose techniques may be also related to our aciton robust RL work.

**Action robust RL.** [81] introduce robust adversarial RL to address the generalization issues in RL by training with a destabilizing adversary that applies disturbance forces to the system. [97] introduce two new criteria of robustness for RL in the face of action uncertainty. We follow its probabilistic action robust MDP (PR-MDP) in which, instead of the action specified by the policy, an alternative adversarial action is taken with probability $\rho$. They generalize their policy iteration approach to deep reinforcement learning (DRL) and provide extensive experiments. A similar uncertainty setting was presented [45], which extends temporal difference (TD) learning algorithms by a new robust operator and shows that the new algorithms converge to the optimal robust $Q$-function. However, no theoretical guarantee on sample complexity or regret is provided in these works. We develop a minimax sample efficient algorithm and fill this gap.

**Sample complexity bounds for the episodic RL.** There is a rich literature on sample complexity guarantees for episodic tabular RL, for example [5, 18, 19, 40, 41, 44, 90, 91, 117, 118]. However, these methods cannot be directly applied in action robust MDP with small technical changes. Most relevant one is the work about policy certificates [19]. The algorithm ORLC in [19] calculates both the upper bound and lower bound of the value functions, and outputs policy certificates that bound the sub-optimality and return of the policy. Our proposed ARRLC shares a similar structure with ORLC, but we develop new adversarial trajectory sampling and action robust value iteration method in ARRLC, and new techniques to bound the sum of variances so that our algorithm suits for action robust MDPs.

**Sample complexity bounds for the two-player zero-sum Markov game.** Training with an adversary can naturally be formulated as a zero-sum game between the adversary and the RL agent. Some sample efficient algorithms for two-player zero-sum Markov game can be used to train the action robust RL agent. The efficient multi-agent RL algorithms, like [42, 63], can be used to solve the action robust optimal policy but are not minimax optimal. They are a factor of $A$ or $H^2$ above the minimax lower bound. Our algorithm ARRLC is minimax optimal.

**Contributions:** Our major contributions of this work are summarized as follows: (1) We show that the robust problem can be solved by the iteration of the action robust Bellman optimality equations. Motivated by this, we design two efficient algorithms. (2)We develop a model-based algorithm, Action Robust Reinforcement Learning with Certificates (ARRLC), for episodic action robust MDPs, and show that it achieves minimax order optimal regret and minimax order optimal sample complexity. (3) We develop a model-free algorithm for episodic action robust MDPs, and analyze its regret and sample complexity. (4) We conduct numerical experiments to validate the robustness of our approach. In our experiments, our robust algorithm achieves a much higher reward than the non-robust RL algorithm when being tested with some action perturbations; and our ARRLC algorithm converges much faster than the robust TD algorithm in [45].

The results in this part have been submitted for possible publication [57].

# Chapter 2

# Action Attacks on Stochastic Bandits

In this chapter, we focus on stochastic bandit problems. We introduce a new class of attacks named action-manipulation attacks, an efficient attack strategy, and a novel algorithm that is robust to action-manipulation attacks when an upper bound for the total attack cost is given. In Chapter 2.1, we describe the model. In Chapter 2.2, we describe the LCB attack strategy and analyze its accumulative attack cost. In Chapter 2.3, we propose a defense strategy and analyze its regret. In Chapter 2.4, we provide numerical examples to validate the theoretic analysis. Finally, we offer several concluding remarks in Chapter 2.5. The proofs are collected in Appendix A.

## 2.1 Model

In this section, we introduce our model. We consider the standard multi-armed stochastic bandit problems setting. The environment consists of $K$ arms, with each arm corresponds to a fixed but unknown reward distribution. The bandit algorithm, which is also called "user" in this chapter, proceeds in discrete time $t = 1, 2, \dots, T$, in which $T$ is the total number of rounds. At each round $t$, the user pulls an arm (or action) $I_t \in \{1, \dots, K\}$ and receives a random reward $r_t$ drawn from the reward distribution of arm $I_t$. Denote $\mu_i$ as the mean reward of arm $i$. Denote $\tau_i(t) := \{s : s \leq t, I_s = i\}$ as the set of rounds up to $t$ where the user chooses arm $i$, $N_i(t) := |\tau_i(t)|$ as the number of rounds that arm $i$ was pulled by the user up to time $t$ and

$$\hat{\mu}_i(t) := N_i(t)^{-1} \sum_{s \in \tau_i(t)} r_s \tag{2.1}$$

as the empirical mean reward of arm $i$. The pseudo-regret $\bar{R}(T)$ is defined as

$$\bar{R}(T) = T \max_{\max_{i \in [K]}} \mu_i - \mathbb{E}\left[\sum_{t=1}^{T} r_t\right]. \tag{2.2}$$

The goal of the user is to minimize $\bar{R}(T)$.

In this chapter, we introduce a novel adversary setting, in which the attacker sits between the user and the environment. The attacker can monitor the actions of the user and the reward signals from the environment. Furthermore, the attacker can introduce action-manipulation attacks on stochastic bandits. In particular, at each round $t$, after the user chooses an arm $I_t$, the attacker can manipulate the user's action by changing $I_t$ to another $I_t^0 \in \{1, \dots, K\}$. If the attacker decides not to attack, $I_t^0 = I_t$. Then the environment generates a random reward $r_t$ from the reward distribution of post-attack arm $I_t^0$. The user and the attacker receive reward $r_t$ from the environment.

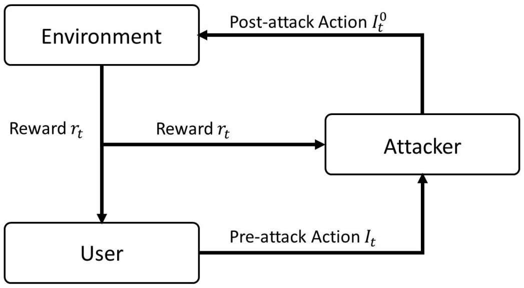


Figure 2.1: Action-manipulation attack model

Without loss of generality and for notation convenience, we assume arm $K$ is the "attack target" arm or the target arm. The attacker's goal is to manipulate the user into pulling the target arm very frequently but by making attacks as rarely as possible. Define the set of rounds when the attacker decides to attack as $\mathcal{C} := \{t : t \leq T, I_t^0 \neq I_t\}$. The cumulative attack cost is the total number of rounds where the attacker decides to attack, i.e., $|\mathcal{C}|$.

In this chapter, we follow the general assumption of the previous works [12, 56, 65] on bandits problem, which consider the short-tail reward distribution environments, e.g. the clicks of article recommendation, and assume that the reward distribution of arm $i$ follows $\sigma^2$-sub-Gaussian distributions with mean $\mu_i$. Denote the true reward vector as $\boldsymbol{\mu} = [\mu_1, \cdots, \mu_K]$. Neither the user nor the attacker knows $\boldsymbol{\mu}$, but $\sigma^2$ is known to both the user and the attacker. We note that the assumption that the attacker does not know $\boldsymbol{\mu}$ is only necessary for Chapter 2.2, in which we design attack strategies. We do not use this assumption in Chapter 2.3 where we design defense strategies. Define the difference of mean value of arm $i$ and $j$ as $\Delta_{i,j} = \mu_i - \mu_j$. Furthermore, we refer to the best arm as $i_O = \arg\max_i \mu_i$ and the worst arm as $i_W = \arg\min_i \mu_i$.

In Chapter 2.2, the assumption that the attacker does not know $\boldsymbol{\mu}$ is important. If the attacker knows these values, the attacker can adopt a trivial oracle attack scheme: whenever the user pulls a non-target arm $I_t$, the attacker changes $I_t$ to the worst arm $i_W$. Assuming that the target arm is not the worst, it is easy to show that, if the user uses a bandit algorithm that has a regret upper bounded of $\mathcal{O}(\log(T))$ when there is no attack, the oracle attack scheme can force the user to pull the target arm $T - \mathcal{O}(\log(T))$ times, using a cumulative cost $|\mathcal{C}| = \mathcal{O}(\log(T))$. However, the oracle attack scheme is not practical when the true reward vector is unknown. In this chapter, we will first design an effective attack scheme, which does not assume the knowledge of true reward vector and nearly matches the performance of the oracle attack scheme, to attack the UCB algorithm. We will then design a new bandit algorithm that is robust against the action-manipulation attack.

The action-manipulation attack considered here is different from reward-manipulation attacks introduced by interesting recent work [43, 56], where the attacker can change the reward signal from the environment. In the setting considered in [43, 56], the attacker can change the reward signal $r_t$ from the environment to an arbitrary value chosen by the attacker. Correspondingly, the cumulative attack cost in [43,56] is defined to be the sum of the absolute value of the changes on the reward. Compared with the reward-manipulation attacks discussed above, the action-manipulation attack is more difficult to carry out. In particular, as the action-manipulation attack only changes the action, it can impact but does not have direct control of the reward signal, which will be a random variable drawn from a distribution depending on the action chosen by the attacker. This is in contrast to reward-manipulation attacks where an attacker can change the reward to any value.

## 2.2 Attack on UCB and Cost Analysis

In this section, we use UCB algorithm as an example to illustrate the effects of action-manipulation attack. We will introduce LCB attack strategy on the UCB bandit algorithm and analyze the cost.

### 2.2.1 Attack strategy

UCB algorithm [12] is one of the most popular bandit algorithm. UCB algorithm keeps optimism in the face of uncertainty and chooses the arm with the highest upper confidence bound of its estimated reward. In UCB algorithm, the user initially pulls each of the $K$ arms once in the first $K$ rounds. After that, the user chooses arms according to

$$I_t = \arg\max_i \left\{ \hat{\mu}_i(t-1) + 3\sigma\sqrt{\frac{\log t}{N_i(t-1)}} \right\}. \tag{2.3}$$

Under the action-manipulation attack, as the user does not know that $r_t$ is generated from arm $I_t^0$ instead of $I_t$, the empirical mean $\hat{\mu}_i(t)$ computed using (2.1) is not a proper estimate of the true mean reward of arm $i$ anymore. On the other hand, the attacker is able to obtain a good estimate of $\mu_i$ by

$$\hat{\mu}_i^0(t) := N_i^0(t)^{-1} \sum_{s \in \tau_i^0(t)} r_s, \tag{2.4}$$

where $\tau_i^0(t) := \{s : s \leq t, I_s^0 = i\}$ is the set of rounds up to $t$ when the attacker changes an arm to arm $i$, and $N_i^0(t) = |\tau_i^0(t)|$ is the number of pulls of post-attack arm $i$ up to round $t$. This information gap provides a chance for attack. In this section, we assume that the target arm is not the worst arm, i.e., $\mu_K > \mu_{i_W}$. We will discuss the case where the target arm is the worst arm in Chapter 2.2.3.

The proposed attack strategy works as follows. In the first $K$ rounds, the attacker does not attack. After that, at round $t$, if the user chooses a non-target arm $I_t$, the attacker changes it to arm $I_t^0$ that has the smallest lower confidence bound (LCB):

$$I_t^0 = \arg\min_i \left\{\hat{\mu}_i^0(t-1) - \mathbf{CB}\left(N_i^0(t-1), \delta\right)\right\}, \tag{2.5}$$

where

$$\mathbf{CB}(N, \delta) = \sqrt{\frac{2\sigma^2}{N} \log \frac{\pi^2 K N^2}{3\delta}}. \tag{2.6}$$

Here $\delta \in (0,1)$ is a parameter that is related to the probability statements in the analytical results presented in Chapter 2.2.2. We call our scheme as LCB attack strategy. Note that the form of (2.6) is slightly different from typical form used in UCB algorithms. We choose to use this form for the simplicity of proofs. If at round $t$ the user chooses the target arm, the attacker does not attack. Thus the cumulative attack cost of our LCB attack scheme is equal to the total of times when the non-target arms are selected by the user. The algorithm is summarized in Algorithm 2.1.

**Algorithm 2.1** LCB attack strategy on UCB algorithm

```
Require:
    The user's bandit algorithm namely UCB algorithm, target arm $K$
for $t = 1, 2, \dots$ do
    The user chooses arm $I_t$ to pull according to UCB algorithm (2.3).
    if $I_t = K$ then
        The attacker does not attack, and $I_t^0 = I_t$.
    else
        The attacker attacks and changes arm $I_t$ to $I_t^0$ chosen according to (2.5).
    end if
    The environment generates reward $r_t$ from arm $I_t^0$.
    The attacker and the user receive $r_t$.
end for
```

Here, we highlight the main idea why LCB attack strategy works. As discussed in Chapter 2.1, if the attacker knows which arm is the worst, the attacker can simply change the action to the worst arm when the user pulls the non-target arm. The main idea of the attack scheme is to estimate the mean of each arm, and change the non-target arm to the arm whose lower confidence bound is the smallest. Effectively, this will almost always change the non-target arm to the worst arm. More formally, for $i \neq K$, we will show that this attack strategy will ensure that $\hat{\mu}_i$ computed using (2.1) by the user converges to $\mu_{i_W}$. On the other hand, as the attacker does not attack when the user selects $K$, $\hat{\mu}_K$ computed by the user will still converge to the true mean $\mu_K$ with $N_K$ increasing. Because the assumption that the target arm is not the worst, which implies that $\mu_K > \mu_{i_W}$, $\hat{\mu}_i$ could be smaller than $\hat{\mu}_K$. Then the user will rarely pull the non-target arms as $\hat{\mu}_i$ is smaller than $\hat{\mu}_K$. Hence, the attack cost would also be small. The rigorous analysis of the cost will be provided in Chapter 2.2.2.

### 2.2.2 Cost analysis

To analyze the cost of the proposed scheme, we need to track $\hat{\mu}_i^0(t)$, the estimate obtained by the attacker using (2.4), and $\hat{\mu}_i(t)$, the estimate obtained by the user using (2.1).

The analysis of $\hat{\mu}_i^0(t)$ is relatively simple, as the attacker knows which arm is truly pulled and hence $\hat{\mu}_i^0(t)$ is the true estimate of the mean of arm $i$. Define event

$$\mathcal{E}_1 := \{\forall i, \forall t > K : |\hat{\mu}_i^0(t) - \mu_i| < \mathbf{CB}(N_i^0(t), \delta)\}. \tag{2.7}$$

In specific, event $\mathcal{E}_1$ is the event that the empirical mean at step $i$ computed by the attacker using (2.4), i.e. $\hat{\mu}_i^0(t)$, is not far from the true mean $\mu_i$ by $\mathbf{CB}(N_i^0(t), \delta)$, for any arm $i$ at any step $t$. The following lemma, proved in [43], shows that the attacker can accurately estimate the average reward to each arm.

**Lemma 1.** (Lemma 1 in [43]) For $\delta \in (0, 1)$, $\mathbb{P}(\mathcal{E}_1) > 1 - \delta$.

The analysis of $\hat{\mu}_i(t)$ computed by the user is more complicated. When the user pulls arm $i$, because of the action-manipulation attacks, the random rewards may be drawn from different reward distributions. Define $\tau_{i,j}(t) := \{s : s \leq t, I_s = i \text{ and } I_s^0 = j\}$ as the set of rounds up to

$t$ when the user chooses arm $i$ and the attacker changes it to arm $j$. We define the empirical mean rewards of a part of arm $i$ whose post-attack arm is $j$ by

$$\hat{\mu}_{i,j}(t) := N_{i,j}(t)^{-1} \sum_{s \in \tau_{i,j}(t)} r_s, \tag{2.8}$$

where $N_{i,j}(t) := |\tau_{i,j}(t)|$.

Define event

$$\mathcal{E}_2 := \left\{ \forall i, \forall j, \forall t > K : |\hat{\mu}_{i,j}(t) - \mu_j| < \mathbf{CB}\left( N_{i,j}(t), \frac{\delta}{K} \right) \right\}. \tag{2.9}$$

In specific, event $\mathcal{E}_2$ is the event that the empirical mean at step $i$ computed by (2.4), i.e. $\hat{\mu}_{i,j}(t)$, is not far from the true mean $\mu_i$ by $\mathbf{CB}(N_{i,j}(t), \delta/K)$, for any arm $i$ and any post-attack arm $j$ at any step $t$.

**Lemma 2.** For $\delta \in (0, 1)$, $\mathbb{P}(\mathcal{E}_2) > 1 - \delta$.

*Proof.* Please refer to Appendix A.1.1. □

Lemma 2 shows a high-probability confidence bounds of the empirical mean rewards of a part of arm $i$ whose post-attack arm is $j$.

Although $r_s$ in (2.1), used to calculate $\hat{\mu}_i(t)$, may be drawn from different reward distributions, we can build a high-probability bound of $\hat{\mu}_i(t)$ with the help of Lemma 2.

**Lemma 3.** Under event $\mathcal{E}_2$, for all arm $i$ and all $t > K$, we have

$$\left| \hat{\mu}_i(t) - \frac{1}{N_i(t)} \sum_{s \in \tau_i(t)} \mu_{I_s^0} \right| < \mathbf{CB}\left( \frac{N_i(t)}{K}, \frac{\delta}{K} \right), \tag{2.10}$$

*Proof.* Please refer to Appendix A.1.2. □

Under events $\mathcal{E}_1$ and $\mathcal{E}_2$, we can build a connection between $\hat{\mu}_i(t)$ and $\mu_{i_W}$. In the proposed LCB attack strategy, the attacker explores and exploits the worst arm by a lower confidence bound method. Thus, when the user pulls a non-target arm, the attacker changes it to the worst arm at most of rounds, which means that for all $i \neq K$, $\hat{\mu}_i(t)$ will converge to $\mu_{i_W}$ as $N_i(t)$ increases. Lemma 4 shows the relationship between $\hat{\mu}_i(t)$ and $\mu_{i_W}$.

**Lemma 4.** Under events $\mathcal{E}_1$ and $\mathcal{E}_2$, using LCB attack strategy 2.1, we have

$$\hat{\mu}_i(t) \leq \mu_{i_W} + \frac{1}{N_i(t)} \sum_{j \neq i_W} \frac{8\sigma^2}{\Delta_{j,i_W}} \log \frac{\pi^2 K t^2}{3\delta} + \sqrt{\frac{2\sigma^2 K}{N_i(t)} \log \frac{\pi^2 (N_i(t))^2}{3\delta}}, \forall i, t. \tag{2.11}$$

*Proof.* Please refer to Appendix A.1.3. □

Lemma 4 shows an upper bound of the empirical mean reward of pre-attack arm $i$, for all arm $i \neq K$. Our main result is the following upper bound on the attack cost $|\mathcal{C}|$.

**Theorem 1.** With probability at least $1 - 2\delta$, when $T \geq \left(\frac{\pi^2 K}{3\delta}\right)^{\frac{2}{5}}$, using LCB attack strategy specified in Algorithm 2.1, the attacker can manipulate the user into pulling the target arm in at least $T - |\mathcal{C}|$ rounds, with an attack cost

$$|\mathcal{C}| \leq \frac{K-1}{4\Delta_{K,i_W}^2} \left( C_1 + \left( C_1^2 + 4\Delta_{K,i_W} \sum_{j \neq i_W} \frac{8\sigma^2}{\Delta_{j,i_W}} \log \frac{\pi^2 K T^2}{3\delta} \right)^{\frac{1}{2}} \right)^2 . \tag{2.12}$$

where $C_1 = 3\sigma\sqrt{\log T} + \sqrt{2\sigma^2 K \log \frac{\pi^2 T^2}{3\delta}}$.

*Proof.* Please refer to Appendix A.1.4. □

The expression of the cost bound in Theorem 1 is complicated. The following corollary provides a simpler bound that is more explicit and interpretable.

**Corollary 1.** Under the same assumptions in Theorem 1, the total attack cost $|\mathcal{C}|$ of Algorithm 2.1 is upper bounded by

$$\mathcal{O}\left( \frac{K\sigma^2 \log T}{\Delta_{K,i_W}^2} \left( K + \sum_{j \neq i_W} \frac{\Delta_{K,i_W}}{\Delta_{j,i_W}} + \sqrt{K \sum_{j \neq i_W} \frac{\Delta_{K,i_W}}{\Delta_{j,i_W}}} \right) \right), \tag{2.13}$$

and the total number of target arm pulls is $T - |\mathcal{C}|$.

From Corollary 1, we can see that the attack cost scales as $\log T$. Two important constants $\frac{\sigma}{\Delta_{K,i_W}}$ and $\sum_{j \neq i_W} \frac{\Delta_{K,i_W}}{\Delta_{j,i_W}}$ have impact on the prelog factor. In Chapter 2.4, we provide some numerical examples to illustrate the effects of these two constants on the attack cost.

In the above analysis, the attacker has only one target arm and aims to force the user to pull it. We can extend our algorithm to the scenario where there is a set of target arms and the attacker aims to manipulate the user into pulling any one of them very frequently. For this case, we need an assumption that the worst arm is not in the target set. When the user pulls a target arm, the adversary does not attack. When the user pulls a non-target arm, the LCB attack strategy can change it to the worst arm at most of rounds. In this way, the estimate of any non-target arm could be smaller than the estimate of any target arm. As a result, the user will rarely pull the non-target arms and pull arms in the target set very frequently. The attack cost also scales as $\log(T)$.

### 2.2.3 Attacks fail when the target arm is the worst arm

One weakness of our LCB attack strategy is that the attack target arm is necessarily a non-worst arm. In the LCB attack strategy, the attacker cannot force the user to pull the worst arm very frequently by spending only logarithmic cost. The main reason is that, when the target arm is the worst, the average reward of each arm is larger or equal to that of the target arm. As the result, our attack scheme is not able to ensure that the target arm has a higher expected reward than the user's estimate of the rewards of other arms. In fact, the following theorem shows that all action-manipulation attack cannot manipulate the UCB algorithm into pulling the worst arm more than $T - \mathcal{O}(\log(T))$ by spending only logarithmic cost.

**Theorem 2.** *Let $\delta < \frac{1}{2}$. Suppose the attack cost is limited by $\mathcal{O}(\log(T))$. Then no attack can force the UCB algorithm to pick the worst arm more than $T - \mathcal{O}(T^\alpha)$ times with probability at least $1 - \delta$, in which $\alpha < 1$.*

*Proof.* Please refer to Appendix A.1.5. □

This theorem shows a contrast between the case where the target arm is not the worst arm and the case where the target arm is the worst arm. If the target arm is not the worst arm, our scheme is able to force the user to pick the target arm $T - \mathcal{O}(\log(T))$ times with only logarithmic cost. On the other hand, if the target arm is the worst, Theorem 2 shows that there is no attack strategy that can force the user to pick the worst arm more than $T - \mathcal{O}(T^{\alpha})$ times while incurring only logarithmic cost.

In the proof of Theorem 2, we do not use the assumption on whether the attacker knows the true underlying mean vector or not. Hence this theorem is also valid even when the attacker knows the true underlying mean vector and can carry out an oracle attack. To further illustrate the challenges arise for the case where the target arm is the worst arm, we now study the oracle attack for this case. Even though the attacker knows the true underlying mean vector, it is difficult for him to carry out the attack. The main reason is that, since the target arm is the worst arm, in order to make this arm appears to be better to the user, the attacker now needs to attack even when the user pulls the target arm, i.e., to change it to the best arm. Hence the attack has two parts: 1) when the user pulls a non-target arm, the attacker changes the arm to the worst arm; 2) when the user pulls the target arm, the attacker changes the arm to the best arm sometimes. We set the number of rounds that the attacker change the target arm to the best arm as $C_K$. So the attack cost has two parts: the number of rounds where the user pulls a non-target arm and $C_K$. The following proposition analyze the cost of this oracle attack.

**Proposition 1.** With probability at least $1-\delta$, when $T > \left(\frac{\pi^2 K^2}{12\delta}\right)^4$, given the number of rounds that the attacker change the target arm to the best arm as $C_K$, the oracle attack can manipulate the user into pulling the target arm that is the worst arm in at most

$$T - \min\left(\frac{\frac{1}{4}(K-1)\sigma^2 T^2 \log\frac{T}{K}}{\left(KC_K\Delta_{i_O,K} + 6\sigma\sqrt{KT\log\frac{T}{K}}\right)^2}, \frac{T(K-1)}{K}\right) \tag{2.14}$$

rounds, with an attack cost $|\mathcal{C}|$ at least

$$C_K + \min\left(\frac{\frac{1}{4}(K-1)\sigma^2T^2\log\frac{T}{K}}{\left(KC_K\Delta_{i_O,K} + 6\sigma\sqrt{KT\log\frac{T}{K}}\right)^2}, \frac{T(K-1)}{K}\right). \tag{2.15}$$

*Proof.* Please refer to Appendix A.1.6. □

Compared with the performance of LCB attacks for the cases when the target arm is the worst arm, the oracle attack for the case when the target arm is the worst arm requires significantly more attack cost to achieve the similar performance. According to Proposition 1, in order to manipulate the user into pulling the target arm in $T - O(\log T)$ rounds, the $C_K$ should scale as $T$. The attack is extremely ineffective, as now the attack cost scales with $T$. Furthermore, from (2.15), to minimize the cost, we need to set

$$C_K = \frac{1}{K\Delta_{i_O,K}}\left(\left(\frac{1}{2}K(K-1)\Delta_{i_O,K}\sigma^2T^2\log\frac{T}{K}\right)^{\frac{1}{3}} - 6\sigma\sqrt{KT\log\frac{T}{K}}\right) \tag{2.16}$$

which scales as $\Omega\left(T^{\frac{2}{3}}(\log T)^{\frac{2}{3}}\right)$. Hence, for the case where the target arm is the worst arm, the minimal attack cost of the oracle attack is large. There is no effective attacks when the target arm is the worst arm.

## 2.3 Robust Algorithm and Regret Analysis

The results in Chapter 2.2 expose a significant security threat of the action-manipulation attacks on MABs. Under only $\mathcal{O}(\log(T))$ times of attacks carried out using the proposed LCB strategy, the UCB algorithm will almost always pull the target arm selected by the attacker. Although there are some defense algorithms [65] and universal best arm identification schemes [87] for stochastic or adversarial bandit, they do not apply to the action-manipulation attack setting. This motivates us to design a new bandit algorithm that is robust against action-manipulation attacks. In this section, we propose such a robust bandit algorithm and analyze its regret.

### 2.3.1 Robust bandit algorithm

In this section, we assume that a valid upper bound $A$ for the cumulative attack cost $|\mathcal{C}|$ is known for the user, although the user does not have to know the exact cumulative attack cost $|\mathcal{C}|$. $A$ does not need to be constant, it can scale with $T$. In other words, for a given $A$, our proposed algorithm is robust to all action-manipulation attacks with a cumulative attack cost $|\mathcal{C}| < A$. This assumption is reasonable, as if the cost is unbounded, it will not be possible to design a robust scheme.

We first introduce some notation. Denote $\mathbf{N}(t-1) := (N_1(t-1), \ldots, N_K(t-1))$ as the vector counting how many times each action has been taken by the user, and $\hat{\boldsymbol{\mu}}(t-1) = (\hat{\mu}_1(t-1), \ldots, \hat{\mu}_K(t-1))$ as the vector of the sample means computed by the user. The proposed algorithm is a modified UCB method by taking the maximum possible mean estimate offset due to attack into consideration. We name our scheme as maximum offset UCB (MOUCB).

The proposed MOUCB works as follows. In the first $2AK$ rounds, MOUCB algorithm pulls each arm $2A$ times. After that, at round $t$, the user chooses an arm $I_t$ by a modified UCB method:

$$I_t = \arg\max_a \left\{ \hat{\mu}_a(t-1) + \beta(N_a(t-1)) + \gamma(\hat{\boldsymbol{\mu}}(t-1), \mathbf{N}(t-1)) \right\}, \tag{2.17}$$

where

$$\gamma(\hat{\boldsymbol{\mu}}(t-1), \mathbf{N}(t-1)) = \frac{2A}{N_a(t-1)} \times \\ \max_{i,j} \left\{ \hat{\mu}_i(t-1) - \hat{\mu}_j(t-1) + \beta(N_i(t-1)) + \beta(N_j(t-1)) \right\},$$

and

$$\beta(N) = \textbf{CB}\left(\frac{N}{K}, \frac{\delta}{K}\right) = \sqrt{\frac{2\sigma^2 K}{N} \log \frac{\pi^2 N^2}{3\delta}}. \tag{2.18}$$

The algorithm is summarized in Algorithm 2.2.

Compared with the original UCB algorithm in (2.3), the main difference is the additional term

**Algorithm 2.2** Proposed MOUCB bandit algorithm

```
Require:
    A valid upper bound $A$ for the cumulative attack cost.
for $t = 1, 2, \dots$ do
    if $t \leq 2AK$ then
        The user pulls the arm whose pull count is the smallest, i.e. $I_t = \arg\min_i N_i(t-1)$.
    else
        The user chooses arm $I_t$ to pull according (2.17).
    end if
    if The attacker decides to attack then
        The attacker attacks and changes $I_t$ to $I_t^0$.
    else
        The attacker does not attack and $I_t^0 = I_t$.
    end if
    The environment generates reward $r_t$ from arm $I_t^0$.
    The attacker and the user receive $r_t$.
end for
```

$\gamma(\hat{\boldsymbol{\mu}}(t-1), \mathbf{N}(t-1))$ in (2.17). We now highlight the main idea why our bandit algorithm works and the role of this additional term. In particular, in the standard multi-armed stochastic bandit problem, $\hat{\mu}_i(t)$ is a proper estimation of $\mu_i$, the true mean reward of arm $i$. However, under the action-manipulation attacks, as the user does not know which arm is used to generate $r_t$, $\hat{\mu}_i(t)$ is not a proper estimate of the true mean reward anymore. However, we can try to find a good bound of the true mean reward. If we know $\Delta_{i_O, i_W}$, the reward difference between the optimal arm and the worst arm, we can describe the maximum offset of the mean rewards caused by the attack. In particular, we have

$$\mu_i - \frac{A}{N_i(t)} \Delta_{i_O, i_W} \leq \frac{1}{N_i(t)} \sum_{s \in \tau_i(t)} \mu_{I_s^0} \leq \mu_i + \frac{A}{N_i(t)} \Delta_{i_O, i_W}, \tag{2.19}$$

which implies

$$\mu_i \leq \frac{A}{N_i(t)} \Delta_{i_O, i_W} + \frac{1}{N_i(t)} \sum_{s \in \tau_i(t)} \mu_{I_s^0}. \tag{2.20}$$

In (2.20), the first term in the right hand side is the maximum offset that an attacker can introduce regardless of the attack strategy. The second term in the right hand side is related to the mean estimated by the user. In particular, under event $\mathcal{E}_2$, as shown in Lemma 3, we have

$$\frac{1}{N_i(t)} \sum_{s \in \tau_i(t)} \mu_{I_s^0} < \hat{\mu}_i(t) + \beta(N_i(t)). \tag{2.21}$$

Hence, regardless the attack strategy, we have a upper confidence bound on $\mu_i$:

$$\mu_i \leq \hat{\mu}_i(t) + \frac{A}{N_i(t)} \Delta_{i_O, i_W} + \beta(N_i(t)). \tag{2.22}$$

In our case, however, $\Delta_{i_O, i_W}$ is also unknown. In our algorithm, we obtain a high-probability bound on $\Delta_{i_O, i_W}$:

$$\Delta_{i_O, i_W} \leq 2 \max_{i,j} \left\{ \hat{\mu}_i - \hat{\mu}_j + \beta\left(N_i(t)\right) + \beta\left(N_j(t)\right) \right\}, \tag{2.23}$$

which will proved in Lemma 5 below. Now, the second term of (2.22) becomes $\gamma(\hat{\boldsymbol{\mu}}(t-1), \mathbf{N}(t-1))$ if we replace $\Delta_{i_O, i_W}$ with the bound (2.23), and we obtain our final algorithm.

The design of robust algorithms under the adversarial setup can be alternatively viewed as a MABs problem with limited number of mean changes. When the user pulls a single arm, the rewards he receives are drawn from different reward distributions with different means. The means are varying with time because of the manipulation of the attacker. The means change between only $K$ fixed values. In our setting, if the attacker does not decide to attack, the arm chosen by the user does not change and the mean does not change. In this sense, the attack cost is the number of rounds when the mean is different from the initial value. In most rounds, each arm corresponds to a fixed but unknown reward distribution. However, in at most $A$ rounds, the mean of each arm is varying between $K - 1$ fixed values.

### 2.3.2 Regret analysis

Lemma 5 shows a bound of $\Delta_{i_O, i_W}$, the maximum reward difference between any two arms, under event $\mathcal{E}_2$.

**Lemma 5.** For $\delta \leq \frac{1}{3}$, $t > 2AK$ and under event $\mathcal{E}_2$, MOUCB algorithm have

$$\begin{aligned}\Delta_{i_O,i_W} &\leq 2\max_{i,j}\left\{\hat{\mu}_i - \hat{\mu}_j + \beta\left(N_i(t)\right) + \beta\left(N_j(t)\right)\right\} \\ &\leq 2\Delta_{i_O,i_W} + 8\sqrt{\frac{\sigma^2 K}{A}\log\frac{4\pi^2 A^2}{3\delta}}.\end{aligned} \tag{2.24}$$

*Proof.* Please refer to Appendix A.2.1. □

Using Lemma 5, we now bound the regret of Algorithm 2.2.

**Theorem 3.** Let $A$ be an upper bound on the total attack cost $|\mathcal{C}|$. For $\delta \leq \frac{1}{3}$ and $T \geq 2AK$, MOUCB algorithm has pseudo-regret $\bar{R}(T)$

$$\bar{R}(T) \leq \sum_{a \neq i_O} \max\left\{\frac{8\sigma^2 K}{\Delta_{i_O,a}}\log\frac{\pi^2 T^2}{3\delta}, A\left(\Delta_{i_O,a} + 2\Delta_{i_O,i_W} + 8\sqrt{\frac{\sigma^2 K}{A}\log\frac{4\pi^2 A^2}{3\delta}}\right)\right\}, \tag{2.25}$$

with probability at least $1 - \delta$.

*Proof.* Please refer to Appendix A.2.2. □

Theorem 3 reveals that our bandit algorithm is robust to the action-manipulation attacks. If the total attack cost is bounded by $\mathcal{O}(\log T)$, the pseudo-regret of MOUCB bandit algorithm is still bounded by $\mathcal{O}(\log T)$. This is in contrast with UCB, for which we have shown that the pseudo-regret is $\mathcal{O}(T)$ with attack cost $\mathcal{O}(\log T)$ in Chapter 2.2. If the total attack cost is up to $\Omega(\log T)$, the pseudo-regret of MOUCB bandit algorithm is bounded by $\mathcal{O}(A)$, which is linear in $A$. Note that in the design of defense strategy, we do not assume what the attack strategy is. MOUCB can defend against both LCB attacks and oracle attacks. In fact, MOUCB is robust to all action-manipulation attacks, as long as the total attack cost is smaller than $A$. In a sense, $A$ can be viewed as a parameter chosen by the user to strike a balance between performance and robustness against attacks: the larger the value $A$ is, the larger class of attacks the user can defend against, but with the cost of a larger regret.

## 2.4 Numerical Results

In this section, we provide numerical examples to illustrate the analytical results obtained. In our simulation, the bandit has 10 arms. The rewards distribution of arm $i$ is $\mathcal{N}(\mu_i, \sigma)$. The attacker's target arm is $K$. We let $\delta = 0.05$. We then run the experiment for 20 trials and in each trial we run $T = 10^7$ rounds.

### 2.4.1 LCB attack strategy

We first illustrate the impact of the proposed LCB attack strategy on UCB algorithm.

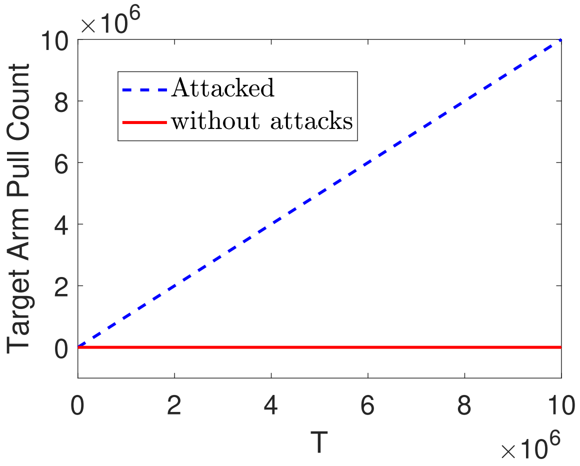


Figure 2.2: Number of rounds the target arm was pulled

In Figure 2.2, we fix $\sigma = 0.1$ and $\Delta_{K,i_W} = 0.1$ and compare the number of rounds at which the target arm is pulled with and without attack. In this experiment, the mean rewards of all arms are 1.0, 0.9, 0.8, 0.7, 0.6, 0.5, 0.4, 0.3, 0.1, and 0.2 respectively. Arm $K$ is not the worst arm, but its average reward is lower than most arms. The results are averaged over 20 trials. The attacker successfully manipulates the user into pulling the target arm very frequently.

In Figure 2.3, in order to study how $\frac{\sigma}{\Delta_{K,i_W}}$ affects the attack cost, we fix $\Delta_{K,i_W} = 0.1$ and set $\sigma$ as 0.1, 0.3 and 0.5 respectively. The mean rewards of all arms are the same as above. From the figure, we can see that as $\frac{\sigma}{\Delta_{K,i_W}}$ increases, the attack cost increases. In addition, as predicted in our analysis, the attack cost increases with $T$, the total number of rounds, in a logarithmic order.

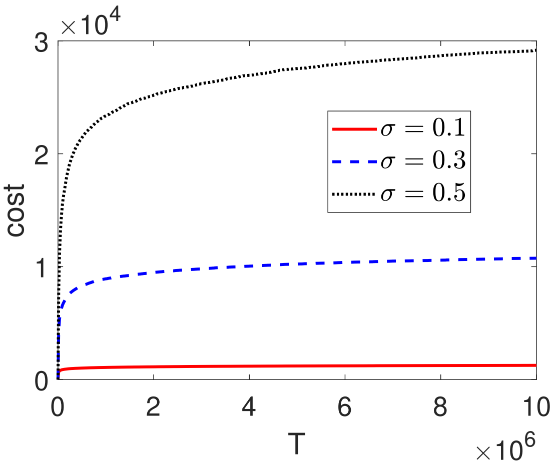


Figure 2.3: Attack cost vs $\frac{\sigma}{\Delta_{K,i_W}}$

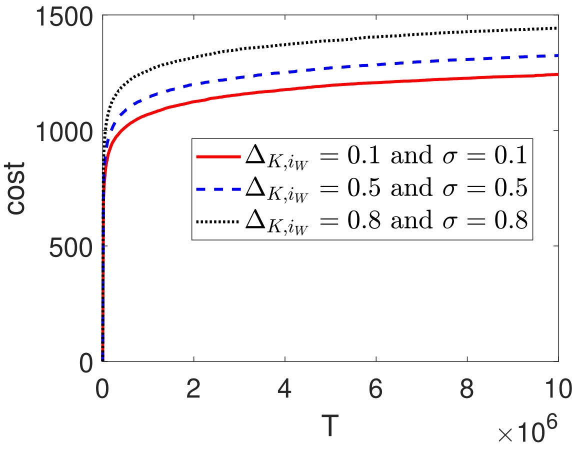


Figure 2.4: Attack cost vs $\sum_{j \neq i_W} \frac{\Delta_{K,i_W}}{\Delta_{j,i_W}}$

Figure 2.4 illustrates how $\sum_{j \neq i_W} \frac{\Delta_{K,i_W}}{\Delta_{j,i_W}}$ affects the attack cost. In this experiment, we fix $\frac{\sigma}{\Delta_{K,i_W}} = 1$ and set $\Delta_{K,i_W}$ as 0.2, 0.6 and 0.9 respectively. The mean rewards of all arms are the same as above. The figure illustrates that, as $\sum_{j \neq i_W} \frac{\Delta_{K,i_W}}{\Delta_{j,i_W}}$ increases, the attack cost also increases. This is consistent with our analysis in Corollary 1.

### 2.4.2 MOUCB bandit algorithm

We now illustrate the effectiveness of MOUCB bandit algorithm.

In this experiment, we use the similar setting as in the simulation of the LCB attack scheme. The mean rewards of all arms are set to be 1.0, 0.8, 0.9, 0.5, 0.2, 0.3, 0.1, 0.4, 0.7, and 0.6 respectively. The total attack cost $|\mathcal{C}|$ is limited by 2000. A given valid upper bound for total attack cost is $A = 3000$. The results are averaged over 20 trials.

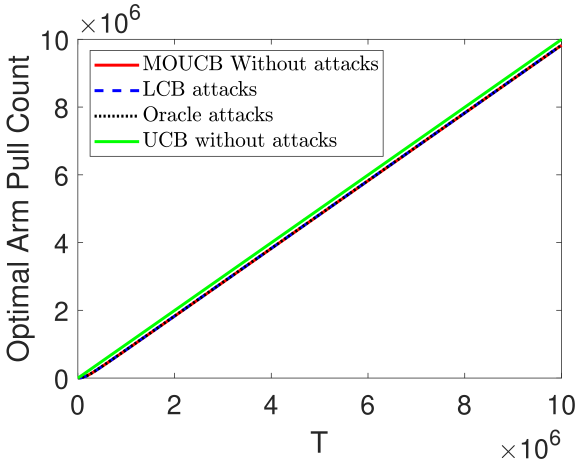


Figure 2.5: Comparison of number of rounds the optimal arm was pulled

In Figure 2.5, we simulate MOUCB algorithm with two different attacks, and compare the numbers of rounds when the optimal arm is pulled under these attacks. The first attack is the LCB attack discussed in Chapter 2.2. The second attack is the oracle attack, in which the attacker knows the true mean reward of arms and implements the oracle attacks that change any non-target arm to a worst arm (see the discussion in Chapter 2.1). For comparison purposes, we also add the curve for MOUCB under no attack, and the curve for UCB under no attack. The results show that, even under the oracle attack, the proposed MOUCB bandit algorithm achieves almost the same performance as the UCB without attack.

To further compare the performance of UCB and MOUCB, in Figure 2.6, we illustrate the performance of UCB algorithm for the three scenarios discussed above: under LCB attack, under oracle attack and under no attack. The results show that both LCB and oracle attacks can successfully manipulates the UCB algorithm into pulling a non-optimal arm very frequently, as the curves for the LCB attack and oracle attack are far away from the curve for no attack. This is in sharp contrast with the situation for MOUCB algorithm shown in Figure 2.5, where the all curves are almost identical.

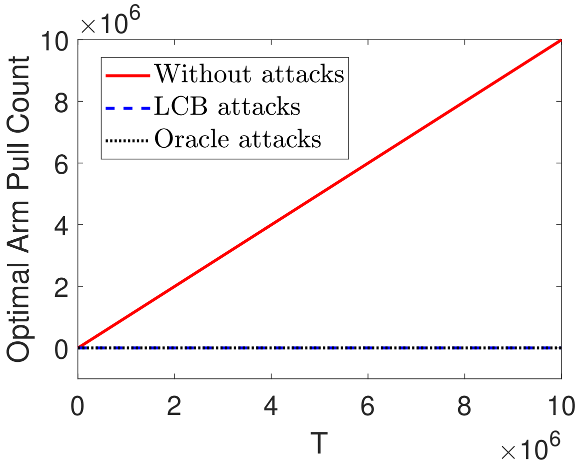


Figure 2.6: Number of rounds the optimal arm was pulled using UCB algorithm

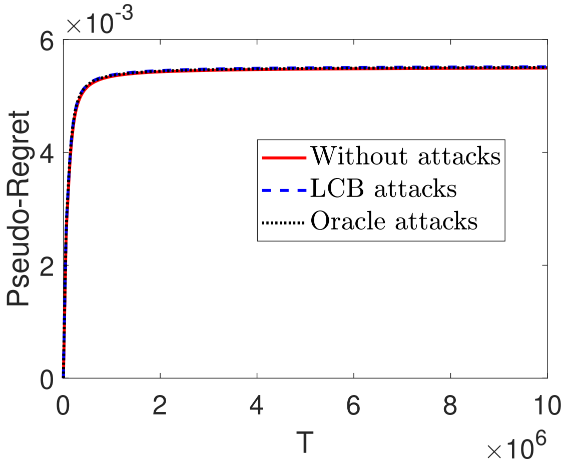


Figure 2.7: Pseudo-regret of MOUCB algorithm

Figure 2.7 and Figure 2.8 illustrate the pseudo-regret of MOUCB bandit algorithm and UCB bandit algorithm respectively. In Figure 2.7, as predicted in our analysis, MOUCB algorithm archives logarithmic pseudo-regrets under both LCB attacks and the oracle attacks. Furthermore, the curves under both attacks are very close to that of the case without attacks. However, as shown in Figure 2.8, the pseudo-regret of UCB grows linearly under both attacks, while grows logarithmically under no attack. The figures again show that UCB is vulnerable to action-manipulation attacks while the proposed MOUCB is robust to the attacks (even for oracle attacks).

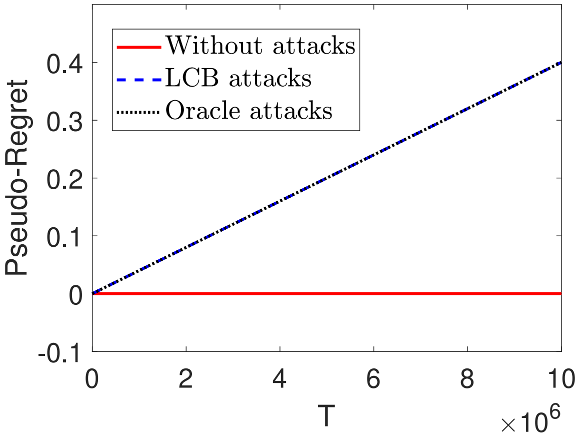


Figure 2.8: Pseudo-regret of UCB algorithm

## 2.5 Conclusion

In this chapter, we have introduced a new class of attacks on stochastic bandits: action-manipulation attacks. We have analyzed the attack against the UCB algorithm and proved that the proposed LCB attack scheme can force the user to almost always pull a non-worst arm with only logarithm effort. To defend against this type of attacks, we have further designed a new bandit algorithm MOUCB that is robust to action-manipulation attacks. We have analyzed the regret of MOUCB under any attack with bounded cost, and have showed that the proposed algorithm is robust to the action-manipulation attacks.

# Chapter 3

# Action Attacks on Contextual Bandits

In this chapter, we introduce action poisoning attacks against linear contextual bandits and some efficient attack strategies. We also extend the proposed attack strategies to generalized linear mode. In Section 3.1, we describe the model. In Section 3.2, we describe the attack strategies and analyze their accumulative attack cost. In Section 3.3, we extend the proposed attack strategies to generalized linear model. In Section 3.4, we provide numerical examples to validate the theoretic analysis. The proofs are collected in Appendix B.

## 3.1 Problem Setup

Consider the standard contextual linear bandit model in which the environment consists of $K$ arms. In each round $t = 1, 2, 3, \dots, T$, the agent observes a context $x_t \in \mathcal{D}$ where $\mathcal{D} \subset \mathbb{R}^d$, pulls an arm $I_t$ and receives a reward $r_{t,I_t}$. Each arm $i$ is associated with an unknown but fixed coefficient vector $\theta_i \in \Theta$ where $\Theta \subset \mathbb{R}^d$. In each round $t$, the reward satisfies $r_{t,I_t} = \langle x_t, \theta_{I_t} \rangle + \eta_t$, where $\eta_t$ is a conditionally independent zero-mean $R$-subgaussian noise and $\langle \cdot, \cdot \rangle$ denotes the inner product. Hence, the expected reward of arm $i$ under context $x_t$ follows the linear setting $\mathbb{E}[r_{t,i}] = \langle x_t, \theta_i \rangle$ for all $t$ and all arm $i$. If we consider the $\sigma$-algebra $F_t = \sigma(x_1, \dots, x_{t+1}, \eta_1, \dots, \eta_t)$, $x_t$ becomes $F_{t-1}$ measurable and $\eta_t$ becomes $F_t$ measurable.

In this chapter, we assume that there exist $L > 0$ and $S > 0$, such that for all round $t$ and arm

$i$, $\|x_t\|_2 \leq L$ and $\|\theta_i\|_2 \leq S$, where $\|\cdot\|_2$ denotes the $\ell_2$-norm. We assume that there exist $\mathcal{D} \subset \mathbb{R}^d$ such that for all $t$, $x_t \in \mathcal{D}$ and, for all $x \in \mathcal{D}$ and all arm $i$, $\langle x, \theta_i \rangle > 0$.

The agent is interested in minimizing the cumulative pseudo-regret

$$\bar{R}_T = \sum_{t=1}^{T} \left( \langle x_t, \theta_{I_t^*} \rangle - \langle x_t, \theta_{I_t} \rangle \right), \tag{3.1}$$

where $I_t^* = \arg\max_i \langle x_t, \theta_i \rangle$.

In this chapter, we introduce a novel adversary setting, in which the attacker can manipulate the action chosen by the agent. In particular, at each round $t$, after the agent chooses an arm $I_t$, the attacker can manipulate the agent's action by changing $I_t$ to another $I_t^0 \in \{1, \ldots, K\}$. If the attacker decides not to attack, $I_t^0 = I_t$. The environment generates a random reward $r_{t,I_t^0}$ based on the post-attack arm $I_t^0$ and the context $x_t$. Then the agent and the attacker receive reward $r_{t,I_t^0}$ from the environment. Since the agent does not know the attacker's manipulations and the presence of the attacker, the agent will still view $r_{t,I_t^0}$ as the reward corresponding to the arm $I_t$.

The goal of the attacker is to design attack strategy to manipulate the agent into pulling a target arm very frequently but by making attacks as rarely as possible. Without loss of generality, we assume arm $K$ is the "attack target" arm or target arm. Define the set of rounds when the attacker decides to attack as $\mathcal{C} := \{t : t \leq T, I_t^0 \neq I_t\}$. The cumulative attack cost is the total number of rounds where the attacker decides to attack, i.e., $|\mathcal{C}|$. The attacker can monitor the contexts, the actions of the agent and the reward signals from the environment.

We now compare the three types of poisoning attacks against contextual linear bandit: reward poisoning attack, action poisoning attack and context poisoning attack. In the reward poisoning attack [26, 67], after the agent observes context $x_t$ and chooses arm $I_t$, the environment will generate reward $r_{t,I_t}$ based on context $x_t$ and arm $I_t$. Then, the attacker can change the reward $r_{t,I_t}$ to $\widetilde{r}_t$ and feed $\widetilde{r}_t$ to the agent. Compared with the reward poisoning attacks, the action poisoning attack considered in this chapter is more difficult to carry out. In particular, as the action poisoning attack only changes the action, it can impact but does not have direct control of the reward signal.

By changing the action $I_t$ to $I_t^0$, the reward received by the agent is changed from $r_{t,I_t}$ to $r_{t,I_t^0}$ which is a random variable drawn from a distribution based on the action $I_t^0$ and context $x_t$. This is in contrast to reward poisoning attacks where an attacker has direct control and can change the reward signal to any value $\widetilde{r}_t$ of his choice. In the context poisoning attack [26], the attacker only changes the context shown to the agent. The reward is also generated based on the true context $x_t$ and the agent's action $I_t$. Nevertheless, the agent's action may be indirectly impacted by the manipulation of the context, and so as the reward. Since the attacker attacks before the agent pulls an arm, the context poisoning attack is the most difficult to carry out. As mentioned in the introduction, the goal of this chapter is not to promote any particular types of poisoning attacks. Instead, our goal is to understand the potential risks of action poisoning attacks, as the safe applications and design of robust contextual bandit algorithm relies on the addressing all possible weakness of the models.

As the action poisoning attack only changes the actions, it can impact but does not have direct control of the agent's observations. Furthermore, when the action space is discrete and finite, the ability of the action poisoning attacker is severely limited. It is reasonable to limit the choice of the target policy. Here we introduce an important assumption that the target arm is not the worst arm:

**Assumption 1.** For all $x \in \mathcal{D}$, the mean reward of the target arm satisfies $\langle x, \theta_K \rangle > \min_{i \in [K]} \langle x, \theta_i \rangle$.

If the target arm is the worst arm in most contexts, the attacker should change the target arm to a better arm or the optimal arm so that the agent learns that the target set is optimal for almost every context. In this case, the cost of attack may be up to $O(T)$. Assumption 1 does not imply that the target arm is optimal at some contexts. The target arm could be sub-optimal for all contexts. Fig. 3.1 shows an example of one dimension linear contextual bandit model, where the $x$-axis represents the contexts and the $y$-axis represents the mean rewards of arms under different contexts. As shown in Fig. 3.1, arms 3 and 4 satisfy Assumption 1. In addition, arm 3 is not optimal at any context.

Under Assumption 1, there exists $\alpha \in (0, \frac{1}{2})$ such that $\max_{x \in \mathcal{D}} \frac{\min_i \langle x, \theta_i \rangle}{\langle x, \theta_K \rangle} \leq (1 - 2\alpha)$.

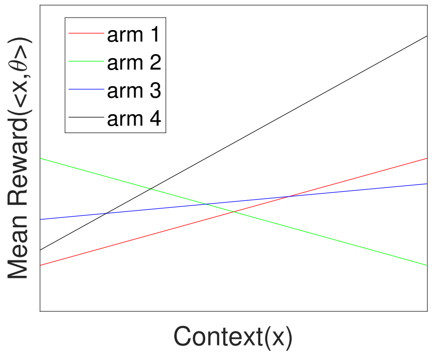


Figure 3.1: An example of one dimension linear contextual bandit model.

Equivalently, Assumption 1 implies that there exists $\alpha \in (0, \frac{1}{2})$, such that for all $t$, we have

$$(1-2\alpha)\langle x_t, \theta_K \rangle \geq \min_{i \in [K]} \langle x_t, \theta_i \rangle. \tag{3.2}$$

Assumption 1 is necessary in our analysis to prove a formal bound of the attack cost. In practice, the proposed algorithms in Section 3.2.2 and 3.2.3 may still work if the target arm is the worst in a small portion of the contexts (as illustrated in the numerical example section).

## 3.2 Attack Schemes and Cost Analysis

In this section, we introduce action poisoning attack schemes in the white-box setting and black-box setting respectively. In order to demonstrate the significant security threat of action poisoning attacks to linear contextual bandits, we investigate our action poisoning attack strategy against a widely used algorithm: LinUCB algorithm. Furthermore, we analyze the attack cost of our action poisoning attack schemes.

### 3.2.1 Overview of LinUCB

For reader's convenience, we first provide a brief overview of the LinUCB algorithm [53]. The LinUCB algorithm is summarized in Algorithm 3.1. The main steps of LinUCB are to obtain estimates of the unknown parameters $\theta_i$ using past observations and then make decisions based on these estimates. Define $\tau_i(t) := \{s : s \leq t, I_s = i\}$ as the set of rounds up to $t$ where the agent pulls arm $i$. Let $N_i(t) = |\tau_i(t)|$. Then, at round $t$, the $\ell_2$-regularized least-squares estimate of $\theta_i$ with regularization parameter $\lambda > 0$ is obtained by [53]

$$\hat{\theta}_{t,i} = V_{t,i}^{-1} \left( \sum_{k \in \tau_i(t-1)} r_{t,i} x_k \right), \tag{3.3}$$

where $V_{t,i} = \sum_{k \in \tau_i(t-1)} x_k x_k^T + \lambda \mathbf{I}$ with $\mathbf{I}$ being identity matrix.

After $\hat{\theta}_i$'s are obtained, at each round, an upper confidence bound of the mean reward has to be calculated for each arm (step 5 of Algorithm 3.1). Then, the LinUCB algorithm picks the arm with the largest upper confidence bound (step 7 of Algorithm 3.1). By following the setup in ”optimism in the face of uncertainty linear algorithm” (OFUL) [1], we set

$$\beta_{t,i} = \sqrt{\lambda} S + R \sqrt{2 \log K/\delta + d \log\left(1 + L^2 N_i(t)/(\lambda d)\right)}.$$

We define $\omega(N) = \sqrt{\lambda} S + R \sqrt{2 \log K/\delta + d \log\left(1 + L^2 N/(\lambda d)\right)}$. It is easy to verify that $\omega(N)$ is a monotonically increasing function over $N \in (0, +\infty)$.

**Algorithm 3.1** Contextual LinUCB [53]

```
Require:
    regularization $\lambda$, number of arms $K$, number of rounds $T$, bound on context norms $L$, bound on parameter norms $S$.
Initialize for every arm $i$, $V_i \leftarrow \lambda \mathbf{I}$, $b_i \leftarrow \mathbf{0}$, $\hat{\theta}_i \leftarrow V_i^{-1} b_i$.
for $t = 1, 2, \ldots, T$ do
    observe the context $x_t$.
    for $i = 1, 2, \ldots, K$ do
        Compute the upper confidence bound: $p_{t,i} \leftarrow x_t^T \hat{\theta}_i + \beta_{t,i} \sqrt{x_t^T V_i^{-1} x_t}$.
    end for
    Pull arm $I_t = \arg\max_i p_{t,i}$.
    The environment generates reward $r_t$ according to arm $I_t$.
    The agent receive $r_t$.
    $V_{I_t} \leftarrow V_{I_t} + x_t x_t^T$, $b_{I_t} \leftarrow b_{I_t} + r_t x_t$, $\hat{\theta}_{I_t} \leftarrow V_{I_t}^{-1} b_{I_t}$.
end for
```

### 3.2.2 White-box attack

We first consider the white-box attack scenario, in which the attacker has knowledge of the environment. In particular, in the white-box attack scenario, the attacker knows the coefficient vectors $\theta_i$'s for all $i$. The understanding of this scenario could provide useful insights for the more practical black-box attack scenario to be discussed in Section 3.2.3.

Our proposed attack strategy works as follows. When the agent chooses arm $K$, the attacker does not attack. When the agent chooses arm $I_t \neq K$, the attacker changes it to arm

$$I_t^0 = \begin{cases} K & \text{with probability} \quad \epsilon_t \\ I_t^\dagger & \text{with probability} \quad 1 - \epsilon_t \end{cases} \tag{3.4}$$

where $I_t^\dagger = \arg\min_i \langle x_t, \theta_i \rangle$ and

$$\epsilon_t = \frac{(1-\alpha)\langle x_t, \theta_K \rangle - \min_i \langle x_t, \theta_i \rangle}{\langle x_t, \theta_K \rangle - \min_i \langle x_t, \theta_i \rangle}. \tag{3.5}$$

We now explain the main idea behind the attack strategy specified in (3.4) and (3.5). Intuitively speaking, using (3.4) and (3.5), the attacker can manipulate the agent into learning some particular coefficient vectors. In particular, for arm $K$ (the target arm), the agent obtains the true coefficient vector $\theta_K$. For any arm $i \neq K$, because of the attacks, the agent will obtain an estimate of arm $i$'s coefficient vector to be close to $(1-\alpha)\theta_K$. This can be seen as follows. Under Assumption 1 and the fact that the mean rewards are positive, we have that $1/2 < \epsilon_t < 1 - \alpha$. Under the white-box attack, when the agent pulls a non-target arm $I_t \neq K$, the mean reward received by the agent should satisfy

$$\begin{aligned} &\mathbb{E}[r_{t,I_t^0} | F_{t-1}, I_t] = \mathbb{E}[\langle x_t, \theta_{I_t^0} \rangle | F_{t-1}, I_t] \\ &= \epsilon_t \langle x_t, \theta_K \rangle + (1-\epsilon_t) \min_i \langle x_t, \theta_i \rangle = (1-\alpha)\langle x_t, \theta_K \rangle. \end{aligned}$$

Because of the positive mean reward assumption and the fact that $0 < 1 - \alpha < 1$, the target arm $K$ appears to be the optimal arm for any context. Hence the proposed white-box attack strategy can easily mislead an agent who employs a contextual bandit algorithm with sublinear pseudo-regret, as shown in the following proposition whose proof can be found in Appendix B.1.1. The proposed attack strategy can be further extended to the generalized linear contextual bandit model. We discuss the generalized linear contextual bandit model in Section 3.3.

**Proposition 2.** For $\delta > 0$. Suppose the agent employs a linear contextual bandit algorithm whose cumulative pseudo-regret is upper bounded by $o(T)$ over $T$ with probability at least $1 - \delta$. When using the proposed white-box attack scheme, with probability at least $1 - \delta$, the attacker can force the agent to pull the target arm for $T - o(T)$ times, while the total attack cost $|\mathcal{C}|$ is upper bounded by $o(T)$.

Proposition 2 applies to any contextual bandit algorithm whose cumulative pseudo-regret is upper bounded by $o(T)$. To further illustrate the proposed attack scheme, we now provide a finer analysis the impact of this attack on LinUCB described in Algorithm 3.1. We first build a high probability bound for the estimate of parameter at the agent, when the agent computes the estimate of $\theta_i$ by (3.3) and the attacker employs the white-box attack. Recall that $\omega(N) = \sqrt{\lambda}S + R\sqrt{2\log K/\delta + d\log\left(1 + L^2N/(\lambda d)\right)}$.

**Lemma 6.** Under the proposed white-box attack, the estimate of $\theta_i$ for each arm $i$ obtained by LinUCB agent as described in Algorithm 3.1 satisfies

$$|x_t^T\hat{\theta}_{t,i} - x_t^T(1-\alpha)\theta_K| \leq \left(\omega(N_i(t)) + LS\sqrt{0.5\log\left(2KT/\delta\right)}\right)\|x_t\|_{V_{t,i}^{-1}}, \tag{3.6}$$

with probability $1 - 2(K-1)\delta/K$, for all arm $i \neq K$ and all $t \geq 0$. Here, $\|x\|_V = \sqrt{x^TVx}$ is the weighted norm of vector $x$ for a positive definite matrix $V$.

The proof of Lemma 6 is provided in Appendix B.1.2. Lemma 6 shows that, under our white-box attack, the agent's estimate of the parameter of non-target arm, i.e. $\hat{\theta}_i$, will converge to $(1-\alpha)\theta_K$. Thus, the agent is misled to believe that arm $K$ is the optimal arm for every context in most rounds. The following theorem provides an upper bound of the cumulative cost of the attack.

**Theorem 4.** Define $\gamma = \min_{x \in \mathcal{D}} \langle x, \theta_K \rangle$. Under the same assumptions as in Lemma 6, for any $\delta > 0$ with probability at least $1 - 2\delta$, for all $T \geq 0$, the attacker can manipulate the LinUCB agent into pulling the target arm in at least $T - |\mathcal{C}|$ rounds, using an attack cost

$$|\mathcal{C}| \leq \frac{2d(K-1)}{(\alpha\gamma)^2} \log\left(1 + TL^2/(d\lambda)\right) \left(2\omega(T) + LS\sqrt{0.5 \log(2KT/\delta)}\right)^2 . \tag{3.7}$$

The proof of Theorem 4 is provided in Appendix B.1.3. Theorem 4 shows that our white-box attack strategy can force LinUCB agent into pulling the target arm $T - O(\log^2 T)$ times with attack cost scaled only as $O(\log^2 T)$.

### 3.2.3 Black-box attack

We now focus on the more practical black-box setting, in which the attacker does not know any of arm's coefficient vector. The attacker knows the value of $\alpha$ (or a lower bound) in which the equation (3.2) holds for all $t$. Although the attacker does not know the coefficient vectors for all arms, the attacker can compute an estimate of the unknown parameters by using past observations. On the other hand, there are multiple challenges brought by the estimation errors that need to properly addressed.

The proposed black-box attack strategy works as follows. When the agent chooses arm $K$, the attacker does not attack. When the agent chooses arm $I_t \neq K$, the attacker changes it to arm

$$I_t^0 = \begin{cases} K & \text{with probability} \quad \epsilon_t \\ I_t^\dagger & \text{with probability} \quad 1 - \epsilon_t \end{cases} \tag{3.8}$$

where

$$I_t^\dagger = \arg\min_{i \neq K} \left( \langle x_t, \hat{\theta}_{t,i}^0 \rangle - \beta_{t,i}^0 \|x_t\|_{(V_{t,i}^0)^{-1}} \right), \tag{3.9}$$

and

$$\beta_{t,i}^0 = \phi_i \left( \omega(N_i^\dagger(t)) + LS\sqrt{0.5 \log(2KT/\delta)} \right), \tag{3.10}$$

$\phi_i = 1/\alpha$ when $i \neq K$ and $\phi_K = 2$, and

$$\epsilon_t = \text{clip}\left(\frac{1}{2}, \frac{(1-\alpha)\langle x_t, \hat{\theta}^0_{t,K}\rangle - \langle x_t, \hat{\theta}^0_{t,I^\dagger_t}\rangle}{\langle x_t, \hat{\theta}^0_{t,K}\rangle - \langle x_t, \hat{\theta}^0_{t,I^\dagger_t}\rangle}, 1-\alpha\right), \tag{3.11}$$

with clip$(a, x, b) = \min(b, \max(x, a))$ where $a \leq b$.

For notational convenience, we set $I^\dagger_t = K$ and $\epsilon_t = 1$ when $I_t = K$. We define that, if $i \neq K$, $\tau^\dagger_i(t) := \{s : s \leq t, I^\dagger_s = i\}$ and $N^\dagger_i(t) = |\tau^\dagger_i(t)|$ ; $\tau^\dagger_K(t) := \{s : s \leq t\}$ and $N^\dagger_K(t) = |\tau^\dagger_K(t)|$.

$$\hat{\theta}^0_{t,i} = \left(V^0_{t,i}\right)^{-1}\left(\sum_{k\in\tau^\dagger_i(t-1)} w_{k,i} r_{k,I^0_k} x_k\right), \tag{3.12}$$

where $V^0_{t,i} = \sum_{k\in\tau^\dagger_i(t-1)} x_k x_k^T + \lambda\mathbf{I}$ and

$$w_{t,i} = \begin{cases} 1/\epsilon_t & \text{if } i = I^0_t = K \\ 1/(1-\epsilon_t) & \text{if } i = I^0_t = I^\dagger_t \\ 0 & \text{if } i \neq I^0_t \end{cases}. \tag{3.13}$$

Here, $\hat{\theta}^0_{t,i}$ is the estimation of $\theta_i$ by the attacker, while $\hat{\theta}_{t,i}$ in (3.3) is the estimation of $\theta_i$ at the agent side. We will show in Lemma 7 and Lemma 9 that $\hat{\theta}^0_{t,i}$ will be close to the true value of $\theta_i$ while $\hat{\theta}_{t,i}$ will be close to a sub-optimal value chosen by the attacker. This disparity gives the attacker the advantage for carrying out the attack.

We now highlight the main idea why our black-box attack strategy works. As discussed in Section 3.2.2, if the attacker knows the coefficient vectors of all arms, the proposed white-box attack scheme can mislead the agent to believe that the coefficient vector of every non-target arm is $(1-\alpha)\theta_K$, hence the agent will think the target arm is optimal. In the black-box setting, the attacker does not know the coefficient vector for any arm. The attacker should estimate the coefficient vector of each arm. Then, the attacker will use the estimated coefficient vector to replace the true coefficient vector in the white-box attack scheme. As the attacker does not know the true values of

$\theta_i$'s, we need to design the estimator $\hat{\theta}^0_{t,i}$, the attack choice $I^\dagger_t$ and the probability $\epsilon_t$ carefully. In the following, we explain the main ideas behind our design choices.

Firstly, we explain why we design estimator $\hat{\theta}^0_{t,i}$ using the form (3.12), in which the attacker employs the importance sampling to obtain an estimate of $\theta_i$. There are two reasons for this. Firstly, for a successful attack, the number of observation in arm $i \neq K$ will be limited. Hence if the importance sampling is not used, the estimation variance of the mean reward $\langle x, \theta_i \rangle$ at the attacker side for some contexts $x$ may be large. Secondly, the attacker's action is stochastic when the agent pulls a non-target arm. Thus, the attacker uses the observations at round $t$ when the attacker pulls arm $i$ with certain probability, i.e. when $t \in \tau^\dagger_i$, to estimate $\theta_i$. At the agent side, since the agent's action is deterministic, the agent uses the observations at round $t$ when the agent pulls arm $i$, i.e. when $t \in \tau_i$, to estimate $\theta_i$.

Secondly, we explain ideas behind the choice of $I^\dagger_t$ in (3.9). Under our black-box attack, when the agent pulls a non-target arm $I_t \neq K$, the mean reward received by the agent satisfies

$$\begin{aligned}\mathbb{E}[r_{t,I^0_t}|F_{t-1}, I_t] &= \mathbb{E}[\langle x_t, \theta_{I^0_t}\rangle|F_{t-1}, I_t] \\ &= \epsilon_t\langle x_t, \theta_K\rangle + (1-\epsilon_t)\langle x_t, \theta_{I^\dagger_t}\rangle.\end{aligned} \tag{3.14}$$

In white-box attack scheme, $I^\dagger_t$ is the worst arm at context $x_t$. In the black-box setting, the attacker does not know a prior which arm is the worst. In the proposed black-box attack scheme, as indicated in (3.9), we use the lower confidence bound (LCB) method to explore the worst arm and $I^\dagger_t$ is the arm whose lower confidence bound is the smallest.

Finally, we provide reasons why we choose $\epsilon_t$ using (3.11). In our white-box attack scheme, we have that $1/2 < \epsilon_t < 1 - \alpha$. Thus, in our black-box attack scheme, we limit the choice of $\epsilon_t$ to $[1/2, 1-\alpha]$. Furthermore, in (3.5) used for the white-box attack, $\epsilon_t$ is computed by the true mean reward. Now, in the black-box attack, as the attacker does not the true coefficient vector, the attacker use the estimation of $\theta$ to compute the second term in the clip function in (3.11).

In summary, our design of $\hat{\theta}^0_{t,i}$, $I^\dagger_t$ and $\epsilon_t$ can ensure that the attacker's estimation $\hat{\theta}^0_{t,i}$ is close to $\theta_i$, while the agent's estimation $\hat{\theta}_{t,i}$ will be close to $(1-\alpha)\theta_K$. In the following, we make these statements precise, and formally analyze the performance of the proposed black-box attack scheme.

First, we analyze the estimation $\hat{\theta}^0_{t,i}$ at the attacker side. We establish a confidence ellipsoid of $\langle x_t, \hat{\theta}^0_{t,i}\rangle$ at the attacker.

**Lemma 7.** Assume the attacker performs the proposed black-box action poisoning attack. With probability $1-2\delta$, we have

$$|x_t^T\hat{\theta}^0_{t,i} - x_t^T\theta_i| \leq \beta^0_{t,i}\|x_t\|_{\left(V^0_{t,i}\right)^{-1}}. \tag{3.15}$$

holds for all arm $i$ and all $t \geq 0$ simultaneously.

Lemma 7 shows that $\hat{\theta}^0_i$ lies in an ellipsoid with center at $\theta_i$ with high probability, which implies that the attacker has good estimate of each arm.

We then analyze the estimation $\hat{\theta}_{t,i}$ at the agent side. The following lemma provides an upper bound on the absolute difference between $\mathbb{E}[r_{t,I_t^0}|F_{t-1}, I_t]$ and $(1-\alpha)\langle x_t, \theta_K\rangle$.

**Lemma 8.** Under the black-box attack, with probability $1-2\delta$, the estimate obtained by an LinUCB agent satisfies

$$\left|\mathbb{E}[r_{t,I_t^0}|F_{t-1}, I_t] - (1-\alpha)\langle x_t, \theta_K\rangle\right| \leq (1-\alpha)\beta^0_{t,K}\|x_t\|_{\left(V^0_{t,K}\right)^{-1}} + (1+\alpha)\beta^0_{t,I_t^\dagger}\|x_t\|_{\left(V^0_{t,I_t^\dagger}\right)^{-1}}$$

simultaneously for all $t \geq 0$ when $I_t \neq K$.

The bound in Lemma 8 consists of the confidence ellipsoid of the estimate of arm $I_t^\dagger$ and that of arm $K$. As mentioned above, for a successful attack, the number of observations on arm $I_t^\dagger$ will be limited. Thus, in our proposed algorithm, the attacker use the importance sampling to obtain the estimate of $\theta_i$, which will increase the number of observations that can be used to estimate the coefficient vector of arm $I_t^\dagger$. Using Lemma 8, we have the following lemma regarding the estimation $\hat{\theta}_{t,i}$ at the agent side.

**Lemma 9.** Consider the same assumption as in Lemma 7. With a probability at least $1 - \frac{(3K-1)\delta}{K}$, the estimate $\hat{\theta}_{t,i}$ obtained by the LinUCB agent will satisfy

$$
\begin{aligned}
&|x_t^T \hat{\theta}_{t,i} - x_t^T (1-\alpha)\theta_K| \\
&\leq \left(1 + 4d/\alpha \sqrt{K \log\left(1 + tL^2/(d\lambda)\right)}\right) \left(\omega(N_i(t)) + LS\sqrt{0.5 \log\left(2KT/\delta\right)}\right) \|x_t\|_{V_{t,i}^{-1}}
\end{aligned}
\tag{3.16}
$$

simultaneously for all arm $i \neq K$ and all $t \geq 0$.

Lemma 9 shows that, under the proposed black-box attack scheme, the agent's estimate of the parameter of non-target arm, i.e. $\hat{\theta}_i$, will converge to $(1-\alpha)\theta_K$. Hence the agent will believe that the target arm $K$ is the optimal arm for any context in most rounds. Using these supporting lemmas, we can then analyze the performance of the proposed black-box attack strategy.

**Theorem 5.** Define $\gamma = \min_{x \in \mathcal{D}} \langle x, \theta_K \rangle$. Under the same assumptions as in Lemma 9, with probability at least $1 - 3\delta$, for all $T \geq 0$, the attacker can manipulate a LinUCB agent into pulling the target arm in at least $T - |\mathcal{C}|$ rounds, using an attack cost

$$
\begin{aligned}
|\mathcal{C}| \leq \frac{2d(K-1)}{(\alpha\gamma)^2} &\left(2 + \frac{4d}{\alpha}\sqrt{K \log\left(1 + \frac{TL^2}{d\lambda}\right)}\right)^2 \times \\
\log\left(1 + \frac{TL^2}{d\lambda}\right) &\left(\omega(T) + LS\sqrt{0.5 \log\left(2KT/\delta\right)}\right)^2 .
\end{aligned}
\tag{3.17}
$$

Theorem 5 shows that our black-box attack strategy can manipulate a LinUCB agent into pulling a target arm $T - O(\log^3 T)$ times with attack cost scaling as $O(\log^3 T)$. Compared with the result for the white-box attack, the black-box attack only brings an additional $\log T$ factor.

## 3.3 Generalized Linear Model

In this section, we extend the proposed attack strategy to the generalized linear contextual bandit model. In the generalized linear model (GLM), there is a fixed, strictly increasing link function $\mu : \mathbb{R} \to \mathbb{R}$ such that the reward satisfies $r_{t,I_t} = \mu(\langle x_t, \theta_{I_t} \rangle) + \eta_t$, Hence, the expected reward of arm $i$ under context $x_t$ follows the GLM setting: $\mathbb{E}[r_{t,i}] = \mu(\langle x_t, \theta_i \rangle)$ for all $t$ and all arm $i$. One can verify that $\mu(x) = x$ leads to the linear model and $\mu(x) = \exp(x)/(1 + \exp(x))$ leads to the logistic model.

We assume that the link function $\mu$ is continuously twice differentiable, Lipschitz with constant $k_\mu$ and such that $c_\mu = \inf_{\theta \in \Theta, x \in \mathcal{D}} \dot{\mu}(x^T \theta) > 0$, where $\dot{\mu}$ denote the first derivatives of $\mu$. It can be verified that the link function of the linear model is Lipschitz with constant $k_\mu = 1$ and which of the logistic model is Lipschitz with constant $k_\mu = 1/4$.

The agent is interested in minimizing the cumulative pseudo-regret, and the cumulative pseudo-regret for the GLM can be formally written as

$$R_T = \sum_{t=1}^{T} \left( \mu(\langle x_t, \theta_{I_t^*} \rangle) - \mu(\langle x_t, \theta_{I_t} \rangle) \right), \tag{3.18}$$

where $I_t^* = \arg\max_i \mu(\langle x_t, \theta_i \rangle)$.

For the GLM considered here, since $\mu$ is a strictly increasing function, $I_t^* = \arg\max_i \mu(\langle x_t, \theta_i \rangle) = \arg\max_i \langle x_t, \theta_i \rangle$. Thus, our proposed attack strategy can be extended to the generalized linear contextual bandit model.

When the agent chooses arm $K$, the attacker does not attack. When the agent chooses arm $I_t \neq K$, the attacker changes it to arm

$$I_t^0 = \begin{cases} K & \text{with probability} \quad \epsilon_t \\ I_t^\dagger & \text{with probability} \quad 1 - \epsilon_t \end{cases} \tag{3.19}$$

where $I_t^\dagger = \arg\min_i \langle x_t, \theta_i \rangle$ and

$$\epsilon_t = \frac{\mu((1-\alpha)\langle x_t, \theta_K \rangle) - \mu(\min_i \langle x_t, \theta_i \rangle)}{\mu(\langle x_t, \theta_K \rangle) - \mu(\min_i \langle x_t, \theta_i \rangle)}. \tag{3.20}$$

Similar as the proposed attack strategy on linear bandit model, the attacker can manipulate the agent into learning some particular coefficient vectors. In particular, for arm $K$ (the target arm), the agent obtains the true coefficient vector $\theta_K$. For any arm $i \neq K$, because of the attacks, the agent will obtain an estimate of arm $i$'s coefficient vector to be close to $(1-\alpha)\theta_K$. Under the white-box attack, when the agent pulls a non-target arm $I_t \neq K$, the mean reward received by the agent should satisfy

$$\mathbb{E}[r_{t,I_t^0}|F_{t-1}, I_t] = \mu((1-\alpha)\langle x_t, \theta_K\rangle).$$

**Proposition 3.** For $\delta > 0$, suppose the agent employs a generalized linear contextual bandit algorithm whose cumulative pseudo-regret is upper bounded by $o(T)$ over $T$ with probability at least $1-\delta$. When using the proposed white-box attack scheme, with probability at least $1-\delta$, the attacker can force the agent to pull the target arm for $T - o(T)$ times, while the total attack cost $|\mathcal{C}|$ is upper bounded by $o(T)$.

*Proof.* Let us consider a contextual bandit problem $P_1$, in which the arm $K$ (the target arm) is associated with a fixed coefficient vector $\theta_K$ and all other arms are associated with the coefficient vector $(1-\alpha)\theta_K$. For a generalized linear contextual bandit algorithm $\mathscr{A}$, we suppose that the cumulative pseudo-regret regret of algorithm $\mathscr{A}$ for the problem $P_1$ is upper bounded with probability at least $1-\delta$ by a function $f_{\mathscr{A}}(T)$ such that $f_{\mathscr{A}}(T) = o(T)$.

Under the proposed white-box attack, when the agent pulls a non-target arm $I_t \neq K$, the mean reward received by the agent should satisfy $\mathbb{E}[r_{t,I_t^0}|F_{t-1}, I_t] = \mu((1-\alpha)\langle x_t, \theta_K\rangle)$. In the observation of the agent, the target arm becomes optimal and the non-target arms are associated with the coefficient vector $(1-\alpha)\theta_K$. For the agent, the combination of the attacker and the environment form problem $P_1$. The cumulative pseudo-regret should satisfy

$$\begin{aligned} R_T =& \sum_{t=1}^{T} \mathbb{1}_{\{I_t \neq K\}} \left(\mu\left(\langle x_t, \theta_K\rangle\right) - \mu\left(\langle x_t, (1-\alpha)\theta_K\rangle\right)\right) \\ \geq& \sum_{t=1}^{T} \mathbb{1}_{\{I_t \neq K\}} c_\mu \langle x_t, \alpha\theta_K\rangle \\ \geq& \sum_{t=1}^{T} \mathbb{1}_{\{I_t \neq K\}} c_\mu LS, \end{aligned}$$

which is equivalent to $\sum_{t=1}^{T} \mathbb{1}_{\{I_t \neq K\}} \leq R_T/(c_\mu LS)$. Since $R_T$ is upper bounded by $f_{\mathscr{A}}(T) = o(T)$, $|\mathcal{C}| = \sum_{t=1}^{T} \mathbb{1}_{\{I_t \neq K\}}$ is also upper bounded by $o(T)$. □

To further illustrate the proposed attack scheme, we now provide a finer analysis the impact of this attack on GLM-UCB described in Algorithm 3.2. The algorithm is simply initialized by playing every arm $j$ times to ensure a unique solution of $\hat{\theta}_i$ for each arm $i$. We assume that after playing arm $i$ $J$ times, $V_i$ is invertible and the minimal eigenvalue of $V_i$ is greater or equal to $\lambda_0$ for all arm $i$. We assume that $x_t$ is drawn iid from some distribution $v$ with support in the unit ball and set $\Sigma := \mathbb{E}[x_t x_t^T]$. Proposition 1 in [54] shows that there exist positive, universal constants $D_1$ and $D_2$ such that $\lambda_{\min}(V_i) \geq \lambda_0$ with probability at least $1 - \delta$, as long as

$$J \geq \left( \frac{D_1\sqrt{d} + D_2\sqrt{\log(1/\delta)}}{\lambda_{\min}(\Sigma)} \right)^2 + \frac{2\lambda_0}{\lambda_{\min}(\Sigma)}. \tag{3.21}$$

**Algorithm 3.2** GLM-UCB [54]

```
Require:
    number of arms $K$, number of rounds $T$, number of initial rounds $j$.
Initialize for every arm $i$.
Play every arm $J$ times. At each time, update $\bar{V}_{I_t} \leftarrow V_{I_t} + x_t x_t^T$.
for $t = KJ+1, KJ+2, \ldots, T$ do
    observe the context $x_t$.
    for $i = 1, 2, \ldots, K$ do
        Calculate the maximum-likelihood estimator $\hat{\theta}_i$ by solving the equation
            $$\sum_{n \in \tau_i(t-1)} (r_n - \mu(x_n^T \hat{\theta}_i)) x_n = 0.$$
        Compute the upper confidence bound: $p_{t,i} \leftarrow x_t^T \hat{\theta}_i + \beta_{t,i}\sqrt{x_t^T \bar{V}_i^{-1} x_t}$.
    end for
    Pull arm $I_t = \arg\max_i p_{t,i}$.
    The environment generates reward $r_t$ according to arm $I_t$.
    The agent receive $r_t$.
    $\bar{V}_{I_t} \leftarrow \bar{V}_{I_t} + x_t x_t^T$.
end for
```

By following the setup in [54], we set

$$\beta_{t,i} = \frac{2R}{c_\mu}\sqrt{\log\frac{K}{\delta} + d\log\left(1 + \frac{L^2 N_i(t)}{\lambda_0 d}\right)}.$$

**Lemma 10.** Under the proposed white-box attack, the estimate of $\theta_i$ for each arm $i \neq K$ obtained by GLM-UCB agent as described in Algorithm 3.2 satisfies

$$|x_t^T \hat{\theta}_{t,i} - x_t^T(1-\alpha)\theta_K| \leq \frac{2k_\mu LS + 2R}{c_\mu}\sqrt{\log\frac{K}{\delta} + d\log\left(1 + \frac{L^2 N_i(t)}{\lambda_0 d}\right)}\|x\|_{\bar{V}_{t,i}^{-1}}. \tag{3.22}$$

The proof of Lemma 10 is provided in Appendix B.3.1.

**Theorem 6.** Define $\gamma = \min_{x\in\mathcal{D}}\langle x, \theta_K\rangle$. Under the same assumptions as in Lemma 6, for any $\delta > 0$ with probability at least $1 - 2\delta$, for all $T \geq 0$, the attacker can manipulate the LinUCB agent into pulling the target arm in at least $T - |\mathcal{C}|$ rounds, using an attack cost

$$|\mathcal{C}| \leq \frac{4d(K-1)}{(\alpha\gamma)^2}\log\left(1 + \frac{tL^2}{d\lambda_0}\right)\left(\frac{2k_\mu LS + 4R}{c_\mu}\right)^2\left(\log\frac{K}{\delta} + d\log\left(1 + \frac{L^2 T}{\lambda_0 d}\right)\right). \tag{3.23}$$

The proof of Theorem 6 is provided in Appendix B.3.2.

The proposed black-box attack strategy works as follows. When the agent chooses arm $K$, the attacker does not attack. When the agent chooses arm $I_t \neq K$, the attacker changes it to arm

$$I_t^0 = \begin{cases} K & \text{with probability} \quad \epsilon_t \\ I_t^\dagger & \text{with probability} \quad 1 - \epsilon_t \end{cases} \tag{3.24}$$

where

$$I_t^\dagger = \arg\min_{i\neq K}\left(\langle x_t, \hat{\theta}_{t,i}^0\rangle - \beta_{t,i}^0\|x_t\|_{(\bar{V}_{t,i}^0)^{-1}}\right), \tag{3.25}$$

and

$$\beta_{t,i}^0 = 2\phi_i\frac{k_\mu LS + R}{c_\mu}\sqrt{\log\frac{K}{\delta} + d\log\left(1 + \frac{L^2 N_i(t)}{\lambda_0 d}\right)}, \tag{3.26}$$

$\phi_i = \frac{k_\mu}{c_\mu \alpha}$ when $i \neq K$ and $\phi_K = 1 + \frac{k_\mu}{c_\mu}$, and

$$\epsilon_t = \text{clip}\left( \frac{c_\mu}{c_\mu + k_\mu}, \frac{\mu((1-\alpha)x_t^T \hat{\theta}^0_{t,K}) - \mu(x_t^T \hat{\theta}^0_{t,I_t^\dagger})}{\mu(x_t^T \hat{\theta}^0_{t,K}) - \mu(x_t^T \hat{\theta}^0_{t,I_t^\dagger})}, 1 - \alpha \frac{c_\mu}{k_\mu} \right), \tag{3.27}$$

with clip$(a, x, b) = \min(b, \max(x, a))$ where $a \leq b$.

For notational convenience, we set $I_t^\dagger = K$ and $\epsilon_t = 1$ when $I_t = K$. We define that, if $i \neq K$, $\tau_i^\dagger(t) := \{s : s \leq t, I_s^\dagger = i\}$ and $N_i^\dagger(t) = |\tau_i^\dagger(t)|$ ; $\tau_K^\dagger(t) := \{s : s \leq t\}$ and $N_K^\dagger(t) = |\tau_K^\dagger(t)|$.

Calculate the maximum-likelihood estimator $\hat{\theta}^0_{t,i}$ by solving the equation

$$\sum_{n \in \tau_i(t-1)^\dagger} (w_{t,i} r_n - \mu(x_n^T \hat{\theta}_{t,i})) x_n = 0.$$

where $\bar{V}^0_{t,i} = \sum_{k \in \tau_i^\dagger(t-1)} x_k x_k^T$

$$w_{t,i} = \begin{cases} 1/\epsilon_t & \text{if } i = I_t^0 = K \\ 1/(1-\epsilon_t) & \text{if } i = I_t^0 = I_t^\dagger \\ 0 & \text{if } i \neq I_t^0 \end{cases}. \tag{3.28}$$

First, we analyze the estimation $\hat{\theta}^0_{t,i}$ at the attacker side. We establish a confidence ellipsoid of $\langle x_t, \hat{\theta}^0_{t,i} \rangle$ at the attacker.

**Lemma 11.** Assume the attacker performs the proposed black-box action poisoning attack. With probability $1 - 2\delta$, we have

$$|x_t^T \hat{\theta}^0_{t,i} - x_t^T \theta_i| \leq \beta^0_{t,i} \|x_t\|_{(\bar{V}^0_{t,i})^{-1}}. \tag{3.29}$$

holds for all arm $i$ and all $t \geq 0$ simultaneously.

The proof of Lemma 11 is provided in Appendix B.3.3.

**Lemma 12.** Under the black-box attack, with probability $1 - 2\delta$, the estimate obtained by an GLM-UCB agent satisfies

$$\left|\mathbb{E}[r_{t,I_t^0}|F_{t-1}, I_t] - \mu((1-\alpha)x_t^T\theta_K)\right| \leq 2k_\mu\beta_{t,K}^0\|x_t\|_{\left(\bar{V}_{t,K}^0\right)^{-1}} + 2k_\mu\beta_{t,I_t^\dagger}^0\|x_t\|_{\left(\bar{V}^0_{t,I_t^\dagger}\right)^{-1}}.$$

simultaneously for all $t \geq 0$ when $I_t \neq K$.

**Lemma 13.** Consider the same assumption as in Lemma 7. With a probability at least $1 - \frac{3K\delta}{K}$, the estimate $\hat{\theta}_{t,i}$ obtained by the GLM-UCB agent will satisfy

$$\begin{aligned}
&|x_t^T\hat{\theta}_{t,i} - x_t^T(1-\alpha)\theta_K| \\
\leq &\left(1 + \frac{16k_\mu^2 d}{c_\mu\alpha}\sqrt{K\log\left(1 + \frac{tL^2}{d\lambda_0}\right)}\right)\frac{2k_\mu LS + 2R}{c_\mu}\sqrt{\log\frac{K}{\delta} + d\log\left(1 + \frac{L^2t}{\lambda_0 d}\right)}\|x\|_{\bar{V}_{t,i}^{-1}}
\end{aligned} \tag{3.30}$$

simultaneously for all arm $i \neq K$ and all $t \geq 0$.

**Theorem 7.** Define $\gamma = \min_{x\in\mathcal{D}}\langle x, \theta_K\rangle$. Under the same assumptions as in Lemma 6, for any $\delta > 0$ with probability at least $1 - 2\delta$, for all $T \geq 0$, the attacker can manipulate the LinUCB agent into pulling the target arm in at least $T - |\mathcal{C}|$ rounds, using an attack cost

$$\begin{aligned}
|\mathcal{C}| \leq &\frac{4d(K-1)}{(\alpha\gamma)^2}\left(\frac{2k_\mu LS + 2R}{c_\mu}\right)^2\log\left(1 + \frac{TL^2}{d\lambda_0}\right) \\
&\times\left(\log\frac{K}{\delta} + d\log\left(1 + \frac{L^2T}{\lambda_0 d}\right)\right)\left(1 + \frac{16k_\mu^2 d}{c_\mu\alpha}\sqrt{K\log\left(1 + \frac{TL^2}{d\lambda_0}\right)}\right)^2.
\end{aligned} \tag{3.31}$$

The proof of Theorem 7 is provided in Appendix B.3.6. Theorem 7 shows that our black-box attack strategy can manipulate a GLM-UCB agent into pulling a target arm $T - O(\log^3 T)$ times with attack cost scaling as $O(\log^3 T)$. Compared with the result for the white-box attack, the black-box attack only brings an additional $\log T$ factor.

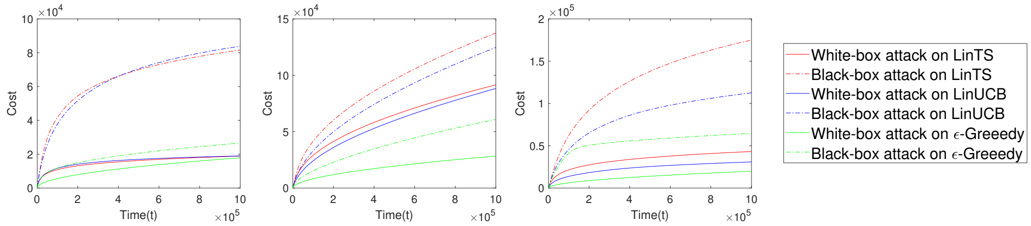


Figure 3.2: The cumulative cost of the attacks for the synthetic (Left), Jester (Center) and MovieLens (Right) datasets.

| | Synthetic | Jester | MovieLens |
|---|---|---|---|
| $\epsilon$-Greeedy without attacks | 2124.6 | 5908.7 | 3273.5 |
| White-box attack on $\epsilon$-Greeedy | 982122.5 | 971650.9 | 980065.6 |
| Black-box attack on $\epsilon$-Greeedy | 973378.5 | 939090.2 | 935293.8 |
| LinUCB without attacks | 8680.9 | 16927.2 | 13303.4 |
| White-box attack on LinUCB | 981018.7 | 911676.9 | 969118.6 |
| Black-box attack on LinUCB | 916140.8 | 875284.7 | 887373.1 |
| LinTS without attacks | 5046.9 | 18038.0 | 9759.0 |
| White-box attack on LinTS | 981112.8 | 908488.3 | 956821.1 |
| Black-box attack on LinTS | 918403.8 | 862556.8 | 825034.8 |

Table 3.1: Average number of rounds when the agent pulls the target arm over $T = 10^6$ rounds.

## 3.4 Numerical Experiments

In this section, we empirically evaluate the performance of the proposed action poisoning attack schemes on three contextual bandit algorithms: LinUCB [1], LinTS [2], and $\epsilon$-Greedy. We run the experiments on three datasets:

**Synthetic data:** The dimension of contexts and the coefficient vectors is $d = 6$. We set the first entry of every context and coefficient vector to $1$. The other entries of every context and coefficient vector are uniformly drawn from $(-\frac{1}{\sqrt{d-1}}, \frac{1}{\sqrt{d-1}})$. Thus, for all round $t$ and arm $i$, $\|x_t\|_2 \leq \sqrt{2}$, $\|\theta_i\|_2 \leq \sqrt{2}$ and mean rewards $\langle x_t, \theta_i \rangle > 0$. The reward noise $\eta_t$ is drawn from a Gaussian distribution $\mathcal{N}(0, 0.01)$.

**Jester dataset [28]:** Jester contains 4.1 million ratings of jokes in which the rating values scale from $-10.00$ to $+10.00$. We normalize the rating to $[0, 1]$. The dataset includes 100 jokes and the ratings were collected from 73,421 users between April 1999 - May 2003. We consider a subset of 10 jokes and 38432 users. Every jokes are rated by each user. We perform a low-rank matrix factorization ($d = 6$) on the ratings data and obtain the features for both users and jokes. At each round, the environment randomly select a user as the context and the reward noise is drawn from a Gaussian distribution $\mathcal{N}(0, 0.01)$.

**MovieLens 25M dataset: [34]** MovieLens 25M dataset contains 25 million 5-star ratings of 62,000 movies by 162,000 users. The preprocessing of this data is almost the same as the Jester dataset, except that we consider a subset of 10 movies and 7344 users. At each round, the environment randomly select a user as the context and the reward noise is drawn from $\mathcal{N}(0, 0.01)$.

We set $\delta = 0.1$ and $\lambda = 2$. For all the experiments, we set the total number of rounds $T = 10^6$ and the number of arms $K = 10$. We independently run ten repeated experiments. Results reported are averaged over the ten experiments. We set $\alpha$ to $0.2$ for the two proposed attack strategies, hence the target arm may be the worst arm in some rounds.

The results are shown in Table 3.1 and Figure 3.2. These experiments show that the action poisoning attacks can force the three agents to pull the target arm very frequently, while the agents rarely pull the target arm under no attack. Under the attacks, the true regret of the agent becomes linear as the target arm is not optimal for most context. Table 3.1 show the number of rounds the agent pulls the target arm among $T = 10^6$ total rounds. In the synthetic dataset, under the proposed white-box attacks, the target arm is pulled more than $98.1\%$ of the times by the three agent (see Table 3.1). The target arm is pulled more than $91.6\%$ of the times in the worst case (the black-box attacks on LinUCB). Fig 3.2 shows the cumulative cost of the attacks on three agents for the three datasets. The results show that the attack cost $|\mathcal{C}|$ of every attack scheme on every agent for every dataset scales sublinearly, which exposes a significant security threat of the action poisoning attacks on linear contextual bandits. These results also illustrate that, even though our theoretical results are derived under Assumption 1, the proposed attack strategies still work in practical scenarios where Assumption 1 may not be strictly satisfied.

## 3.5 Conclusion

In this chapter, we have proposed a class of action poisoning attacks on linear contextual bandits. We have shown that our white-box attack strategy is able to force any linear contextual bandit agent, whose regret scales sublinearly with the total number of rounds, into pulling a target arm chosen by the attacker. We have also shown that our white-box attack strategy can force LinUCB agent into pulling a target arm $T - O(\log^2 T)$ times with attack cost scaled as $O(\log^2 T)$. We have further shown that the proposed black-box attack strategy can force LinUCB agent into pulling a target arm $T - O(\log^3 T)$ times with attack cost scaled as $O(\log^3 T)$. Our results expose a significant security threat to contextual bandit algorithms.

# Chapter 4

# Action Attacks on Reinforcement Learning

In this chapter, we study the action poisoning attack on RL in both white-box and black-box settings. We introduce an adaptive attack scheme called LCB-H, which works for most RL agents in the black-box setting. In Section 4.1, we describe the model. In Section 4.2, we describe the attack strategies and analyze their accumulative attack cost. In Section 4.3, we provide numerical examples to validate the theoretic analysis. The proofs are collected in Appendix C.

## 4.1 Problem Formulation

Consider a tabular episodic MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, H, P, R)$, where $\mathcal{S}$ is the state space with $|\mathcal{S}| = S$, $\mathcal{A}$ is the action space with $|\mathcal{A}| = A$, $H \in \mathbb{Z}^+$ is the number of steps in each episode, $P_h : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ is the probability transition function which maps state-action-state pair to a probability, $R_h : \mathcal{S} \times \mathcal{A} \to [0, 1]$ represents the reward function in the step $h$. In this chapter, the probability transition functions and the reward functions can be different at different steps.

The agent interacts with the environment in a sequence of episodes. The total number of episodes is $K$. In each episode $k \in [K]$ of this MDP, the initial states $s_1$ is generated randomly by a distribution or chosen by the environment. Initial states may be different between episodes. At each step $h \in [H]$ of an episode, the agent observes the state $s_h$ and chooses an action $a_h$. After receiving the action, the environment generates a random reward $r_h \in [0, 1]$ derived from a distribution with mean $R_h(s_h, a_h)$ and next state $s_{h+1}$ which is drawn from the distribution $P_h(\cdot|s_h, a_h)$. $P_h(\cdot|s, a)$ represents the probability distribution over states if action $a$ is taken for state $s$. The agent stops interacting with environment after $H$ steps and starts another episode.

The policy $\pi$ of the agent is expressed as a mappings $\pi : \mathcal{S} \times [H] \times \mathcal{A} \to [0, 1]$. $\pi_h(a|s)$ represents the probability of taking action $a$ in state $s$ under stochastic policy $\pi$ at step $h$. We have that $\sum_{a\in\mathcal{A}} \pi_h(a|s) = 1$. A deterministic policy is a policy that maps each state to a particular action. For notation convenience, for a deterministic policy $\pi$, we use $\pi_h(s)$ to denote the action $a$ which satisfies $\pi_h(a|s) = 1$. Interacting with the environment $\mathcal{M}$, the policy induces a random trajectory $\{s_1, a_1, r_1, s_2, a_2, r_2, \cdots, s_H, a_H, r_H, s_{H+1}\}$.

We use $V_h^\pi : \mathcal{S} \to \mathbb{R}$ to denote the value function at step $h$ under policy $\pi$. Given a policy $\pi$ and step $h$, the value function of a state $s \in \mathcal{S}$ and the $Q$-function $Q_h^\pi : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ of a state-action pair $(s, a)$ are defined as: $V_h^\pi(s) = \mathbb{E}\left[\sum_{h'=h}^{H} r_{h'} | s_h = s, \pi\right]$ and $Q_h^\pi(s, a) = \mathbb{E}\left[\sum_{h'=h}^{H} r_{h'} | s_h = s, a_h = a, \pi\right]$, which represent the expected total rewards received from step $h$ to $H$, under policy $\pi$, starting from state $s$ and state-action pair $(s, a)$ respectively. It is well-known that the value function and $Q$-function satisfy the Bellman consistency equations. For notation simplicity, we denote $V_{H+1}^\pi = \mathbf{0}$, $Q_{H+1}^\pi = \mathbf{0}$ and $P_h V_{h+1}^\pi(s, a) = \mathbb{E}_{s'\sim P_h(\cdot|s,a)}[V_{h+1}^\pi(s')]$.

In this chapter, we assume that the state space $\mathcal{S}$ and action space $\mathcal{A}$ are finite sets, and the planning horizon $H$ is finite. Reward $r_h$ is bounded by $[0, 1]$, so the value function and $Q$-function are bounded. Under this case, there always exists an optimal policy $\pi^*$ such that $\pi^*$ maximize the value function and $Q$-function: $V_h^*(s) := V_h^{\pi^*}(s) = \sup_\pi V_h^\pi(s)$ and $Q_h^*(s, a) := Q_h^{\pi^*}(s, a) = \sup_\pi Q_h^\pi(s, a)$, for all $s$, $a$ and $h$. We measure the performance of the agent over $K$ episodes by the regret defined as:

$$\text{Regret}(K) = \sum_{k=1}^{K} [V_1^*(s_1^k) - V_1^{\pi^k}(s_1^k)], \tag{4.1}$$

where $s_1^k$ is the initial state and $\pi^k$ is the control policy followed by the agent for each episode $k$.

In this chapter, we introduce a novel adversary setting, in which the attacker sits between the agent and the environment. The attacker can monitor the state, the actions of the agent and the reward signals from the environment. Furthermore, the attacker can introduce action poisoning attacks on RL agent. In particular, at each episode $k$ and step $h$, after the agent chooses an action $a_h^k$, the attacker can change it to another action $\widetilde{a}_h^k \in \mathcal{A}$. If the attacker decides not to attack, $\widetilde{a}_h^k = a_h^k$. Then the environment receives $\widetilde{a}_h^k$, and generates a random reward $r_h^k$ with mean $R_h(s_h^k, \widetilde{a}_h^k)$ and the next state $s_{h+1}^k$ which is drawn from the distribution $P_h(\cdot|s_h^k, \widetilde{a}_h^k)$. The agent and attacker receive the reward $r_h^k$ and the next state $s_{h+1}^k$ from the environment. Note that the agent does not know the attacker's manipulations and the presence of the attacker and hence will still view $r_h^k$ as the reward and $s_{h+1}^k$ as the next state generated from state-action pair $(s_h^k, a_h^k)$.

The attacker has a target policy $\pi^\dagger$. We assume that the target policy $\pi^\dagger$ is a deterministic policy. The attacker's goal is to manipulate the agent into following the target policy $\pi^\dagger$ to pick its actions. We measure the performance of the attack over $K$ episodes by the total attack cost and the total number of the steps that the agent does not follow the target policy $\pi^\dagger$. By setting $\mathbb{1}(\cdot)$ as the indicator function, the attack cost function and the loss function are defined as

$$\text{Cost}(K,H) = \sum_{k=1}^{K}\sum_{h=1}^{H} \mathbb{1}\left(\widetilde{a}_h^k \neq a_h^k\right), \ \ \text{Loss}(K,H) = \sum_{k=1}^{K}\sum_{h=1}^{H} \mathbb{1}\left(a_h^k \neq \pi^\dagger(s_h^k)\right), \tag{4.2}$$

The attacker aims to minimize both the attack cost and the loss of attacks, or minimize one of them subject to a constraint on the one another. However, obtaining optimal solutions to these optimization problems is challenging. As the first step towards understanding the impact of action poisoning attacks, we design some specific simple yet effective attack strategies.

## 4.2 Attack Strategy and Analysis

In this chapter, we study the black-box action poisoning attack problem. In black-box attack case, the attacker has no prior knowledge about the underlying environment and the agent's policy. It only knows the observations, i.e., $s_h^k$, $a_h^k$, and $r_h^k$, generated when the agent interacts with the environment. This makes the attack practical as the attacker only needs to hijack the communication between the environment and the agent without stealing information from or attacking the agent and the environment. To build up intuitions about the proposed black-box action poisoning attack strategy, we first consider a white-box attack model, in which the attacker knows the underlying MDP and hence it is easier to design attack schemes. Building on insights obtained from the white-box attack schemes, we then introduce our proposed black-box attack strategy and analyze its performance.

### 4.2.1 White-box attack

In the white-box attack model, the attacker has full information of the underlying MDP $\mathcal{M}$. Thus, the attacker is able to calculate $V_h^*(s)$ and $Q_h^*(s,a)$ according to the Bellman optimality equations:

$$Q_h^*(s,a) = R_h(s,a) + P_h V_{h+1}^*(s,a), \;\; V_h^*(s) = \max_{a \in \mathcal{A}} Q_h^*(s,a). \tag{4.3}$$

Since $V_{H+1}^{\pi} = \mathbf{0}$ and $Q_{H+1}^{\pi} = \mathbf{0}$, $V_h^*(s)$ and $Q_h^*(s,a)$ can be obtained from the Bellman optimality equation. The optimal policy $\pi^*$ are derived from $\pi_h^*(s) = \arg\max_{a \in \mathcal{A}} Q_h^*(s,a)$.

With the knowledge of the optimal policy, the attacker can perform an intuitive attack: exchange the optimal action and the target action. In particular, at the step $h$ and state $s$, when the agent picks the optimal action $a = \pi_h^*(s)$, the attacker changes it to the action specified by the target policy $\widetilde{a} = \pi_h^{\dagger}(s)$. When the agent selects the target action $a = \pi_h^{\dagger}(s)$, the attacker changes it to the action might be taken under the optimal policy $\widetilde{a} = \pi_h^*(s)$. In addition, when the agent's action does not follow the optimal policy or the target policy, the attacker does not attack. We name this attack scheme as the exchange attack (E-attack) strategy. From the agent's point of view, $\pi^{\dagger}$ becomes the optimal strategy under the E-attack strategy, as the agent does not know the presence of the attacker. If the optimal policy is singular, any RL algorithm with sub-linear regret will learn to follow the optimal strategy in his observation, i.e. $\pi^{\dagger}$, with a sub-linear regret. As the result, the loss will be sub-linear. However, the cost of the E-attack strategy may be up to $\mathcal{O}(T)$, where $T = KH$ is the total number of steps. The main reason is that, in the E-attack strategy, the attacker needs to change the actions whenever the agent chooses an action specified by the target policy $\pi^{\dagger}$, which happens most of the time as the agent views $\pi^{\dagger}$ as the optimal policy. Furthermore, another drawback of the

E-attack is that the expected reward the agent receives is not impacted.

Even though the E-attack strategy discussed above could force the agent to follow the target policy $\pi^\dagger$, the cost is too high and it does not affect the agent's total expected rewards. In order to reduce the cost and have real impact on the agent's total expected rewards, the attacker should avoid to attack when the agent takes an action specified by $\pi^\dagger$. This is possible if $\pi^\dagger$ satisfies certain conditions to be specified in the sequel.

Before presenting the proposed attack strategy, we first discuss conditions under which such an attack is possible. If the target policy is the worst policy such that $V_h^{\pi^\dagger}(s) = \inf_\pi V_h^\pi(s)$, the attacker cannot force the agent to learn the target policy without attacking the target action. For notation simplicity, we denote $V_h^\dagger(s) := V_h^{\pi^\dagger}(s)$. The combination of the attacker and the environment can be considered as a new environment to the agent. As the action poisoning attack only changes the actions, it can impact but does not have direct control of the agent's observations. Although the action poisoning attack is widely applicable, the attacker's ability is weaker than the attacker in the environment poisoning attack model. It is reasonable to limit the choice of the target policy. In this chapter, we study a class of target policies denoted as $\Pi^\dagger$, for which each element $\pi^\dagger \in \Pi^\dagger$ satisfies

$$V_h^\dagger(s) > \min_{a\in\mathcal{A}} Q_h^\dagger(s,a), \tag{4.4}$$

for all state $s$ and all step $h$. That is $\pi^\dagger$ is not the worst policy.

**Assumption 2.** For the underlying MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, H, P, R)$, $\Pi^\dagger \neq \emptyset$, and the attacker's target policy $\pi^\dagger$ satisfies $\pi^\dagger \in \Pi^\dagger$.

For a given target policy $\pi^\dagger$, as $|\mathcal{S}|$, $|\mathcal{A}|$ and $H$ are finite, the minimum of $Q_h^\dagger(s,a)$ subject to $a \in \mathcal{A}$ exists for all step $h$ and state $s$. We define the minimum gap $\Delta_{min}$ by

$$\Delta_{min} = \min_{h\in[H],s\in\mathcal{S}} \left( V_h^\dagger(s) - \min_{a\in\mathcal{A}} Q_h^\dagger(s,a) \right). \tag{4.5}$$

Under Assumption 2, the minimum gap is positive, i.e. $\Delta_{min} > 0$. This positive gap provides a chance of efficient action poisoning attacks. All results in this chapter are based on Assumption 2.

We now introduce an effective white-box attack schemes: $\alpha$-portion attack. Specifically, at the step $h$ and state $s$, if the agent picks the target action, i.e., $a = \pi_h^\dagger(s)$, the attacker does not attack, i.e. $\widetilde{a} = a = \pi_h^\dagger(s)$. If the agent picks a non-target action, i.e., $a \neq \pi_h^\dagger(s)$, the $\alpha$-portion attack sets $\widetilde{a}$ as

$$\widetilde{a} = \begin{cases} \pi_h^\dagger(s), \text{with probability } 1-\alpha \\ \arg\min_{a\in\mathcal{A}} Q_h^\dagger(s,a), \text{with probability } \alpha. \end{cases} \tag{4.6}$$

For a given target policy $\pi^\dagger$, we define $\pi_h^-(s) = \arg\min_{a\in\mathcal{A}} Q_h^\dagger(s,a)$. We have the following result:

**Lemma 1.** If the attacker follows the $\alpha$-portion attack scheme on an RL agent, in the observation of the agent, the target policy $\pi^\dagger$ is the optimal policy.

The detailed proof can be found in the Appendix C.1.1. Using Lemma 1, we can derive upper bounds of the loss and the cost functions of the $\alpha$-portion attack scheme.

**Theorem 8.** Assume the expected regret Regret$(K)$ of the RL agent's algorithm is bounded by a sub-linear bound $\mathcal{R}(T)$, i.e., Regret$(K) \leq \mathcal{R}(T)$. The $\alpha$-portion attack will force the agent to learn the target policy $\pi^\dagger$ with the expected cost and the expected loss bounded by

$$\mathbb{E}[\text{Cost}(K,H)] \leq \mathbb{E}[\text{Loss}(K,H)] \leq \mathcal{R}(T)/(\alpha\Delta_{min}), \tag{4.7}$$

In addition, with probability $1-p$, the loss and the cost is bounded by

$$\text{Cost}(K,H) \leq \text{Loss}(K,H) \leq \left(\mathcal{R}(T) + 2H^2\sqrt{\log(1/p)\mathcal{R}(T)}\right)/(\alpha\Delta_{min}). \tag{4.8}$$

The detailed proofs can be found in Appendix C.1.2. In the white-box setting, the attacker can simply choose $\alpha = 1$ to most effectively attack RL agents. Intuitively speaking, if $\alpha = 1$, whenever the agent chooses a non-target action, the attacker changes it to the worst action under policy $\pi^\dagger$, so that all non-target actions become worse than the target action and the target policy becomes optimal in the observation of the agent.

### 4.2.2 Black-box attack

In the black-box attack setting, the attacker has no prior information about the underlying environment and the agent's algorithm, it only observes the samples generated when the agent interacts with the environment. Since the $\alpha$-portion attack described in (4.6) for the white-box setting relies on the information of the underlying environment to solve $\pi_h^-$, the $\alpha$-portion attack is not applicable in the black-box setting. However, by collecting the observations and evaluating the $Q$-function $Q_h^\dagger(s,a)$, the attacker can perform an attack to approximate the $\alpha$-portion attack. In the proposed attack scheme, the attacker evaluates the $Q$-values of the target policy $\pi^\dagger$ with an important sampling (IS) estimator. Then, the attacker calculates the lower confidence bound (LCB) on the $Q$-values so that he can explore and exploit the worst action by the LCB method. In this chapter, we use Hoeffding-type martingale concentration inequalities to build the confidence bound. Using this information, the attacker can then carry out an attack similar to the $\alpha$-portion attack. We name the proposed attack strategy as LCB-H attack. The algorithm is summarized in Algorithm 1. In the following, we will show that the LCB-H attack nearly matches the performance of the $\alpha$-portion attack.

Here, we highlight the main idea of the LCB-H attack. As discussed in Section 4.2.1, if the attacker has full information of the MDP and knows the worst action of any state $s$ at any step $h$ under policy $\pi^\dagger$, he can simply change the agent's non-target action to the worst action. However, in the black-box setting, the attacker does not know the worst actions under policy $\pi^\dagger$. One intuitive idea is to estimate $Q^\dagger$ and find the possible worst actions by the estimates of $Q^\dagger$. Once the attacker obtains esimate $\hat{Q}^\dagger$, it carries out an attack similar to the $\alpha$-portion attack by setting $\alpha = 1/H$ (the reason why we set $\alpha = 1/H$ will be discussed in the sequel): 1) when the agent picks a target action, the LCB-H attacker does not attack ; 2) when the agent picks a non-target action, with probability $1 - \frac{1}{H}$, the LCB-H attacker changes it to the target action, while with probability $\frac{1}{H}$, the LCB-H attacker changes it to the action that has the lowest lower confidence bound value. Here, we use the lower confidence bound value because in the black-box setting, the LCB-H attacker does not know which action is the worst, and hence uses confidence bounds to explore and exploit

**Algorithm 4.1** LCB-H attack strategy on RL algorithm

```
Require:
    Target policy $\pi^\dagger$.
Initialize $L_h(s,a) = -\infty$, $\hat{Q}_h^\dagger(s,a) = 0$, and $N_h(s,a) = 0$ for all state $s \in \mathcal{S}$, all action $a \in \mathcal{A}$ and all step $h \in [H]$.
for episode $k = 1, 2, \ldots, K$ do
    Receive $s_1^k$. Initialize the set of trajectory $traj = \{s_1^k\}$.
    for step $h = 1, 2, \ldots, H$ do
        The agent chooses an action $a_h^k$.
        if $a_h^k = \pi_h^\dagger(s_h^k)$ then
            The attacker does not attack, i.e. $\widetilde{a}_h^k = a_h^k$, and sets the IS weight $w_h = 1$.
        else
            $\widetilde{a}_h^k = \begin{cases} \arg\min_{a \neq \pi_h^\dagger(s_h^k)} L_h(s_h^k, a) \text{ and set } w_h = 0, & \text{with probability } 1/H, \\ \pi_h^\dagger(s_h^k) \text{ and set } w_h = H/(H-1), & \text{with probability } 1 - 1/H. \end{cases}$
        end if
        The environment receives action $\widetilde{a}_h^k$, and returns the reward $r_h^k$ and the next state $s_{h+1}^k$.
        Update the trajectory by plugging $\widetilde{a}_h^k$, $r_h^k$ and $s_{h+1}^k$ into $traj$.
    end for
    Set the cumulative reward $G_{H+1} = 0$ and the importance ratio $\rho_{H+1:H+1} = 1$.
    for step $h = H, H-1, \ldots, 1$ do
        $G_h = r_h^k + G_{h+1}$, $\rho_{h:H+1} = \rho_{h+1:H+1} \cdot w_h$, $t = N_h(s_h^k, \widetilde{a}_h^k) \leftarrow N_h(s_h^k, \widetilde{a}_h^k) + 1$.
        $\hat{Q}_h^\dagger(s_h^k, \widetilde{a}_h^k) \leftarrow (1 - \frac{1}{t})\hat{Q}_h^\dagger(s_h^k, \widetilde{a}_h^k) + \frac{1}{t}\left(r_h^k + G_{h+1} \cdot \rho_{h+1:H+1}\right)$.
        $L_h(s_h^k, \widetilde{a}_h^k) = \hat{Q}_h^\dagger(s_h^k, \widetilde{a}_h^k) - (e(H-h)+1)\sqrt{2\log(2SAT/p)/t}$.
    end for
end for
```

the worst action.

As shown in Algorithm 4.1, after collecting observations, the LCB-H attacker uses IS estimator to evaluate the target policy, which is an off-policy method [82, 98]. The IS estimator provides an unbiased estimate of the target policy $\pi^\dagger$. Suppose $\pi^k$ is the control policy followed by the agent at episode $k$ and the attacker applies LCB-H attack on the agent. From Algorithm 4.1, the probability of an action $\widetilde{a}$ chosen by the behavior policy $b_h^k$ at the state $s$ and step $h$ can be written as

$$\mathbb{P}(\widetilde{a}|s, b_h^k) = \begin{cases} 1 & \text{if } \widetilde{a} = a_h^k = \pi_h^\dagger(s), \\ 1/H & \text{if } \widetilde{a} = \underset{a \neq \pi_h^\dagger(s)}{\arg\min}\, L_h^k(s,a) \text{ and } a_h^k \neq \pi_h^\dagger(s), \\ 1 - 1/H & \text{if } \widetilde{a} = \pi_h^\dagger(s) \text{ and } a_h^k \neq \pi_h^\dagger(s), \\ 0 & \text{if otherwise.} \end{cases} \tag{4.9}$$

Then, the trajectory at episode $k$, $\{s_1^k, \widetilde{a}_1^k, r_1^k, s_2^k, \widetilde{a}_2^k, r_2^k, \cdots, s_H^k, \widetilde{a}_H^k, r_H^k, s_{H+1}^k\}$, is generated under the behavior policy $b^k$. Since we assume that the target policy is a deterministic function, we have

$$\mathbb{P}(\widetilde{a}|s, \pi_h^\dagger) = \mathbb{1}(\widetilde{a} = \pi_h^\dagger(s)). \tag{4.10}$$

The importance sampling ratio $\rho_{h:H}^k = \prod_{h'=h}^{H} \frac{\mathbb{P}(\widetilde{a}_{h'}^k|s_{h'}^k, \pi^\dagger)}{\mathbb{P}(\widetilde{a}_{h'}^k|s_{h'}^k, b^k)}$ can be computed using (4.9) and (4.10), which is also used in Algorithm 4.1. Define the cumulative reward as $G_{h:H}^k = \sum_{h'=h}^{H} r_h^k$. For notation simplicity, we set $\rho_{h:H+1}^k = \rho_{h:H}^k$ and $G_{h:H+1}^k = G_{h:H}^k$ when $1 \leq h \leq H$, and $\rho_{H+1:H+1}^k = 1$ and $G_{H+1:H+1}^k = 0$. Since the trajectory is generated by following the behavior policy $b^k$, we have that for all step $h$ with $1 \leq h \leq H+1$, $V_h^{b^k}(s) = \mathbb{E}[G_{h:H}^k | s_h^k = s]$ and $V_h^\dagger(s) = \mathbb{E}[\rho_{h:H}^k G_{h:H}^k | s_h^k = s] = \mathbb{E}[\rho_{h:H+1}^k G_{h:H+1}^k | s_h^k = s]$.

We here explain why we set $\alpha = 1/H$. The main reason is that the performance of the estimates of $Q^\dagger$ depends on the number of the observations that follow $\pi^\dagger$. Even though IS estimator provides an unbiased estimate, the variance of the IS might be very high. By choosing $\alpha = 1/H$, we can control the variance. In particular, to obtain estimate $\hat{Q}_h^\dagger$, by the Bellman consistency equations, we first estimate $V_{h+1}^\dagger$. By setting $\alpha = \frac{1}{H}$, we can show that the importance ratio $\rho_{h+1:H+1}^k = \rho_{h+1:H}^k \leq \frac{1}{(1-\alpha)^{H-h}} \leq \frac{1}{(1-\alpha)^{H-1}} \leq e$ when $H \geq 2$ and $1 \leq h \leq H-1$, and $\rho_{H+1:H+1}^k = 1$. As the result, $\rho_{h+1:H}^k G_{h+1:H}^k$ will be bounded, and the variance of the estimate of $V_{h+1}^\dagger$ can be controlled.

We build a confidence bound to show the performance of the estimate error of $Q^\dagger$. The confidence bound is built based on Hoeffding inequalities and shown in the following Lemma.

**Lemma 2.** If the attacker follows the LCB-H attack strategy on the RL agent, for any $p \in (0,1)$, with probability at least $1-p$, the following confidence bound of $\hat{Q}^{\dagger}_{h}$ holds simultaneously for all $(s,a,h,k) \in \mathcal{S} \times \mathcal{A} \times [H] \times [k]$:

$$\left|\hat{Q}^{\dagger}_{h,k}(s,a) - Q^{\dagger}_{h}(s,a)\right| \leq (e(H-h)+1)\sqrt{2\log(2SAT/p)/N^{k}_{h}(s,a)}, \tag{4.11}$$

where $\hat{Q}^{\dagger}_{h,k}(s,a)$ represents the attacker's evaluation of $Q$-values at the step $h$ at the beginning of the episode $k$, and $N^{k}_{h}(s,a)$ represents the cumulative number of attacker's state-action pair $(s,a)$ at the step $h$ until the beginning of the episode $k$, i.e. $N^{k}_{h}(s,a) = \sum_{k'=1}^{k-1} \mathbb{1}(s^{k'}_{h} = s)\mathbb{1}(\widetilde{a}^{k'}_{h} = a)$ .

The detailed proof can be found in Appendix C.2.1. In Lemma 2, the given bound on LCB-H attacker's estimation of the $Q$-values mainly based on $N^{k}_{h}(s,a)$ the number of state-action pair $(s,a)$ at the step $h$. This bound is similar to the confidence bound in the UCB algorithm for the bandit problem, except for the additional $H$ factor. Compared with the bandit problem, $Q$-values are the expected cumulative rewards, which bring the additional $H$ factor.

The LCB-H attack scheme uses a LCB method to explore and exploit the worst action. Thus, when the agent picks a non-target action, the LCB-H attacker changes it to different post-attack actions in different episodes. In the observation of the agent, the environment is non-stationary, i.e., the reward functions and probability transition function may change over episodes. Following the existing works on non-stationary RL [15,23,72], we define $V^{k,\pi}_{h}(s) = \mathbb{E}\left[\sum_{h'=h}^{H} r^{k}_{h'} | s^{k}_{h} = s, \pi\right]$ and define the expected dynamic regret for the agent as:

$$\text{D-Regret}(K) = \sum_{k=1}^{K}[V^{k,\pi^{k,*}}_{1}(s^{k}_{1}) - V^{k,\pi^{k}}_{1}(s^{k}_{1})], \tag{4.12}$$

where $\pi^{k,*}$ is the optimal policy at episode $k$, i.e. $V^{k,\pi^{k,*}}_{h}(s) = \sup_{\pi} V^{k,\pi}_{h}(s)$.

Here we state our main theorem, whose proof is deferred to Appendix C.2.2.

**Theorem 9.** Assume the expected dynamic regret of the RL agent's algorithm D-Regret$(K)$ is bounded by a sub-linear bound $\mathcal{R}(T)$, i.e., D-Regret$(K) \leq \mathcal{R}(T)$. With probability $1-4p$, the

LCB-H attack will force the agent to learn the target policy $\pi^\dagger$ with the cost and loss bounded by

$$\text{Cost}(K,H) \leq \text{Loss}(K,H) \leq \frac{H\left(\mathcal{R}(T) + 2H^2\sqrt{\log(1/p)\mathcal{R}(T)}\right)}{\Delta_{min}} + \frac{307SAH^4\log(2SAT/p)}{\Delta_{min}^2}.$$

From Theorem 9 we see that when $\mathcal{R}(T) \leq O(\frac{SAH^3\log(2SAT/p)}{\Delta_{min}})$, the cost and loss are bounded by $O(\frac{SAH^4\log(2SAT/p)}{\Delta_{min}^2})$, which scales as $\log(T)$, otherwise the cost and loss are bounded by $O(\mathcal{R}(T))$ that scales linearly with $\mathcal{R}(T)$. The LCB-H attack nearly matches the performance of the $\alpha$-portion attack, without requiring any information of the underlying environment and the agent's algorithm. Compared with the results of the $\alpha$-portion attack, the additional part of the bound in Theorem 9, i.e. $\frac{307SAH^4\log(2SAT/p)}{\Delta_{min}^2}$, is from the cost of exploring the worst action. Note that we use a LCB method to explore the worst action and the confidence bound is built by Hoeffding inequalities. The bound of the loss and cost can potentially be improved by using Bernstein-type concentration inequalities.

### 4.2.3 Black-box attack on UCB-H

In this section, we use UCB-H algorithm [41] as an example to illustrate the effects of the proposed LCB-H action poisoning attack strategy. UCB-H algorithm is a model-free $Q$-learning algorithm equipped with a UCB-Hoeffding exploration policy. At a high level, this algorithm builds an high-probability upper bound of $Q$-function for every state-action pairs. Then, it greedily chooses the action according to the optimistic estimations. In this section, we derive an upper bound of the loss and the cost of the LCB-H attack against UCB-H agent. Complete proofs of Theorem 10 and its supporting lemmas are provided in Appendix C.3.

**Theorem 10.** For any given target policy $\pi^\dagger \in \Pi^\dagger$, with probability $1-4p$, the LCB-H attacker can successfully manipulate the UCB-H algorithm to implement the target policy $\pi^\dagger$, with the cost and the loss bounded as follow:

$$\text{Cost}(K,H) \leq \text{Loss}(K) \leq O\left(H^5\log(2H/p) + \frac{1}{\Delta_{min}}SAH^4 + \frac{1}{\Delta_{min}^2}H^{10}SA\log(2SAT/p)\right).$$

Theorem 10 reveals a significant security threat of efficient RL agents. It shows that by spending only logarithm cost, the LCB-H attack is able to force UCB-H agent to choose actions specified by a policy decided by the attack with only logarithm loss.

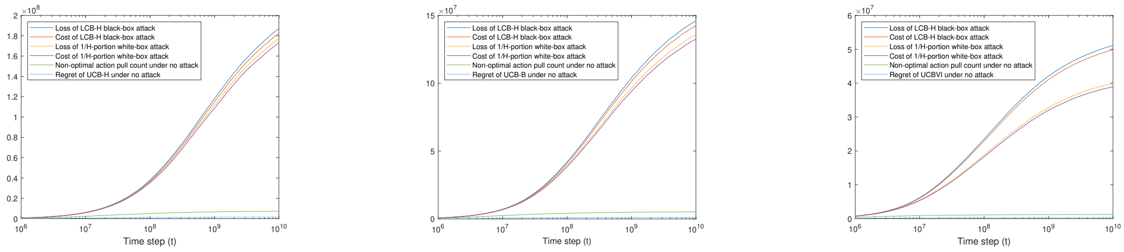
The results in Theorem 10 are consistent with the results in Theorem 9. In particular, [111] proved a gap-independence bound on UCB-H that scales as $O(\frac{H^6SA}{\Delta}\log(T))$, where $\Delta = \min_{h,s,a}\{V_h^*(s)-Q_h^*(s,a) : V_h^*(s)-Q_h^*(s,a) > 0\}$ is the sub-optimality gap. If an algorithm whose dynamic regret bound scales as $O(\frac{H^6SA}{\Delta}\log(T))$, the cost and loss are scale as $O(\frac{H^7SA}{\Delta^2}\log(T))$. UCB-H is a stationary RL algorithm, while the LCB-H adaptively attacks the agent and hence the effective environment observed by the agent is non-stationary. This adds a factor to the loss and cost.

## 4.3 Numerical Experiments

In this section, we empirically evaluate the performance of LCB-H attacks against three efficient RL agents, namely UCB-H [41], UCB-B [41] and UCBVI-CH [5], respectively.

### 4.3.1 1D grid world

We perform numerical simulations on an environment represented as an MDP with ten states and five actions, i.e. $S = 10$ and $A = 5$. The environment is a periodic 1-d grid world. The action space $\mathcal{A}$ is given by {two steps left, one step left, stay, one step right, two steps right}. For any given state-action pair $(s, a)$, with probability $p(s, a)$, the agent navigates by the action; with probability $1 - p(s, a)$, the agent's next state is sampled randomly from the five adjacent states (include itself). For example, if the environment receives state-action pair $(s, a) = (5, \text{stay})$, with probability $p(5, \text{stay})$, the next state is 5; with probability $\frac{1-p(5,\text{stay})}{5}$, the next state is 3, 4, 5, 6 or 7. By randomly generating $p(s, a)$ with $0.5 < p(s, a) < 1$, we randomly generate the transition probabilities $P(s'|s, a)$ for all action $a$ and state $s$. The mean reward of state-action pairs are randomly generated from a set of values $\{0.2, 0.35, 0.5, 0.65, 0.8\}$. In this chapter, we assume the rewards are bounded by $[0, 1]$. Thus, we use Bernoulli distribution to randomize the reward signal. The target policy is randomly chosen by deleting the worst action, so as to satisfy Assumption 2. We set the total number of steps $H = 10$ and the total number of episodes $K = 10^9$.



(a) Attack UCB-H

(b) Attack UCB-B

(c) Attack UCBVI-CH

Figure 4.1: Action poisoning attacks against RL agents

In Figure 4.1, we illustrate $\frac{1}{H}$-portion white-box attack and LCB-H black-box attack against three different agents separately and compare the loss and cost of these two attack schemes. For comparison purposes, we also add the curves for the regret of three agents under no attack. In the figure, the non-optimal action pull count are defined as $\sum_{k=1}^{K} \sum_{h=1}^{H} \mathbb{1}(Q_h^*(s_h^k, a_h^k) < V_h^*(s_h^k))$. The x-axis uses a base-10 logarithmic scale and represents the time step $t$ with the total time step $T = KH$. The y-axis represents the cumulative loss, cost and regret that change over time steps. The results show that, the loss and cost of $\frac{1}{H}$-portion white-box attack and LCB-H black-box attack scale as $\log(T)$. Furthermore the performances of our black-box attack scheme, LCB-H, nearly matches those of the $\frac{1}{H}$-portion white-box attack scheme. In addition, the cost and loss are about $H/\Delta_{min}$ times as much as the regret. This is consistent with our analysis in Theorem 9. Each of the individual experimental runs costs about twenty hours on one physical CPU core. The type of CPU is Intel Core i7-8700.

### 4.3.2 2D grid world

In this section, we introduce some additional numerical experiments. We perform numerical simulations on an environment represented as an MDP with $12$ states and $4$ actions, i.e. $S = 12$ and

$A = 4$. The environment is a 4-by-4 grid world. The action space $\mathcal{A}$ is given by $\{$North = 1, South = 2, West = 3, East = 4$\}$. The terminal state is at cell $[4, 4]$ (blue cell). If the agent at the terminal state and chooses any actions, the next state will be the beginning state at cell $[1, 1]$ and the agent receives reward $+1$. The agent is blocked by obstacles in cells $[2, 2]$, $[2, 3]$, $[2, 4]$ and $[3, 2]$ (black cells). The environment contains a special jump from cell $[1, 3]$ to cell $[3, 3]$ with $+1$ reward. When the agent at the cell $[1, 3]$ and chooses action ”South”, the agent will jump to the cell $[3, 3]$. Actions that would take the agent off the grid leave its location unchanged.

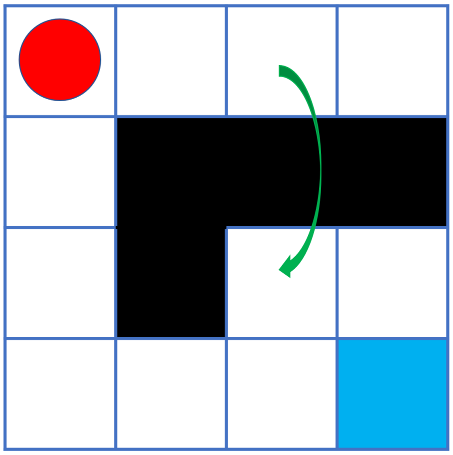
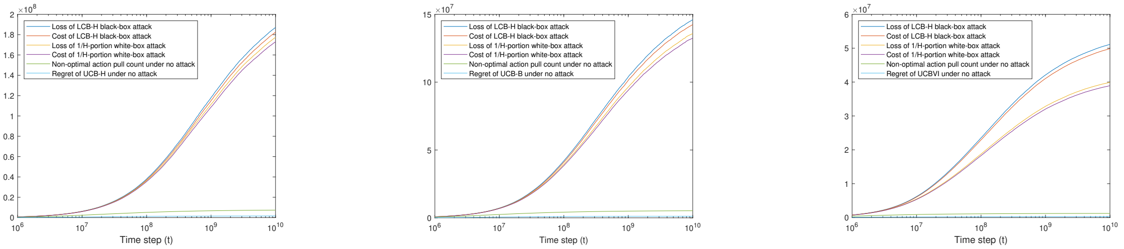


Figure 4.2: 2-d grid world

To add randomness to the environment, we set the states transit randomly: after the environment receives the action signal, the next state may generated by following the action with probability 0.7 and any of the other three actions with probability $0.1$ separately. For example, if the agent is at cell $[4, 3]$ and chooses action ”North”, the next state will be $[3, 3]$ with probability $0.7$, $[4, 2]$ with probability $0.1$, $[4, 3]$ with probability $0.1$, or $[4, 4]$ with probability $0.1$. The rewards of actions that would take the agent off the grid or towards the obstacle are $0$. The rewards of other state-action pairs are $0.2$ or $0.4$. In this Chapter, we assume the rewards are bounded by $[0, 1]$. Thus, we use Bernoulli distribution to randomize the reward signal. The optimal policy encourages the agent to take the special jump and reach the terminal state. In the target policy, the agent will reach the terminal state as soon as possible but avoid to take the special jump. We set the total number of steps $H = 10$ and the total number of episodes $K = 10^9$. We empirically evaluate the performance of LCB-H attacks against three efficient RL agents, namely UCB-H [41], UCB-B [41] and UCBVI-CH [5], respectively.



(a) Attack UCB-H

(b) Attack UCB-B

(c) Attack UCBVI-CH

Figure 4.3: Action poisoning attacks against RL agents

In Figure 4.3, we illustrate $\frac{1}{H}$-portion white-box attack and LCB-H black-box attack against three different agents separately and compare the loss and cost of these two attack schemes. For comparison purposes, we also add the curves for the regret of three agents under no attack. The x-axis uses a base-10 logarithmic scale and represents the time step $t$ with the total time step $T = KH$. Similar to the results in Figure 4.1, the results in 4.3 show that the loss and cost of $\frac{1}{H}$-portion white-box attack and LCB-H black-box attack scale as $\log(T)$. Furthermore the performances of our black-box attack scheme, LCB-H, nearly matches those of the $\frac{1}{H}$-portion white-box attack scheme.

## 4.4 Limitations

Here we highlight the assumptions and limitations of our work. Our theoretical results rely on Assumption 2 which limits the choice of the target policy. A violation of Assumption 2 may cause linear cost or loss of the proposed attack scheme. In Theorem 9, we assume the expected dynamic regret of the RL agent is bounded. Generally, the expected dynamic regret is a stronger notation than the dynamic regret. In other words, the optimal policy $\pi^{k,*}$ in (4.12) may change in each episode $k$, while the optimal policy $\pi^k$ in (4.1) is fixed over episodes. In this chapter, we discuss the action poisoning attack in the tabular episodic MDP context. Although we are convinced that the idea of our proposed attack scheme can be carried over to RL with function approximation, the current results only apply to the tabular episodic MDP setting.

## 4.5 Conclusions

In this chapter, we have introduced a new class of attacks on RL: action poisoning attacks. We have proposed the $\alpha$-portion white-box attack and the LCB-H black-box attack. We have shown that the $\alpha$-portion white-box attack is able to attack any efficient RL agent and the LCB-H black-box attack nearly matches the performance of the $\alpha$-portion attack. We have analyzed the LCB-H attack against the UCB-H algorithm and proved that the proposed attack scheme can force the agent to almost always follow a particular class of target policy with only logarithm loss and cost.

# Chapter 5

# Adversarial Attacks on Multi-agent Reinforcement Learning

Building on insights obtained in previous chapters, we now investigate the impact of adversarial attacks on MARL. In the considered setup, there is an exogenous attacker who is able to modify the rewards before the agents receive them or manipulate the actions before the environment receives them. The attacker aims to guide each agent into a target policy or maximize the cumulative rewards under some specific reward function chosen by the attacker, while minimizing the amount of manipulation on feedback and action. In Section 5.1, we describe the problem setup. In Section 5.2, we show hat the effectiveness of action poisoning only attacks and reward poisoning only attacks is limited. In Section 5.3, we introduce a mixed attack strategy in the gray-box setting which can can force any sub-linear-regret MARL agents to choose actions according to the target policy specified by the attacker with sub-linear cost and sub-linear loss. In Section 5.4, we an approximate mixed attack strategy in the black-box setting, and investigate the impact of the approximate mixed attack strategy attack on V-learning [42]. In Section 5.5, we conduct numerical experiments to validate the analysis of our attack strategies. The proofs are collected in Appendix D

## 5.1 Problem Setup

### 5.1.1 Definitions

We first introduce some standard definitions related to MARL that will be used throughout of this chapter. These definitions mostly follow those defined in [42]. We denote a tabular episodic MG with $m$ agents by a tuple MG$(\mathcal{S}, \{\mathcal{A}_i\}_{i=1}^m, H, P, \{R_i\}_{i=1}^m)$, where $\mathcal{S}$ is the state space with $|\mathcal{S}| = S$, $\mathcal{A}_i$ is the action space for the $i^{\text{th}}$ agent with $|\mathcal{A}_i| = A_i$, $H \in \mathbb{Z}^+$ is the number of steps in each episode. We let $\boldsymbol{a} := (a_1, \cdots, a_m)$ denote the joint action of all the $m$ agents and $\mathcal{A} := \mathcal{A}_i \times \cdots \times \mathcal{A}_m$ denote the joint action space. $P = \{P_h\}_{h \in [H]}$ is a collection of transition matrices. $P_h : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ is the probability transition function that maps state-action-state pair to a probability, $R_{i,h} : \mathcal{S} \times \mathcal{A} \to [0, 1]$ represents the reward function for the $i^{\text{th}}$ agent in the step $h$. The probability transition functions and the reward functions can be different at different steps. We note that this MG model incorporates both cooperation and competition because the reward functions of different agents can be arbitrary.

**Interaction protocol:** The agents interact with the environment in a sequence of episodes. The total number of episodes is $K$. In each episode $k \in [K]$ of MG, the initial states $s_1$ is generated randomly by a distribution $P_0(\cdot)$. Initial states may be different between episodes. At each step $h \in [H]$ of an episode, each agent $i$ observes the state $s_h$ and chooses an action $a_{i,h}$ simultaneously. After receiving the action, the environment generates a random reward $r_{i,h} \in [0, 1]$ for each agent $i$ derived from a distribution with mean $R_{i,h}(s_h, \boldsymbol{a}_h)$, and transits to the next state $s_{h+1}$ drawn from the distribution $P_h(\cdot|s_h, \boldsymbol{a}_h)$. $P_h(\cdot|s, \boldsymbol{a})$ represents the probability distribution over states if joint action $\boldsymbol{a}$ is taken for state $s$. The agent stops interacting with environment after $H$ steps and starts another episode. At each time step, the agents may observe the actions played by other agents.

**Policy and value function:** A Markov policy takes actions only based on the current state. The policy $\pi_{i,h}$ of agent $i$ at step $h$ is expressed as a mappings $\pi_{i,h} : \mathcal{S} \to \Delta_{\mathcal{A}_i}$. $\pi_{i,h}(a_i|s)$ represents the probability of agent $i$ taking action $a_i$ in state $s$ under policy $\pi_i$ at step $h$. A deterministic policy is a policy that maps each state to a particular action. For notation convenience, for a deterministic policy $\pi_i$, we use $\pi_{i,h}(s)$ to denote the action $a_i$ which satisfies $\pi_{i,h}(a_i|s) = 1$. We denote the product policy of all the agents as $\pi := \pi_1 \times \cdots \times \pi_m$. We also denote $\pi_{-i} := \pi_1 \times \cdots \times \pi_{i-1} \times \pi_{i+1} \times \cdots \times \pi_m$ to be the product policy excluding agent $i$. If every agent follows a deterministic policy, the product policy of all the agents is also deterministic. We use $V_{i,h}^{\pi} : \mathcal{S} \to \mathbb{R}$ to denote the value function of agent $i$ at step $h$ under policy $\pi$ and define $V_{i,h}^{\pi}(s) := \mathbb{E}\left[\sum_{h'=h}^{H} r_{i,h'} | s_h = s, \pi\right]$. Given a policy $\pi$ and step $h$, the $i^{\text{th}}$ agent's $Q$-function $Q_{i,h}^{\pi} : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ of a state-action pair $(s, \boldsymbol{a})$ is defined as: $Q_{i,h}^{\pi}(s, \boldsymbol{a}) = \mathbb{E}\left[\sum_{h'=h}^{H} r_{i,h'} | s_h = s, \boldsymbol{a}_h = \boldsymbol{a}, \pi\right]$.

**Best response:** For any policy $\pi_{-i}$, there exists a best response of agent $i$, which is a policy that achieves the highest cumulative reward for itself if all other agents follow policy $\pi_{-i}$. We define the best response of agent $i$ towards policy $\pi_{-i}$ as $\mu^{\dagger}(\pi_{-i})$, which satisfies $\mu^{\dagger}(\pi_{-i}) := \arg\max_{\pi_i} V_{i,h}^{\pi_i \times \pi_{-i}}(s)$ for any state $s$ and any step $h$. We denote $\max_{\pi_i} V_{i,h}^{\pi_i \times \pi_{-i}}(s)$ as $V_{i,h}^{\dagger, \pi_{-i}}(s)$ for notation simplicity. By its definition, we know that the best response can always be achieved by a deterministic policy.

Nash Equilibrium (NE) is defined as a product policy where no agent can improve his own cumulative reward by unilaterally changing his strategy.

**Nash Equilibrium (NE) [42]:** A product policy $\pi$ is a NE if for all initial state $s$, $\max_{i \in [m]}(V_{i,1}^{\dagger, \pi_{-i}}(s) - V_{i,1}^{\pi}(s)) = 0$ holds. A product policy $\pi$ is an $\epsilon$-approximate Nash Equilibrium if for all initial state $s$, $\max_{i \in [m]}(V_{i,1}^{\dagger, \pi_{-i}}(s) - V_{i,1}^{\pi}(s)) \leq \epsilon$ holds.

**General correlated policy:** A general Markov correlated policy $\pi$ is a set of $H$ mappings $\pi := \{\pi_h : \Omega \times \mathcal{S} \to \Delta_{\mathcal{A}}\}_{h \in [H]}$. The first argument of $\pi_h$ is a random variable $\omega \in \Omega$ sampled from some underlying distributions. For any correlated policy $\pi = \{\pi_h\}_{h \in [H]}$ and any agent $i$, we can define a marginal policy $\pi_{-i}$ as a set of $H$ maps $\pi_i = \{\pi_{h,-i} : \Omega \times \mathcal{S} \to \Delta_{\mathcal{A}_{-i}}\}_{h \in [H]}$, where $\mathcal{A}_{-i} = \mathcal{A}_1 \times \cdots \times \mathcal{A}_{i-1} \times \mathcal{A}_{i+1} \times \cdots \times \mathcal{A}_m$. It is easy to verify that a deterministic joint policy is a product policy. The best response value of agent $i$ towards policy $\pi_{-i}$ as $\mu^{\dagger}(\pi_{-i})$, which satisfies $\mu^{\dagger}(\pi_{-i}) := \arg\max_{\pi_i} V_{i,h}^{\pi_i \times \pi_{-i}}(s)$ for any state $s$ and any step $h$.

**Coarse Correlated Equilibrium (CCE)[42]:** A correlated policy $\pi$ is an CCE if for all initial state $s$, $\max_{i \in [m]}(V_{i,1}^{\dagger, \pi_{-i}}(s) - V_{i,1}^{\pi}(s)) = 0$ holds. A correlated policy $\pi$ is an $\epsilon$-approximate CCE if for all initial state $s$, $\max_{i\in[m]}(V_{i,1}^{\dagger,\pi_{-i}}(s) - V_{i,1}^{\pi}(s)) \leq \epsilon$ holds.

**Strategy modification:** A strategy modification $\phi_i$ for agent $i$ is a set of mappings $\phi_i := \{(\mathcal{S}\times\mathcal{A})^{h-1}\times\mathcal{S}\times\mathcal{A}_i \to \mathcal{A}_i\}_{h\in[H]}$. For any policy $\pi_i$, the modified policy (denoted as $\phi_i \diamond \pi_i$) changes the action $\pi_{i,h}(\omega, s)$ under random sample $\omega$ and state $s$ to $\phi_i((s_1, \boldsymbol{a}_1, \ldots, s_h, a_{i,h}), \pi_{i,h}(\omega, s))$. For any joint policy $\pi$, we define the best strategy modification of agent $i$ as the maximizer of $\max_{\phi_i} V_{i,1}^{(\phi_i\diamond\pi_i)\odot\pi_{-i}}(s)$ for any initial state $s$.

**Correlated Equilibrium (CE)[42]:** A correlated policy $\pi$ is an CE if for all initial state $s$, $\max_{i\in[m]}\max_{\phi_i}(V_{i,1}^{(\phi_i\diamond\pi_i)\odot\pi_{-i}}(s) - V_{i,1}^{\pi}(s)) = 0$. A correlated policy $\pi$ is an $\epsilon$-approximate CE if for all initial state $s$, $\max_{i\in[m]}\max_{\phi_i}(V_{i,1}^{(\phi_i\diamond\pi_i)\odot\pi_{-i}}(s) - V_{i,1}^{\pi}(s)) \leq \epsilon$ holds.

In Markov games, it is known that an NE is an CE, and an CE is an CCE.

**Best-in-hindsight Regret:** Let $\pi^k$ denote the product policy deployed by the agents for each episode $k$. After $K$ episodes, the best-in-hindsight regret of agent $i$ is defined as $\text{Reg}_i(K, H) = \max_{\pi_i'}\sum_{k=1}^{K}[V_{i,1}^{\pi_i',\pi_{-i}^k}(s_1^k) - V_{i,1}^{\pi^k}(s_1^k)]$.

### 5.1.2 Poisoning attack setting

We are now ready to introduce the considered poisoning attack setting, in which there is an attacker sits between the agents and the environment. The attacker can monitor the states, the actions of the agents and the reward signals from the environment. Furthermore, the attacker can override actions and observations of agents. In particular, at each episode $k$ and step $h$, after each agent $i$ chooses an action $a_{i,h}^k$, the attacker may change it to another action $\widetilde{a}_{i,h}^k \in \mathcal{A}_i$. If the attacker does not override the actions, then $\widetilde{a}_{i,h}^k = a_i$. When the environment receives $\widetilde{\boldsymbol{a}}_h^k$, it generates random rewards $r_{i,h}^k$ with mean $R_{i,h}(s_h^k, \widetilde{\boldsymbol{a}}_h^k)$ for each agent $i$ and the next state $s_{h+1}^k$ is drawn from the distribution $P_h(\cdot|s_h^k, \widetilde{\boldsymbol{a}}_h^k)$. Before each agent $i$ receives the reward $r_{i,h}^k$, the attacker may change it to another reward $\widetilde{r}_{i,h}^k$. Agent $i$ receives the reward $\widetilde{r}_{i,h}^k$ and the next state $s_{h+1}^k$ from the environment. Note that agent $i$ does not know the attacker's manipulations and the presence of the attacker and hence will still view $\widetilde{r}_{i,h}^k$ as the reward and $s_{h+1}^k$ as the next state generated from state-action pair $(s_h^k, \boldsymbol{a}_h^k)$.

We call an attack as *action poisoning only attack*, if the attacker only overrides the action but not the rewards. We call an attack as *reward poisoning only attack* if the attacker only overrides the rewards but not the actions. In addition, we call an attack as *mixed attack* if the attack can carry out both action poisoning and reward poisoning attacks simultaneously.

The goal of the MARL learners is to learn an equilibrium. On the other hand, the attacker's goal is to either force the agents to learn a target policy $\pi^\dagger$ of the attacker's choice or to force the agents to learn a policy that maximizes the cumulative rewards under a specific reward function $R_{\dagger,h} : \mathcal{S} \times \mathcal{A} \to (0,1]$ chosen by the attacker. We note that this setup is very general. Different choices of $\pi^\dagger$ or $R_{\dagger,h}$ could lead to different objectives. For example, if the attacker aims to reduce the benefit of the agent $i$, the attacker's reward function $R_{\dagger,h}$ can be set to $1 - R_{i,h}$, or choose a target policy $\pi^\dagger$ that is detrimental to the agent $i$'s reward. If the attacker aims to maximize the total rewards of a subset of agents $\mathcal{C}$, the attacker's reward function $R_{\dagger,h}$ can be set to $\sum_{i\in\mathcal{C}} R_{i,h}$, or choose a target policy $\pi^\dagger = \arg\max \sum_{i\in\mathcal{C}} V_{i,1}^{\pi}(s_1)$ that maximizes the total rewards of agents in $\mathcal{C}$. We assume that the target policy $\pi^\dagger$ is deterministic and $R_{i,h}(s, \pi^\dagger(s)) > 0$. We measure the performance of the attack over $K$ episodes by the total attack cost and the attack loss. Set $\mathbb{1}(\cdot)$ as the indicator function. The attack cost over $K$ episodes is defined as $\text{Cost}(K,H) = \sum_{k=1}^{K} \sum_{h=1}^{H} \sum_{i=1}^{m} \left( \mathbb{1}(\widetilde{a}_{i,h}^k \neq a_{i,h}^k) + |\widetilde{r}_{i,h}^k - r_{i,h}^k| \right)$.

There are two different forms of attack loss based on the different goals of the attacker.

If the attacker's goal is to force the agents to learn a target policy $\pi^\dagger$, the attack loss over $K$ episodes is defined as $\text{Loss1}(K,H) = \sum_{k=1}^{K} \sum_{h=1}^{H} \sum_{i=1}^{m} \mathbb{1}\left( a_{i,h}^k \neq \pi_{i,h}^\dagger(s_{i,h}^k) \right)$.

If the attacker's goal is to force the agents to maximize the cumulative rewards under some specific reward function $R_\dagger$ chosen by the attacker, the attack loss over $K$ episodes is defined as $\text{Loss2}(K,H) = \sum_{k=1}^{K} [V_{\dagger,1}^{\pi^*}(s_1^k) - V_{\dagger,1}^{\pi^k}(s_1^k)]$. Here, $V_{\dagger,1}^{\pi}(s)$ is the expected cumulative rewards in state $s$ based on the attacker's reward function $R_\dagger$ under product policy $\pi$ and $V_{\dagger,1}^{\pi^*}(s) = \max_\pi V_{\dagger,1}^{\pi}(s)$. $\pi^k$ denote the product policy deployed by the agents for each episode $k$. $\pi^*$ is the optimal policy that maximizes the attacker's cumulative rewards. We have $\text{Loss2}(K,H) \leq H * \text{Loss1}(K,H)$.

Denote the total number of steps as $T = KH$. In the proposed poisoning attack problem, we call an attack strategy *successful* if the attack loss of the strategy scales as $o(T)$. Furthermore, we call an attack strategy *efficient and successful* if both the attack cost and attack loss scale as $o(T)$.

The attacker aims to minimize both the attack cost and the attack loss, or minimize one of them subject to a constraint on the other. However, obtaining optimal solutions to these optimization problems is challenging. As the first step towards understanding the attack problem, we show the limitations of the action poisoning only or the reward poisoning only attacks and then propose a simple mixed attack strategy that is efficient and successful.

Depending on the capability of the attacker, we consider three settings: the white-box, the gray-box and the black-box settings. The table below summarizes the differences among these settings.

Table 5.1: Differences of the white/gray/black-box attackers

| | white-box attacker | gray-box attacker | black-box attacker |
|---|---|---|---|
| MG | Has full information | No information | No information |
| $\pi^\dagger$ | Can be calculated if $R_\dagger$ given | Required and given | Not given |
| $R_\dagger$ | Not required if $\pi^\dagger$ given | Not required if $\pi^\dagger$ given | Required and given |
| Loss1 | Suitable by specify $\pi^\dagger$ | Suitable | Not suitable |
| Loss2 | Suitable if $R_\dagger$ given | Suitable if $R_\dagger$ given | Suitable |

## 5.2 White-box Attack Strategy and Analysis

In this section, to obtain insights to the problem, we consider the white-box model, in which the attacker has full information of the underlying MG $(\mathcal{S}, \{\mathcal{A}_i\}_{i=1}^m, H, P, \{R_i\}_{i=1}^m)$. Even in the white-box attack model, we show that there exist some environments where the attacker's goal cannot be achieved by reward poisoning only attacks or action poisoning only attacks in Section 5.2.1. Then, in Section 5.2.2 and Section 5.2.3, we provide some sufficient conditions under which the action poisoning attacks alone or the reward poisoning attacks alone can efficiently attack MARL algorithms. Under such conditions, we then introduce an efficient action poisoning attack strategy and an efficient reward poisoning attack strategy.

### 5.2.1 The limitations of the action poisoning attacks and the reward poisoning attacks

As discussed in Section 5.1, the attacker aims to force the agents to either follow the target policy $\pi^\dagger$ or to maximize the cumulative rewards under attacker's reward function $R_\dagger$. In the white-box poisoning attack model, these two goals are equivalent as the optimal policy $\pi^*$ on the attacker's reward function $R_\dagger$ can be calculated by the Bellman optimality equations. To maximize the cumulative rewards under attacker's reward function $R_\dagger$ is equivalent to force the agents follow the policy $\pi^\dagger = \pi^*$.

Existing MARL algorithms [42,63] can learn an $\epsilon$-approximate {NE, CE, CCE} with $\widetilde{\mathcal{O}}(1/\epsilon^2)$ sample complexities. To force the MARL agents to follow the policy $\pi^\dagger$, the attacker first needs to attack the agents such that the target policy $\pi^\dagger$ is the unique NE in the observation of the agents. However, this alone is not enough to force the MARL agents to follow the policy $\pi^\dagger$. Any other distinct policy should not be an $\epsilon$-approximate CCE. The reason is that, if there exists an $\epsilon$-approximate CCE $\pi$ such that $\pi(\pi^\dagger(s)|s) = 0$ for any state $s$, the agents, using existing MARL algorithms, may learn and then follow $\pi$, which will lead the attack loss to be $\mathcal{O}(T) = \mathcal{O}(KH)$. Hence, we need to ensure that any $\epsilon$-approximate CCE stays in the neighborhood of the target policy. This requirement is equivalent to achieve the following objective: for all $s \in \mathcal{S}$, and policy $\pi$,

$$\begin{aligned}
&\max_{i\in[m]}(\widetilde{V}_{i,1}^{\dagger,\pi^\dagger_{-i}}(s) - \widetilde{V}_{i,1}^{\pi^\dagger}(s)) = 0;\\
&\text{if } \pi \text{ is a product policy and } \pi \neq \pi^\dagger, \text{then } \max_{i\in[m]}(\widetilde{V}_{i,1}^{\dagger,\pi_{-i}}(s) - \widetilde{V}_{i,1}^{\pi}(s)) > 0;\\
&\text{if } \pi(\pi^\dagger(s')|s') = 0 \text{ for all } s', \text{then } \max_{i\in[m]}(\widetilde{V}_{i,1}^{\dagger,\pi_{-i}}(s) - \widetilde{V}_{i,1}^{\pi}(s)) > \epsilon,
\end{aligned} \tag{5.1}$$

where $\widetilde{V}$ is the expected reward based on the post-attack environments.

We now investigate whether there exist efficient and successful attack strategies that use action poisoning alone or reward poisoning alone. We first show that the power of action poisoning attack alone is limited.

**Theorem 11.** There exists a target policy $\pi^\dagger$ and a MG $(\mathcal{S}, \{\mathcal{A}_i\}_{i=1}^m, H, P, \{R_i\}_{i=1}^m)$ such that no action poisoning Markov attack strategy alone can efficiently and successfully attack MARL agents by achieving the objective in (5.1).

We now focus on strategies that use only reward poisoning. If the post-attack mean reward $\widetilde{R}$ is unbounded and the attacker can arbitrarily manipulate the rewards, there always exists an efficient and successful poisoning attack strategy. For example, the attacker can change the rewards of non-target actions to $-H$. However, such attacks can be easily detected, as the boundary of post-attack mean reward is distinct from the boundary of pre-attack mean reward. The following theorem shows that if the post-attack mean reward has the same boundary conditions as the pre-attack mean reward, the power of reward poisoning only attack is limited.

**Theorem 12.** If we limit the post-attack mean reward $\widetilde{R}$ to have the same boundary condition as that of the pre-attack mean reward $R$, i.e. $\widetilde{R} \in [0, 1]$, there exists a MG $(\mathcal{S}, \{\mathcal{A}_i\}_{i=1}^m, H, P, \{R_i\}_{i=1}^m)$ and a target policy $\pi^\dagger$ such that no reward poisoning Markov attack strategy alone can efficiently and successfully attack MARL agents by achieving the objective in (5.1).

The proofs of Theorem 11 and Theorem 12 are provided in Appendix D.2. The main idea of the proofs is as follows. In successful poisoning attacks, the attack loss scales as $o(T)$ so that the agents will follow the target policy $\pi^\dagger$ in at least $T - o(T)$ times. To efficiently attack the MARL agents, the attacker should avoid to attack when the agents follow the target policy. Otherwise, the poisoning attack cost will grow linearly with $T$. The proofs of Theorem 11 and Theorem 12 proceed by constructing an MG and a target policy $\pi^\dagger$ where the expected rewards under $\pi^\dagger$ is always the worst for some agents if the attacker avoids to attack when the agents follow the target policy.

### 5.2.2 White-box action poisoning attacks

Even though Section 5.2.1 shows that there exists MG and target policy such that the action poisoning only attacks cannot be efficiently successful, here we show that it can be efficient and successful for a class of target policies. The following condition characterizes such class of target policies.

**Condition 1:** *For the underlying environment MG* $(\mathcal{S}, \{\mathcal{A}_i\}_{i=1}^m, H, P, \{R_i\}_{i=1}^m)$*, the attacker's target policy* $\pi^\dagger$ *satisfies that for any state* $s$ *and any step* $h$*, there exists an action* $\boldsymbol{a}$ *such that* $V_{i,h}^{\pi^\dagger}(s) > Q_{i,h}^{\pi^\dagger}(s, \boldsymbol{a})$*, for any agent* $i$.

Under Condition 1, we can find a worse policy $\pi^-$ by

$$\pi_h^-(s) = \arg\max_{\boldsymbol{a}\in\mathcal{A}} \min_{i\in[m]} \left(V_{i,h}^{\pi^\dagger}(s) - Q_{i,h}^{\pi^\dagger}(s, \boldsymbol{a})\right) \; s.t. \forall i \in [m], V_{i,h}^{\pi^\dagger}(s) > Q_{i,h}^{\pi^\dagger}(s, \boldsymbol{a}). \tag{5.2}$$

Under this condition, we now introduce an effective white-box action attack strategies: $d$-portion attack. Specifically, at the step $h$ and state $s$, if all agents pick the target action, i.e., $\boldsymbol{a} = \pi_h^\dagger(s)$, the attacker does not attack, i.e. $\widetilde{\boldsymbol{a}} = \boldsymbol{a} = \pi_h^\dagger(s)$. If some agents pick a non-target action, i.e., $\boldsymbol{a} \neq \pi_h^\dagger(s)$, the $d$-portion attack sets $\widetilde{\boldsymbol{a}}$ as

$$\widetilde{\boldsymbol{a}} = \begin{cases} \pi_h^\dagger(s), \text{with probability } d_h(s, \boldsymbol{a})/m \\ \pi_h^-(s), \text{with probability } 1 - d_h(s, \boldsymbol{a})/m, \end{cases} \tag{5.3}$$

where $d_h(s, \boldsymbol{a}) = m/2 + \sum_{i=1}^m \mathbb{1}(a_i = \pi_{i,h}^\dagger(s))/2$.

**Theorem 13.** If the attacker follows the $d$-portion attack strategy on the MG agents, the best response of each agent $i$ towards the target policy $\pi_{-i}^\dagger$ is $\pi_i^\dagger$. The target policy $\pi^\dagger$ is an {NE, CE, CCE} from any agent's point of view. If every state $s \in \mathcal{S}$ is reachable at every step $h \in [H]$ under the target policy, $\pi^\dagger$ is the unique {NE, CE, CCE}.

The detailed proof can be found in Appendix D.3.1. Theorem 13 shows that the target policy $\pi^\dagger$ is the unique {NE, CE, CCE} under the $d$-portion attack. Thus, if the agents follow an MARL algorithm that is able to learn an $\epsilon$-approximate {NE, CE, CCE}, the agents will learn a policy approximate to the target policy. We now discuss the high-level idea why the $d$-portion attack works. Under Condition 1, $\pi^-$ is worse than the target policy $\pi^\dagger$ at the step $H$ from every agent's point of view. Thus, under the $d$-portion attack, the target action strictly dominates any other action at the step $H$, and $\pi^\dagger$ is the unique {NE, CE, CCE} at the step $H$. From induction on $h = H, H-1, \cdots, 1$, we can further prove that the $\pi^\dagger$ is the unique {NE, CE, CCE} at any step $h$. We define $\Delta_{i,h}^{\dagger -}(s) = Q_{i,h}^{\pi^\dagger}(s, \pi_h^\dagger(s)) - Q_{i,h}^{\pi^\dagger}(s, \pi_h^-(s))$ and the minimum gap $\Delta_{min} = \min_{h\in[H], s\in\mathcal{S}, i\in[m]} = \Delta_{i,h}^{\dagger -}(s)$. In addition, any other distinct policy is not an $\epsilon$-approximate CCE with different gap $\epsilon < \Delta_{min}/2$. We can derive upper bounds of the attack loss and the attack cost when attacking some special MARL algorithms.

**Theorem 14.** If the best-in-hindsight regret $\text{Reg}(K, H)$ of each agent's algorithm is bounded by a sub-linear bound $\mathcal{R}(T)$ for any MG in the absence of attack, and $\min_{s\in\mathcal{S}, i\in[m]} \Delta_{i,h}^{\dagger -}(s) \geq \sum_{h'=h+1}^{H} \max_{s\in\mathcal{S}, i\in[m]} \Delta_{i,h'}^{\dagger -}(s)$ holds for any $h \in [H]$, then $d$-portion attack will force the agents to follow the target policy with the attack loss and the attack cost bounded by

$$\mathbb{E}[\text{Loss1}(K, H)] \leq 2m^2\mathcal{R}(T)/\Delta_{min}, \ \mathbb{E}[\text{Cost}(K, H)] \leq 2m^3\mathcal{R}(T)/\Delta_{min}. \tag{5.4}$$

### 5.2.3 White-box reward poisoning attacks

As stated in Theorem 12, the reward poisoning only attacks may fail, if we limit the post-attack mean reward $\widetilde{R}$ to satisfy the same boundary conditions as those of the pre-attack mean reward $R$, i.e. $\widetilde{R} \in [0, 1]$. However, similar to the case with action poisoning only attacks, the reward poisoning only attacks can be efficiently successful for a class of target policies. The following condition specifies such class of target policies.

**Condition 2:** *For the underlying environment MG* $(\mathcal{S}, \{\mathcal{A}_i\}_{i=1}^m, H, P, \{R_i\}_{i=1}^m)$*, there exists constant* $\eta > 0$ *such that for any state* $s$*, any step* $h$*, and any agent* $i$*,* $(R_{i,h}(s, \pi^\dagger(s)) - \eta)/(H-h) \geq \Delta_R > 0$ *where* $\Delta_R = [\max_{s\times a\times h'} R_{i,h'}(s, a) - \min_{s\times a\times h'} R_{i,h'}(s, a)]$.

We now introduce an effective white-box reward attack strategies: $\eta$-gap attack. Specifically, at the step $h$ and state $s$, if agents all pick the target action, i.e., $\boldsymbol{a} = \pi_h^\dagger(s)$, the attacker does not attack, i.e. $\widetilde{r}_{i,h} = r_{i,h}$ for each agent $i$. If agent $i$ picks a non-target action, i.e., $\boldsymbol{a} \neq \pi_h^\dagger(s)$, the $\eta$-gap attack sets $\widetilde{r}_{i,h} = R_{i,h}(s, \pi^\dagger(s)) - (\eta + (H-h)\Delta_R)\mathbb{1}(a_i \neq \pi_{i,h}^\dagger(s))$ for each agent $i$. From Condition 2, we have $\widetilde{r}_{i,h} \in [0,1]$.

**Theorem 15.** If the attacker follows the $\eta$-gap attack strategy on the MG agents, the best response of each agent $i$ towards any policy $\pi_{-i}$ is $\pi_i^\dagger$. The target policy $\pi^\dagger$ is the {NE, CE, CCE} from any agent's point of view. If every state $s \in \mathcal{S}$ is reachable at every step $h \in [H]$ under the target policy, $\pi^\dagger$ is the unique {NE, CE, CCE}.

The detailed proof can be found in Appendix D.4.1. Theorem 15 shows that the target policy $\pi^\dagger$ is the unique {NE, CE, CCE} under the $\eta$-gap attack. Thus, if the agents follow an MARL algorithm that is able to learn an $\epsilon$-approximate {NE, CE, CCE}, the agents will learn a policy approximate to the target policy. Here, we discuss the high-level idea why the $\eta$-gap attack works. $\Delta_R$ is the difference between the upper bound and the lower bound of the mean rewards. Condition 2 implies that each action is close to other actions from every agent's point of view. Although we limit the post-attack mean reward $\widetilde{R}$ in $[0,1]$, the target policy can still appear to be optimal by making small changing to the rewards. Under Condition 2 and the $\eta$-gap attacks, the target actions strictly dominates any other non-target actions by at least $\eta$ and any other distinct policy is not an $\epsilon$-approximate CCE with different gap $\epsilon < \eta$. Thus, $\pi^\dagger$ becomes the unique {NE, CE, CCE}. In addition, we can derive upper bounds of the attack loss and the attack cost when attacking MARL algorithms with sub-linear best-in-hindsight regret.

**Theorem 16.** If the best-in-hindsight regret $\text{Reg}(K, H)$ of each agent's algorithm is bounded by a sub-linear bound $\mathcal{R}(T)$ for any MG in the absence of attack, then $\eta$-gap attack will force the agents to follow the target policy with the attack loss and the attack cost bounded by

$$\mathbb{E}[\text{Loss1}(k, H)] \leq m\mathcal{R}(T)/\eta, \ \mathbb{E}[\text{Cost}(K, H)] \leq m^2\mathcal{R}(T)/\eta. \tag{5.5}$$

We note that proposed sufficient conditions (namely Condition 1 and Condition 2), under which the action poisoning only attacks or the reward poisoning only attacks can be efficient and successful, may be strict. They may not always hold in practice. This motivates us to investigate mixed attack strategy to be discussed in the sequel.

## 5.3 Gray-box Attack Strategy and Analysis

In the gray-box attack setting, the attacker has no prior information about the underlying environment and the agents' algorithm, and it only observes samples generated when the agents interact with the environment. However, the attacker is given the target policy $\pi^\dagger$. Since the $\eta$-gap reward attack strategy and $d$-portion action attack strategy described in Section 5.2 for the white-box setting rely on the information of the underlying environment, these two attack strategies are not applicable in the gray-box setting. In addition, without the information of the underlying environment, the attacker cannot check whether the action poisoning attack alone or the reward poisoning attack alone can be efficiently successful. Building on insights obtained from the white-box attack strategies, we develop a mixed attack strategy for MG in the gray-box attack setting.

In the proposed mixed attack strategy, at the step $h$ and state $s$, if agent $i$ picks the target action, i.e., $a_{i,h} = \pi^\dagger_{i,h}(s)$, the attacker does not override the action and the reward, i.e. $\widetilde{a}_{i,h} = a_{i,h}$ and $\widetilde{r}_{i,h} = r_{i,h}$. If agent $i$ picks a non-target action, i.e., $a_{i,h} \neq \pi^\dagger_{i,h}(s)$, the attacker overrides its action $\widetilde{a}_{i,h} = \pi^\dagger_{i,h}(s)$ and then overrides the reward $\widetilde{r}_{i,h} = 0$.

**Theorem 17.** If the attacker follows the mixed attack strategy the best response of each agent $i$ towards any product policy $\pi_{-i}$ is $\pi^\dagger_i$. The optimal policy $\pi^\dagger$ is the unique {NE, CE, CCE}.

The detailed proof can be found in Appendix D.5.1. Here, we discuss the high-level idea why the mixed attack works. Under the mixed attacks, the state transitions are the same over the different actions and the reward of the non-target actions is worse than the target action. Thus, in the post-attack environment, the target policy is better than any other policy from any agent's point of view, and any other distinct policy is not an $\epsilon$-approximate CCE with different gap $\epsilon < R_{min}$, where $R_{min} = \min_{h \in [H]} \min_{s \in \mathcal{S}} \min_{i \in [m]} R_{i,h}(s, \pi_h^\dagger(s))$. Thus, $\pi^\dagger$ is the unique {NE, CE, CCE}. In addition, we can derive upper bounds of the attack loss and the attack cost when attacking some special MARL algorithms.

**Theorem 18.** If the best-in-hindsight regret Reg$(K, H)$ of each agent's algorithm is bounded by a sub-linear bound $\mathcal{R}(T)$ for any MG in the absence of attacks, then the mixed attacks will force the agents to follow the target policy $\pi^\dagger$ with the attack loss and the attack cost bounded by

$$\mathbb{E}[\text{Loss1}(K, H)] \leq m\mathcal{R}(T)/R_{min}, \ \mathbb{E}[\text{Cost}(K, H)] \leq 2m\mathcal{R}(T)/R_{min}. \tag{5.6}$$

## 5.4 Black-box Attack Strategy and Analysis

In the black-box attack setting, the attacker has no prior information about the underlying environment and the agents' algorithm, and it only observes the samples generated when the agents interact with the environment. The attacker aims to maximize the cumulative rewards under some specific reward functions $R_\dagger$ chosen by the attacker. But unlike in the gray-box case, the corresponding target policy $\pi^\dagger$ is also unknown for the attacker. After each time step, the attacker will receive the attacker reward $r_\dagger$. Since the optimal (target) policy that maximizes the attacker's reward is unknown, the attacker needs to explore the environment to obtain the optimal policy. As the mixed attack strategy described in Section 5.3 for the gray-box setting relies on the knowledge of the target policy, it is not applicable in the black-box setting.

However, by collecting observations and evaluating the attacker's reward function and transition probabilities of the underlying environment, the attacker can perform an approximate mixed attack strategy. In particular, we propose an approximate mixed attack strategy that has two phases: the exploration phase and the attack phase. In the exploration phase, the attacker explores the environment to identify an approximate optimal policy, while in the attack phase, the attacker performs the mixed attack strategy and forces the agents to learn the approximate optimal policy. The total attack cost (loss) will be the sum of attack cost (loss) of these two phases.

**Algorithm 5.1** Exploration phase for Markov games

```
Require: Stopping time $\tau$. Set $B(N) = (H\sqrt{S}+1)\sqrt{\log(2AH\tau/\delta)/(2N)}$.
Initialize $\overline{Q}_{\dagger,h}(s,\boldsymbol{a}) = \overline{V}_{\dagger,h}(s,\boldsymbol{a}) = H$, $\underline{Q}_{\dagger,h}(s,\boldsymbol{a}) = \underline{V}_{\dagger,h}(s,\boldsymbol{a}) = 0$, $\overline{V}_{\dagger,H+1} = \underline{V}_{\dagger,H+1} = \mathbf{0}$,
    $\Delta = \infty$, $N_0(s) = N_h(s,\boldsymbol{a}) = N_h(s,\boldsymbol{a},s') = 0$ and $\hat{R}_{\dagger,h}(s,\boldsymbol{a}) = 0$ for any $(s,s',\boldsymbol{a},i,h)$.
for episode $k = 1,\dots,\tau$ do
    for step $h = H,\dots,1$ do
        for each $(s,\boldsymbol{a}) \in \mathcal{S}\times\mathcal{A}$ with $N_h(s,\boldsymbol{a}) > 0$ do
            Update $\overline{Q}_{\dagger,h}(s,\boldsymbol{a}) = \min\{\hat{R}_{\dagger,h} + \hat{\mathbb{P}}_h\overline{V}_{\dagger,h+1}(s,\boldsymbol{a}) + B(N_h(s,\boldsymbol{a})), H\}$ and $\underline{Q}_{\dagger,h}(s,\boldsymbol{a}) =$
            $\max\{\hat{R}_{\dagger,h} + \hat{\mathbb{P}}_h\underline{V}_{\dagger,h+1}(s,\boldsymbol{a}) - B(N_h(s,\boldsymbol{a})), 0\}$.
        end for
        for each $s \in \mathcal{S}$ with $N_h(s,\boldsymbol{a}) > 0$ do
            Update $\pi_h(s) = \max_{\boldsymbol{a}\in\mathcal{A}} \overline{Q}_{\dagger,h}(s,\boldsymbol{a})$.
            Update $\overline{V}_{\dagger,h}(s,\boldsymbol{a}) = \overline{Q}_{\dagger,h}(s,\pi_h(s))$ and $\underline{V}_{\dagger,h}(s,\boldsymbol{a}) = \underline{Q}_{\dagger,h}(s,\pi_h(s))$.
        end for
    end for
    if $\mathbb{E}_{s\sim\hat{\mathbb{P}}_0(\cdot)}(\overline{V}_{\dagger,1}(s) - \underline{V}_{\dagger,1}(s)) + H\sqrt{\frac{S\log(2\tau/\delta)}{2k}} \le \Delta$ then
        $\Delta = \mathbb{E}_{s\sim\hat{\mathbb{P}}_0(\cdot)}(\overline{V}_{\dagger,1}(s) - \underline{V}_{\dagger,1}(s)) + H\sqrt{\frac{S\log(2\tau/\delta)}{2k}}$ and $\pi^\dagger = \pi$.
    end if
    for step $h = 1,\dots,H$ do
        Attacker overrides each agent's action by changing $a_{i,h}$ to $\widetilde{a}_{i,h}$, where $\widetilde{\boldsymbol{a}}_h = \pi_h(s_h)$.
        The environment returns the reward $r_{i,h}$ and the next state $s_{h+1}$ according to action $\widetilde{\boldsymbol{a}}_h$.
        The attacker receive its reward $r_{\dagger,h}$.
        Attacker overrides each agent's reward by changing $r_{i,h}$ to $\widetilde{r}_{i,h} = 1$.
        Add 1 to $N_h(s_h,\widetilde{\boldsymbol{a}}_h)$ and $N_h(s_h,\widetilde{\boldsymbol{a}}_h,s_{h+1})$. $\hat{\mathbb{P}}_h(\cdot|s_h,\widetilde{\boldsymbol{a}}_h) = N_h(s_h,\widetilde{\boldsymbol{a}}_h,\cdot)/N_h(s_h,\widetilde{\boldsymbol{a}}_h)$
        Update $\hat{R}_{\dagger,h}(s_h,\widetilde{\boldsymbol{a}}_h) = \hat{R}_{\dagger,h}(s_h,\widetilde{\boldsymbol{a}}_h) + (r_{\dagger,t} - \hat{R}_{\dagger,h}(s_h,\widetilde{\boldsymbol{a}}_h)/N_h(s_h,\widetilde{\boldsymbol{a}}_h)$.
    end for
    Update $N_0(s_1) = N_0(s_1) + 1$ and $\hat{\mathbb{P}}_0(\cdot) = N_0(\cdot)/k$.
end for
Return $\pi^\dagger$.
```

In the exploration phase, the approximate mixed attack strategy uses an optimal-policy identification algorithm, which is summarized in Algorithm 5.1. It will return an approximate optimal policy $\pi^\dagger$. Note that $\pi^k$ denotes the product policy deployed by the agents for each episode $k$. $\overline{V}$ is the upper bound of $V^{\pi^*}$ and $\underline{V}$ is the lower bound of $V^{\pi^k}$. By minimizing $\overline{V} - \underline{V}$, Algorithm 5.1 finds an approximate optimal policy $\pi^\dagger$. Here, we assume that the reward on the approximate optimal policy $\pi^\dagger$ is positive, i.e. $R_{min} = \min_{h\in[H]} \min_{s\in\mathcal{S}} \min_{i\in[m]} R_{i,h}(s, \pi^\dagger_h(s)) > 0$. In the exploration phase, the attacker will override both the agents' actions and rewards.

After the exploration phase, the approximate mixed attack strategy performs the attack phase. The attacker will override both the agents' actions and rewards in this phase. At the step $h$ and state $s$, if agent $i$ picks the action $\pi^\dagger_{i,h}(s)$, the attacker does not override actions and rewards, i.e. $\widetilde{a}_{i,h} = a_{i,h}$ and $\widetilde{r}_{i,h} = r_{i,h}$. If agent $i$ picks action $a_{i,h} \neq \pi^\dagger_{i,h}(s)$, the attacker overrides the action $\widetilde{a}_{i,h} = a_{i,h}$ and then overrides the reward $\widetilde{r}_{i,h} = 0$. The attack strategy in the attack phase is same with the mixed attack strategy. From Theorem 17, in the attack phase, the best response of each agent $i$ towards product policy $\pi^\dagger_{-i}$ is $\pi^\dagger_i$ and $\pi^\dagger$ is the unique NE. Here, we discuss the high-level idea why the approximate mixed attack works. The attacker finds an approximate optimal policy $\pi^\dagger$ by Algorithm 5.1. If $\pi^*$ is close to $\pi^\dagger$ and the exploration phase is sub-linear time dependent, the performance of the approximate mixed attack strategy will be close to the mixed attack strategy. We build a confidence bound to show the value function difference between $\pi^*$ and $\pi^\dagger$ in the following lemma.

**Lemma 14.** If the attacker follows the Algorithm 5.1 on the agents, for any $\delta \in (0, 1)$, with probability at least $1 - 5\delta$, the following bound holds:

$$\mathbb{E}_{s_1\sim P_0(\cdot)}[V^{\pi^*}_{\dagger,1}(s_1) - V^{\pi^\dagger}_{\dagger,1}(s_1)] \leq 2H^2 S\sqrt{2A\log(2SAH\tau/\delta)/\tau}. \tag{5.7}$$

We now investigate the impact of the approximate mixed attack strategy attack on V-learning [42], a simple, efficient, decentralized algorithm for MARL. For completeness, we describe the main steps of V-learning algorithm [42] in Algorithm 5.2 and the adversarial bandit algorithm in Algorithm 5.3.

**Algorithm 5.2** V-learning [42]

```
For any $(s, a, h)$, $V_h(s) \leftarrow H + 1 - h$, $N_h(s) \leftarrow 0$, $\pi_h(a|s) \leftarrow 1/A$.
for episodes $k = 1, \ldots, K$ do
    receive $s_1$
    for episodes $h = 1, \ldots, H$ do
        take action $a_h \sim \pi_h(\cdot|s_h)$, observe reward $r_h$ and next state $s_{h+1}$.
        $t = N_h(s_h) \leftarrow N_h(s_h) + 1$.
        $\overline{V}_h(s_h) \leftarrow (1 - \alpha_t)\overline{V}_h(s_h) + \alpha_t(rh + V_{h+1}(s_{h+1}) + \beta_t)$.
        $V_h(s_h) \leftarrow \min\{H + 1 - h, \overline{V}_h(s_h)\}$
        $\pi_h(\cdot|s_h) \leftarrow$ ADV_BANDIT_UPDATE$(a_h, \frac{H - r_h - V_{h+1}(s_{h+1})}{H})$ on $(s_h, h)^{th}$ adversarial
        bandit.
    end for
end for
```

**Algorithm 5.3** FTRL for Weighted External Regret [42]

```
For any $b \in \mathcal{B}$, $\theta_1(b) \leftarrow 1/B$.
for episode $t = 1, \ldots, K$ do
    Take action $b_t \sim \theta_t(\cdot)$, and observe loss $\widetilde{l}_t(b_t)$.
    $\hat{l}_t(b) \leftarrow \widetilde{l}_t(b_t)\mathbb{1}[b_t = b]/(\theta_t(b) + \gamma_t)$ for all $b \in \mathcal{B}$.
    $\theta_{t+1}(b) \propto \exp[-(\gamma_t/w_t)\sum_{i=1}^{t} w_i\hat{l}_i(b)]$
end for
```

We use the same learning rate $\alpha_t$ in [42].

**Theorem 19.** Suppose ADV_BANDIT_UPDATE of V-learning follows Algorithm 5.3 and it chooses hyper-parameter $w_t = \alpha_t \left(\prod_{i=2}^{t}(1 - \alpha_i)\right)^{-1}$, $\gamma_t = \sqrt{\frac{H \log B}{Bt}}$ and $\alpha_t = \frac{H+1}{H+t}$. For given $K$ and any $\delta \in (0, 1)$, let $\iota = \log(mHSAK/\delta)$. The attack loss and the attack cost of the approximate mixed attack strategy during these $K$ episodes are bounded by

$$
\begin{aligned}
&\mathbb{E}\left[\text{Loss2}(K, H)\right] \leq H\tau + \frac{40}{R_{min}} m\sqrt{H^9 ASK\iota} + 2H^2 SK\sqrt{2A\iota/\tau}, \\
&\mathbb{E}\left[\text{Cost}(K, H)\right] \leq 2mH\tau + \frac{80}{R_{min}}\sqrt{H^5 ASK\iota}.
\end{aligned}
\tag{5.8}
$$

Let $\hat{\pi}$ be the executing output policy of V-learning, the attack loss of the executing output policy $\hat{\pi}$

is upper bounded by

$$V^{\pi^*}_{\dagger,1}(s_1) - V^{\hat{\pi}}_{\dagger,1}(s_1) \leq \frac{20mS}{R_{min}}\sqrt{\frac{H^7 A\iota}{K}} + \frac{2\tau m H^2 S}{K} + 2H^2 S\sqrt{2A\iota/\tau}. \tag{5.9}$$

If we choose the stopping time of the exploration phase $\tau = K^{2/3}$, the attack loss and the attack cost of the approximate mixed attack strategy during these $K$ episodes are bounded by $\mathcal{O}(K^{2/3})$ and $V^{\pi^*}_{\dagger,1}(s_1) - V^{\hat{\pi}}_{\dagger,1}(s_1) \leq \mathcal{O}(K^{-1/3})$.

## 5.5 Numerical Results

In this section, we empirically compare the performance of the action poisoning only attack strategy ($d$-portion attack), the reward poisoning only attack strategy ($\eta$-gap attack) and the mixed attack strategy.

We consider a simple case of Markov game where $m = 2$, $H = 2$ and $|\mathcal{S}| = 3$. This Markov game is the example in Appendix D.2.2. The initial state is $s_1$ at $h = 1$ and the transition probabilities are:

$$\begin{aligned} &P(s_2|s_1, a) = 0.9, P(s_3|s_1, a) = 0.1, \text{ if } a = \text{(Defect, Defect)}, \\ &P(s_2|s_1, a) = 0.1, P(s_3|s_1, a) = 0.9, \text{ if } a \neq \text{(Defect, Defect)}. \end{aligned} \tag{5.10}$$

The reward functions are expressed in the following Table 5.2.

Table 5.2: Reward matrices

| state $s_1$ | Cooperate | Defect |
|---|---|---|
| Cooperate | (1, 1) | (0.5, 0.5) |
| Defect | (0.5, 0.5) | (0.2, 0.2) |

| state $s_2$ | Cooperate | Defect |
|---|---|---|
| Cooperate | (1, 1) | (0.5, 0.5) |
| Defect | (0.5, 0.5) | (0.1, 0.1) |

| state $s_3$ | Cooperate | Defect |
|---|---|---|
| Cooperate | (1, 1) | (0.5, 0.5) |
| Defect | (0.5, 0.5) | (0.9, 0.9) |

We set the total number of episodes $K = 10^7$. We set two different target policies. For the first target policy, no action/reward poisoning Markov attack strategy alone can efficiently and successfully attack MARL agents. For the second target policy, the $d$-portion attack and the $\eta$-gap attack can efficiently and successfully attack MARL agents.

**Case 1.** The target policy is that the two agents both choose to defect at any state. As stated in Section 5.2 and Appendix 5.2.1, the Condition 1 and Condition 2 do not hold for this Markov game and target policy, and no action/reward poisoning Markov attack strategy alone can efficiently and successfully attack MARL agents.

In Figure 5.1, we illustrate the mixed attack strategy, the $d$-portion attack strategy and the $\eta$-gap attack strategy on V-learning agents for the proposed MG. The $x$-axis represents the episode $k$ in the MG. The $y$-axis represents the cumulative attack cost and attack loss that change over time steps. The results show that, the attack cost and attack loss of the mixed attack strategy sublinearly scale as $T$, but the attack cost and attack loss of the $d$-portion attack strategy and the $\eta$-gap attack strategy linearly scale as $T$, which is consistent with our analysis.

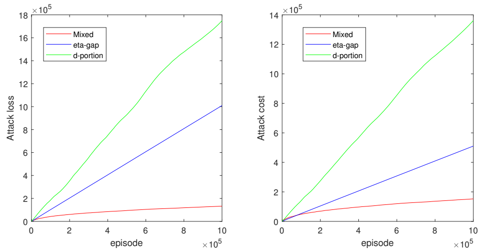


Figure 5.1: The attack loss (cost) on case 1.

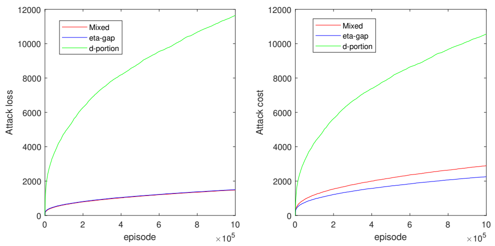


Figure 5.2: The attack loss (cost) on case 2.

**Case 2.** The target policy is that the two agents choose to cooperate at state $s_1$ and $s_2$ but to defect at state $s_3$. As stated in Section 5.2 and Appendix D.2, the Condition 1 and Condition 2 hold for this Markov game and target policy. Thus, the $d$-portion attack strategy and the $\eta$-gap attack strategy can efficiently and successfully attack MARL agents.

In Figure 5.2, we illustrate the mixed attack strategy, the $d$-portion attack strategy and the $\eta$-gap attack strategy on V-learning agents for the proposed MG. The results show that, the attack cost and attack loss of all three strategies sublinearly scale as $T$, which is consistent with our analysis.

Additional numerical results that compare the performance of the mixed attack strategy and the approximate mixed attack strategy are provided in the following. We consider a multi-agent system with three recycling robots. In this scenario, a mobile robot with a rechargeable battery and a solar battery collects empty soda cans in a city. The number of agents is 3, i.e. $m = 3$. Each robot has two different energy levels, high energy level and low energy level, resulting in $8$ states in total, i.e. $S = 8$.

Each robot can choose a conservative action or an aggressive action, so $A_i = 2$ and $A = 8$. At the high energy level, the conservative action is to wait in some place to save energy and then the mean reward is $0.4$. At the high energy level, the aggressive action is to search for cans. All the robots that choose to search will get a basic reward $0.2$ and equally share an additionally mean reward $0.9$. For example, if all robots choose to search at a step, the mean reward of each robot is $0.5$. At the low energy level, the conservative action is to return to change the battery and find the cans on the way. In this state and action, the robot only gets a low mean reward $0.2$. At the low energy level, the conservative action is to wait in some place to save energy and then the mean reward is $0.3$. We use Gaussian distribution to randomize the reward signal.

We set the total number of steps $H = 6$. At the step $h \leq 3$, it is the daytime and the robot who chooses to search will change to the low energy level with low probability $0.3$. At the step $h \geq 4$, it is the night and the robot who chooses to search will change to the low energy level with high probability $0.7$. The energy level transition probabilities are stated in Figure 5.3 and Figure 5.4. 'H' represents the high energy level. 'L' represents the low energy level. 'C' represents the conservative action. 'A' represents the aggressive action.

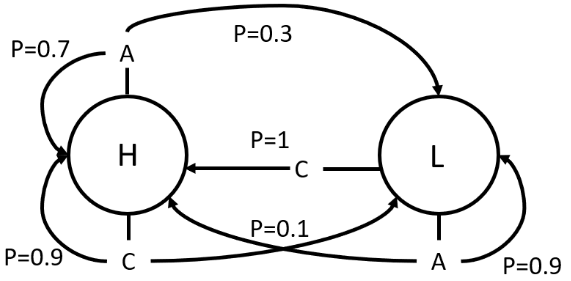


Figure 5.3: Energy level transitions at $h \leq 3$.

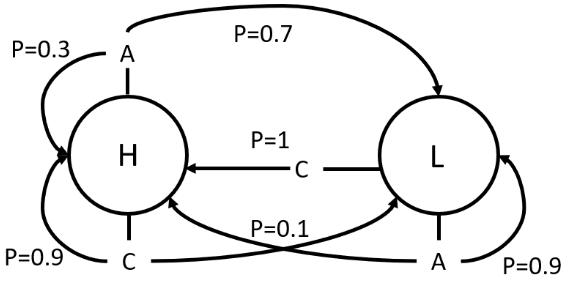


Figure 5.4: Energy level transitions at $h \geq 4$.

We consider two different attack goals: (1) maximize the first robot's rewards; (2) minimize the the second robot's and the third robot's rewards. For the gray box case, we provide the target policy that maximizes the first robot's rewards or minimizes the the second robot's and the third robot's rewards. For the black box case, we set $R_{\dagger,h} = R_{1,h}$ to maximize the first robot's rewards and set $R_{\dagger,h} = 1 - R_{2,h}/2 - R_{3,h}/2$ to minimize the second robot's and the third robot's rewards.

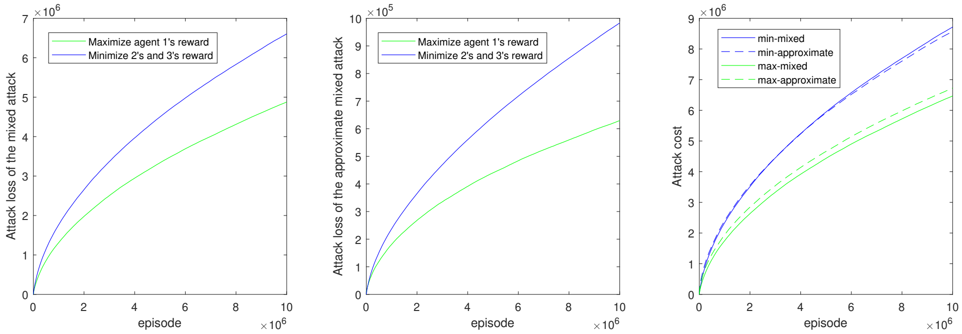


Figure 5.5: The cumulative attack loss and cost of the mixed attack and the approximate mixed attack.

We set the total number of episodes $K = 10^7$. In Figure 5.5, we illustrate the mixed attack strategy and approximate-mixed attack strategy on V-learning agents for the proposed MG. The $x$-axis represents the episode $k$ in the MG. The $y$-axis represents the cumulative attack cost and attack loss that change over time steps. The results show that, the attack cost and attack loss of the mixed attack strategy and approximate-mixed attack strategy sublinearly scale as $T$, which is consistent with our analysis. Furthermore, Figure 5.5 shows that the performance of the approximate-mixed attack strategy nearly match that of the mixed attack strategy. This illustrates that the proposed approximate-mixed attack strategy is very effective in the black-box scenario.

## 5.6 Conclusion

In this chapter, we have introduced an adversarial attack model on MARL. We have discussed the attack problem in three different settings: the white-box, the gray-box and the black-box settings. We have shown that the power of action poisoning only attacks and reward poisoning only attacks is limited. Even in the white-box setting, there exist some MGs, under which no action poisoning only attack strategy or reward poisoning only attack strategy can be efficient and successful. We have then characterized conditions when action poisoning only attacks or only reward poisoning only attacks can efficiently work. We have further introduced the mixed attack strategy in the gray-box setting that can efficiently attack any sub-linear-regret MARL agents. Finally, we have proposed the approximate mixed attack strategy in the black-box setting and shown its effectiveness on V-learning.

# Chapter 6

# Action Robust Reinforcement Learning

In this chapter, we focus on action robust RL with the probabilistic policy execution uncertainty, in which, instead of always carrying out the action specified by the policy, the agent will take the action specified by the policy with probability $1-\rho$ and an alternative adversarial action with probability $\rho$. In Section 6.1, we describe the model. In Section 6.2, we show the existence of an optimal policy on the action robust MDPs with probabilistic policy execution uncertainty and provide the action robust Bellman optimality equation for its solution. In Section 6.3, we develop a model-based algorithm, Action Robust Reinforcement Learning with Certificates (ARRLC), for episodic action robust MDPs, and show that it achieves minimax order optimal regret and minimax order optimal sample complexity. In Section 6.4, we develop a model-free algorithm for episodic action robust MDPs, and analyze its regret and sample complexity. In Section 6.5, we conduct numerical experiments to validate the robustness of our approach. In our experiments, our robust algorithm achieves a much higher reward than the non-robust RL algorithm when being tested with some action perturbations; and our ARRLC algorithm converges much faster than other robust algorithms. The proofs are collected in Appendix E

## 6.1 Problem formulation

**Tabular MDPs.** We consider a tabular episodic MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, H, P, R)$ stated in Chapter 4.1.

The agent interacts with the MDP in episodes indexed by $k$. Each episode $k$ is a trajectory $\{s_1^k, a_1^k, r_1^k, \cdots, s_H^k, a_H^k, r_H^k\}$ of $H$ states $s_h^k \in \mathcal{S}$, actions $a_h^k \in \mathcal{A}$, and rewards $r_h^k \in [0,1]$. At each step $h \in [H]$ of episode $k$, the agent observes the state $s_h^k$ and chooses an action $a_h^k$. After receiving the action, the environment generates a random reward $r_h^k \in [0,1]$ derived from a distribution with mean $R_h(s_h^k, a_h^k)$ and next state $s_{h+1}^k$ that is drawn from the distribution $P_h(\cdot|s_h^k, a_h^k)$. For notational simplicity, we assume that the initial states $s_1^k = s_1$ are deterministic in different episode $k$.

A (stochastic) Markov policy of the agent is a set of $H$ maps $\pi := \{\pi_h : \mathcal{S} \to \Delta_{\mathcal{A}}\}_{h\in[H]}$, where $\Delta_{\mathcal{A}}$ denotes the simplex over $\mathcal{A}$. We use notation $\pi_h(a|s)$ to denote the probability of taking action $a$ in state $s$ under stochastic policy $\pi$ at step $h$. A deterministic policy is a policy that maps each state to a particular action. Therefore, when it is clear from the context, we abuse the notation $\pi_h(s)$ for a deterministic policy $\pi$ to denote the action $a$ which satisfies $\pi_h(a|s) = 1$.

**Action robust MDPs.** In the action robust case, the policy execution is not accurate and lies in some uncertainty set centered on the agent's policy $\pi$. Denote the actual behavior policy by $\widetilde{\pi}$ where $\widetilde{\pi} \in \Pi(\pi)$ and $\Pi(\pi)$ is the uncertainty set of the policy execution. Denote the actual behavior action at episode $k$ and step $h$ by $\widetilde{a}_h^k$ where $\widetilde{a}_h^k \sim \widetilde{\pi}_h^k$. Define the action robust value function of a policy $\pi$ as the worst-case expected accumulated reward over following any policy in the uncertainty set $\Pi(\pi)$ centered on a fixed policy $\pi$:

$$V_h^{\pi}(s) = \min_{\widetilde{\pi}\in\Pi(\pi)} \mathbb{E}_{\widetilde{\pi}}\left[\sum_{h'=h}^{H} R_{h'}(s_{h'}, a_{h'})|s_h = s\right]. \tag{6.1}$$

$V_h^{\pi}$ represents the action robust value function of policy $\pi$ at step $h$. Similarly, define the action robust $Q$-function of a policy $\pi$:

$$Q_h^{\pi}(s,a) = \min_{\widetilde{\pi}\in\Pi(\pi)} \mathbb{E}_{\widetilde{\pi}}\left[\sum_{h'=h}^{H} R_{h'}(s_{h'}, a_{h'})|s_h = s, a_h = a\right]. \tag{6.2}$$

The goal of action robust RL is to find the optimal robust policy $\pi^*$ that maximizes the worst-case accumulated reward: $\pi^* = \arg\max_{\pi} V_1^{\pi}(s), \forall s \in \mathcal{S}$. We also denote $V^{\pi^*}$ and $Q^{\pi^*}$ by $V^*$ and $Q^*$.

**Probabilistic policy execution uncertain set.** We follow the setting of the probabilistic action robust MDP (PR-MDP) introduced in [97] to construct the probabilistic policy execution uncertain set. For some $0 \leq \rho \leq 1$, the policy execution uncertain set is defined as:

$$\Pi^\rho(\pi) := \{\widetilde{\pi} : \forall s, \forall h, \exists \pi'_h(\cdot|s) \in \Delta_\mathcal{A} \text{ such that } \widetilde{\pi}_h(\cdot|s) = (1-\rho)\pi_h(\cdot|s) + \rho\pi'_h(\cdot|s)\}. \tag{6.3}$$

The policy execution uncertain set can be even simpler expressed as $\Pi^\rho(\pi) = (1-\rho)\pi + \rho(\Delta_\mathcal{A})^{S\times H}$.

In this setting, an optimal probabilistic robust policy is optimal w.r.t. a scenario in which, with probability at most $\rho$, an adversary takes control and performs the worst possible action. We call $\pi'$ as the adversarial policy. For different agent's policy $\pi$, the corresponding adversarial policy $\pi'$ that minimizes the cumulative reward may be different.

**Additional notations.** We set $\iota = \log(2SAHK/\delta)$ for $\delta > 0$. For simplicity of notation, we treat $P$ as a linear operator such that $[P_hV](s,a) := \mathbb{E}_{s'\sim P_h(\cdot|s,a)}V(s')$, and we define two additional operators $\mathbb{D}$ and $\mathbb{V}$ as follows: $[\mathbb{D}_{\pi_h}Q](s) := \mathbb{E}_{a\sim\pi_h(\cdot|s)}Q(s,a)$ and

$$\begin{aligned}\mathbb{V}_{P_h}V_{h+1}(s,a) &:= \sum_{s'} P_h(s'|s,a)\left(V_{h+1}(s') - [P_hV_{h+1}](s,a)\right)^2 \\ &= [P_h(V_{h+1})^2](s,a) - ([P_hV_{h+1}](s,a))^2.\end{aligned}$$

## 6.2 Existence of the optimal robust policy

For the standard tabular MDPs, when the state space, action space, and the horizon are all finite, there always exists an optimal policy. In addition, if the reward functions and the transition probabilities are known to the agent, the optimal policy can be solved by solving the Bellman optimality equation. In the following theorem, we show that the optimal policy also always exists in action robust MDPs and can be solved by the action robust Bellman optimality equation.

**Proposition 4.** If the uncertainty set of the policy execution has the form in (6.3), the following perfect duality holds for all $s \in \mathcal{S}$ and all $h \in [H]$:

$$\begin{aligned} &\max_{\pi} \min_{\widetilde{\pi} \in \Pi^{\rho}(\pi)} \mathbb{E}_{\widetilde{\pi}} \left[ \sum_{h'=h}^{H} R_{h'}(s_{h'}, a_{h'}) | s_h = s \right] \\ &= \min_{\widetilde{\pi} \in \Pi^{\rho}(\pi)} \max_{\pi} \mathbb{E}_{\widetilde{\pi}} \left[ \sum_{h'=h}^{H} R_{h'}(s_{h'}, a_{h'}) | s_h = s \right]. \end{aligned} \tag{6.4}$$

There always exists a deterministic optimal robust policy $\pi^*$. The problem can be solved by the iteration of the action robust Bellman optimality equation on $h = H, \cdots, 1$. The action robust Bellman equation and the action robust Bellman optimality equation are:

$$\begin{cases} V_h^{\pi}(s) = (1-\rho)[\mathbb{D}_{\pi_h} Q_h^{\pi}](s) + \rho \min_{a \in \mathcal{A}} Q_h^{\pi}(s,a), \\ Q_h^{\pi}(s,a) = R_h(s,a) + [P_h V_{h+1}^{\pi}](s,a), \\ V_{H+1}^{\pi}(s) = 0, \ \forall s \in \mathcal{S}. \end{cases} \tag{6.5}$$

$$\begin{cases} V_h^{*}(s) = (1-\rho) \max_{a \in \mathcal{A}} Q_h^{*}(s,a) + \rho \min_{b \in \mathcal{A}} Q_h^{*}(s,b), \\ Q_h^{*}(s,a) = R_h(s,a) + [P_h V_{h+1}^{*}](s,a), \\ V_{H+1}^{*}(s) = 0, \ \forall s \in \mathcal{S}. \end{cases} \tag{6.6}$$

We define

$$C_h^{\pi,\pi',\rho}(s) := \mathbb{E}_{\widetilde{\pi}} \left[ \sum_{h'=h}^{H} R_{h'}(s_{h'}, a_{h'}) | s_h = s \right]. \tag{6.7}$$

The perfect duality of the control problems in (6.4) is equivalent to $\max_{\pi} \min_{\pi'} C_h^{\pi,\pi',\rho}(s) = \min_{\pi'} \max_{\pi} C_h^{\pi,\pi',\rho}(s)$. We provide the detailed proof of the perfect duality and the existence of the optimal policy in Appendix E.1. Our proposed model-based algorithm in Section 6.3 and model-free algorithm in Section 6.4 are based on the action robust Bellman optimality equation. Using the iteration of the proposed action robust Bellman equation to solve the robust problem can simultaneously update the adversary policy and agent policy and avoid inefficient alternating updates.

**Algorithm 6.1** ARRLC (**A**ction **R**obust **R**einforcement **L**earning with **C**ertificates)

```
Initialize $\overline{V}_h(s) = H-h+1$, $\overline{Q}_h(s,a) = H-h+1$, $\underline{V}_h(s) = 0$, $\underline{Q}_h(s,a) = 0$, $\hat{r}_h(s,a)$, $N_h(s,a) = 0$ and $N_h(s,a,s') = 0$ for any state $s \in \mathcal{S}$, any action $a \in \mathcal{A}$ and any step $h \in [H]$. $\overline{V}_{H+1}(s) = \underline{V}_{H+1}(s) = 0$ and $\overline{Q}_{H+1}(s,a) = \underline{Q}_{H+1}(s,a) = 0$ for any $s$ and $a$. $\Delta = H$.
for episode $k = 1, 2, \ldots, K$ do
    for step $h = 1, 2, \ldots, H$ do
        Observe $s_h^k$.
        Set $\overline{\pi}_h^k(s) = \arg\max_a \overline{Q}_h(s,a)$ , $\underline{\pi}_h^k(s) = \arg\min_a \underline{Q}_h(s,a)$, $\widetilde{\pi}_h^k = (1-\rho)\overline{\pi}_h^k + \rho\underline{\pi}_h^k$.
        Take action $a_h^k \sim \widetilde{\pi}_h^k(\cdot|s_h^k)$.
        Receive reward $r_h^k$ and observe $s_{h+1}^k$.
        Set $N_h(s_h^k, a_h^k) \leftarrow N_h(s_h^k, a_h^k) + 1$, $N_h(s_h^k, a_h^k, s_{h+1}^k) \leftarrow N_h(s_h^k, a_h^k, s_{h+1}^k) + 1$.
        Set $\hat{r}_h^k(s_h^k, a_h^k) \leftarrow \hat{r}_h^k(s_h^k, a_h^k) + (r_h^k - \hat{r}_h^k(s_h^k, a_h^k))/N_h(s_h^k, a_h^k)$.
        Set $\hat{P}_h(\cdot|s_h^k, a_h^k) = N_h(s_h^k, a_h^k, \cdot)/N_h(s_h^k, a_h^k)$.
    end for
    Output policy $\overline{\pi}^k$ with certificates $\mathcal{I}_k = [\underline{V}_1(s_1^k), \overline{V}_1(s_1^k)]$ and $\epsilon_k = |\mathcal{I}_k|$ .
    if $\epsilon_k < \Delta$ then
        $\Delta \leftarrow \epsilon_k$ and $\pi^{out} \leftarrow \overline{\pi}^k$.
    end if
    for step $h = H, H-1, \ldots, 1$ do
        for each $(s,a) \in \mathcal{S} \times \mathcal{A}$ with $N_h(s,a) > 0$ do
            Set $\theta_h(s,a) = \sqrt{\frac{2\mathbb{V}_{\hat{P}_h}[(\overline{V}_{h+1}+\underline{V}_{h+1})/2](s,a)\iota}{N_h(s,a)}} + \sqrt{\frac{2\hat{r}_h(s,a)\iota}{N_h(s,a)}} + \frac{\hat{P}_h\left(\overline{V}_{h+1}-\underline{V}_{h+1}\right)(s,a)}{H} + \frac{(24H^2+7H+7)\iota}{3N_h(s,a)}$,
            $\overline{Q}_h(s,a) \leftarrow \min\{H-h+1, \hat{r}_h(s,a) + \hat{P}_h\overline{V}_{h+1}(s,a) + \theta_h(s,a)\}$,
            $\underline{Q}_h(s,a) \leftarrow \max\{0, \hat{r}_h(s,a) + \hat{P}_h\underline{V}_{h+1}(s,a) - \theta_h(s,a)\}$,
            $\overline{\pi}_h^{k+1}(s) = \arg\max_a \overline{Q}_h(s,a)$ , $\underline{\pi}_h^{k+1}(s) = \arg\min_a \underline{Q}_h(s,a)$,
            $\overline{V}_h(s) \leftarrow (1-\rho)\overline{Q}_h(s, \overline{\pi}_h^{k+1}(s)) + \rho\overline{Q}_h(s, \underline{\pi}_h^{k+1}(s))$,
            $\underline{V}_h(s) \leftarrow (1-\rho)\underline{Q}_h(s, \overline{\pi}_h^{k+1}(s)) + \rho\underline{Q}_h(s, \underline{\pi}_h^{k+1}(s))$.
        end for
    end for
end for
Return $\pi^{out}$
```

## 6.3 Model-based algorithm and main results

In this section, we introduce the proposed **A**ction **R**obust **R**einforcement **L**earning with **C**ertificates (ARRLC) algorithm and provides its theoretical guarantee. The pseudo code is listed in Algorithm 6.1. Here, we highlight the main idea of our algorithm. Algorithm 6.1 trains the agent in a clean (simulation) environment and learns a policy that performs well when applied to a perturbed environment with probabilistic policy execution uncertainty. To simulate the action perturbation, Algorithm 6.1 chooses an adversarial action with probability $\rho$. To learn the agent's optimal policy and the corresponding adversarial policy, Algorithm 6.1 computes an optimistic estimate $\overline{Q}$ of $Q^*$ and a pessimistic estimate $\underline{Q}$ of $Q^{\overline{\pi}^k}$. Algorithm 6.1 uses the optimistic estimates to explore the possible optimal policy $\overline{\pi}$ and uses the pessimistic estimates to explore the possible adversarial policy $\underline{\pi}$. As shown later in Lemma 17, $\overline{V} \geq V^* \geq V^{\overline{\pi}} \geq \underline{V}$ holds with high probabilities. The optimistic and pessimistic estimates $\overline{V}$ and $\underline{V}$ can provide policy certificates, which bounds the cumulative rewards of the return policy $\overline{\pi}^k$ and $\overline{V}-\underline{V}$ bounds the sub-optimality of the return policy $\overline{\pi}^k$ with high probabilities. The policy certificates can give us some insights about the performance of $\overline{\pi}^k$ in the perturbed environment with probabilistic policy execution uncertainty.

### 6.3.1 Algorithm description

We now describe the proposed ARRLC algorithm in more details. In each episode, the ARRLC algorithm can be decomposed into two parts.

- Line 3-11 (Sample trajectory and update the model estimate): Simulates the action robust MDP, executes the behavior policy $\widetilde{\pi}$, collects samples, and updates the estimate of the reward and the transition.

- Line 16-25 (Adversarial planning from the estimated model): Performs value iteration with bonus to estimate the robust value functions using the empirical estimate of the transition $\hat{P}$, computes a new policy $\overline{\pi}$ that is optimal respect to the estimated robust value functions, and computes a new optimal adversarial policy $\underline{\pi}$ respect to the agent's policy $\overline{\pi}$.

At a high-level, this two-phase policy is standard in the majority of model-based RL algorithms [5, 19]. Algorithm 6.1 shares similar structure with ORLC (Optimistic Reinforcement Learning with Certificates) in [19] but has some significant differences in line 5-6 and line 18-23. The first main difference is that the ARRLC algorithm simulates the probabilistic policy execution uncertainty by choosing an adversarial action with probability $\rho$. The adversarial policy and the adversarial actions are computed by the ARRLC algorithm. The second main difference is that the ARRLC algorithm simultaneously plans the agent policy $\overline{\pi}$ and the adversarial policy $\underline{\pi}$ by the action robust Bellman optimality equation.

These two main difference brings two main challenges in the design and analysis of our algorithm.

(1) The ARRLC algorithm simultaneously plans the agent policy and the adversarial policy. However the planned adversarial policy $\underline{\pi}$ is not necessarily the true optimal adversary policy towards the agent policy $\overline{\pi}$ because of the estimation error of the value functions. We carefully design the bonus items and the update role of the value functions so that $\overline{V}_h(s) \geq V_h^*(s) \geq V_h^{\overline{\pi}}(s) \geq \underline{V}_h(s)$ and $\overline{Q}_h(s,a) \geq Q_h^*(s,a) \geq Q_h^{\overline{\pi}}(s,a) \geq \underline{Q}_h(s,a)$ hold for all $s$ and $a$.

(2) A crucial step in many UCB-type algorithms based on Bernstein inequality is bounding the sum of variance of estimated value function across the planning horizon. The behavior policies in these UCB-type algorithms are deterministic. However, the behavior policy in our ARRLC algorithm is not deterministic due to the simulation of the adversary's behavior. The total variance is the weighted sum of the sum of variance of estimated value function across two trajectories. Even if action $\overline{\pi}(s_h^k)$ or $\underline{\pi}(s_h^k)$ is not sampled at state $s_h^k$, it counts in the total variance. Thus, the sum of variance is no longer simply the variance of the sum of rewards per episode, and new techniques are introduced. For example, the variance of $\overline{V}+\underline{V}$ can be connected to the variance of $C^{\pi^{k*},\underline{\pi}^k,\rho}$, where $\pi^{k*}$ is the optimal policy towards the adversary policy $\underline{\pi}^k$ with $\pi_h^{k*}(s) = \arg\max_\pi C_h^{\pi,\underline{\pi}^k,\rho}(s)$. Then the variance of $C^{\pi^{k*},\underline{\pi}^k,\rho}$ can be bounded via recursion on the sampled trajectories.

### 6.3.2 Theoretical guarantee

We define the cumulative regret of the output policy $\overline{\pi}^k$ at each episodes $k$ as $Regret(K) := \sum_{k=1}^{K}(V_1^*(s_1^k) - V_1^{\overline{\pi}^k}(s_1^k))$.

**Theorem 20.** For any $\delta \in (0,1]$, letting $\iota = \log(2SAHK/\delta)$, then with probability at least $1-\delta$, Algorithm 6.1 achieves:

- $V_1^*(s_1) - V_1^{\pi^{out}}(s_1) \leq \epsilon$, if the number of episodes $K \geq \Omega(SAH^3\iota^2/\epsilon^2 + S^2AH^3\iota^2/\epsilon)$.
- $Regret(K) = \sum_{k=1}^{K}(V_1^*(s_1^k) - V_1^{\overline{\pi}^k}(s_1^k)) \leq \mathcal{O}(\sqrt{SAH^3K}\iota + S^2AH^3\iota^2)$.

For small $\epsilon \leq H/S$, the sample complexity scales as $\mathcal{O}(SAH^3\iota^2/\epsilon^2)$. For the case with a large number of episodes $K \geq S^3AH^3\iota$, the regret scales as $\mathcal{O}(\sqrt{SAH^3K\iota})$. For the standard MDPs, the information-theoretic sample complexity lower bound is $\Omega(SAH^3/\epsilon^2)$ provided in [118] and the regret lower bound is $\Omega(\sqrt{SAH^3K})$ provided in [41]. When $\rho = 0$, the action robust MDPs is equivalent to the standard MDPs. Thus, the information-theoretic sample complexity lower bound and the regret lower bound of the action robust MDPs should have same dependency on $S$, $A$, $H$, $K$ or $\epsilon$. The lower bounds show the optimality of our algorithm up to logarithmic factors.

## 6.4 Model-free method

In this section, we develop a model-free algorithm, called Action Robust Q-learning with Hoeffding confidence bound (ARQ-H), and analyze its theoretical guarantee. The pseudo code is listed in Algorithm 6.2. Here, we highlight the main idea of Algorithm 6.2. Algorithm 6.2 follows the same idea of Algorithm 6.1, which trains the agent in a clean (simulation) environment and learns a policy that performs well when applied to a perturbed environment with probabilistic policy execution uncertainty. To simulate the action perturbation, Algorithm 6.2 chooses an adversarial action with probability $\rho$. To learn the agent's optimal policy and the corresponding adversarial policy, Algorithm 6.2 computes an optimistic estimate $\overline{Q}$ of $Q^*$ and a pessimistic estimate $\underline{Q}$ of $Q^{\overline{\pi}^k}$. Algorithm 6.2 uses the optimistic estimates to explore the possible optimal policy $\overline{\pi}$ and uses the pessimistic estimates to explore the possible adversarial policy $\underline{\pi}$. The difference is that Algorithm 6.2 use a model-free method to update $Q$ and $V$ values.

**Algorithm 6.2** Action Robust Q-learning with Hoeffding Confidence Bound (ARQ-H)

```
Set $\alpha_t = \frac{H+1}{H+t}$. Initialize $\overline{V}_h(s) = H - h + 1$, $\overline{Q}_h(s,a) = H - h + 1$, $\underline{V}_h(s) = 0$, $\underline{Q}_h(s,a) = 0$, $\hat{r}_h(s,a)$, $N_h(s,a) = 0$ for any state $s \in \mathcal{S}$, any action $a \in \mathcal{A}$ and any step $h \in [H]$. $\overline{V}_{H+1}(s) = \underline{V}_{H+1}(s) = 0$ and $\overline{Q}_{H+1}(s,a) = \underline{Q}_{H+1}(s,a) = 0$ for all $s$ and $a$. $\Delta = H$. Initial policy $\overline{\pi}_h^1(a|s)$ and $\underline{\pi}_h^1(a|s) = 1/A$ for any state $s$, action $a$ and any step $h \in [H]$.
for episode $k = 1, 2, \ldots, K$ do
  for step $h = 1, 2, \ldots, H$ do
    Observe $s_h^k$.
    Set $\overline{a}_h^k = \arg\max_a \overline{Q}_h(s_h^k, a)$ , $\underline{a}_h^k = \arg\min_a \underline{Q}_h(s_h^k, a)$, $\widetilde{\pi}_h^k(\overline{a}_h^k|s_h^k) = 1 - \rho$ and $\widetilde{\pi}_h^k(\underline{a}_h^k|s_h^k) = \rho$.
    Take action $a_h^k \sim \widetilde{\pi}_h^k(\cdot|s_h^k)$.
    Receive reward $r_h^k$ and observe $s_{h+1}^k$.
    Set $t = N_h(s_h^k, a_h^k) \leftarrow N_h(s_h^k, a_h^k) + 1$; $b_t = \sqrt{H^3\iota/t}$.
    $\overline{Q}_h(s_h^k, a_h^k) \leftarrow (1 - \alpha_t)\overline{Q}_h(s_h^k, a_h^k) + \alpha_t(r_h^k + \overline{V}_{h+1}(s_{h+1}^k) + b_t)$,
    $\underline{Q}_h(s_h^k, a_h^k) \leftarrow (1 - \alpha_t)\underline{Q}_h(s_h^k, a_h^k) + \alpha_t(r_h^k + \underline{V}_{h+1}(s_{h+1}^k) - b_t)$.
    Set $\overline{\pi}_h^{k+1}(s_h^k) = \arg\max_a \overline{Q}_h(s_h^k, a)$, $\underline{\pi}_h^{k+1}(s_h^k) = \arg\min_a \underline{Q}_h(s_h^{k+1}, a)$.
    $\overline{V}_h(s_h^k) \leftarrow \min\{\overline{V}_h(s_h^k), (1 - \rho)\overline{Q}_h(s_h^k, \overline{\pi}_h^{k+1}(s_h^k)) + \rho\overline{Q}_h(s_h^k, \underline{\pi}_h^{k+1}(s_h^k))\}$.
    $\underline{V}_h(s_h^k) \leftarrow \max\{\underline{V}_h(s_h^k), (1 - \rho)\underline{Q}_h(s_h^k, \overline{\pi}_h^{k+1}(s_h^k)) + \rho\underline{Q}_h(s_h^k, \underline{\pi}_h^{k+1}(s_h^k))\}$.
    if $\underline{V}_h(s_h^k) > (1 - \rho)\underline{Q}_h(s_h^k, \overline{\pi}_h^{k+1}(s_h^k)) + \rho\underline{Q}_h(s_h^k, \underline{\pi}_h^{k+1}(s_h^k))$ then
      $\overline{\pi}_h^{k+1} = \overline{\pi}_h^k$.
    end if
  end for
  Output policy $\overline{\pi}^{k+1}$ with certificates $\mathcal{I}_{k+1} = [\underline{V}_1(s_1^k), \overline{V}_1(s_1^k)]$ and $\epsilon_{k+1} = |\mathcal{I}_{k+1}|$.
end for
Return $\pi^{out}$
```

Here, we highlight the challenges of the model-free method compared with the model-based method. In the model-based planning, we perform value iteration and the $Q$ values, $V$ values, agent policy $\overline{\pi}$ and adversarial policy $\underline{\pi}$ are updated on all $(s, a)$. However, in the model-free method, the $Q$ values and $V$ values are updated only on $(s_h^k, a_h^k)$ which are the samples on the trajectories. The variances of the $Q$ values and $V$ values in model-free method are larger than the model-based method. Compared with the model-based method, the update of the $Q$ values and $V$ values in the model-free method is slower and less stable.

To deal with this challenges, we design a special update rule of the output policy. In line 14-16, Algorithm 6.2 do not update the output policy until the lower bound on the value function of the new output policy is improved. By this, the output policies are stably improved after every update. The adversary policy is still updated at each episode.

We provide the regret and sample complexity bounds of Algorithm 6.2 in the following:

**Theorem 21.** For any $\delta \in (0, 1]$, letting $\iota = \log(2SABHK/\delta)$, then with probability at least $1-\delta$, Algorithm 6.2 achieves:

- $V_1^*(s_1) - V_1^{\pi^{out}}(s_1) \leq \epsilon$, if the number of episodes $K \geq \Omega(SAH^5\iota/\epsilon^2 + SAH^2/\epsilon)$.
- $Regret(K) = \sum_{k=1}^{K}(V_1^*(s_1^k) - V_1^{\overline{\pi}^k}(s_1^k)) \leq \mathcal{O}(\sqrt{SAH^5K\iota} + SAH^2)$.

The detailed proof is provided in Appendix E.3. The model-free method is more computational efficient than the model-based method but is less sample efficient.

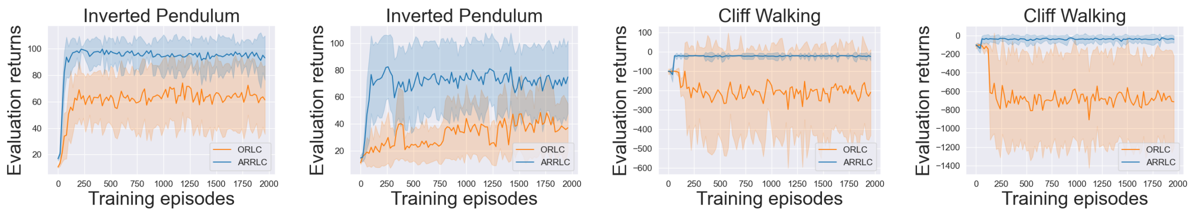
## 6.5 Simulation results

We use OpenAI gym framework [11], and consider two different problems: Cliff Walking, a toy text environment, and Inverted Pendulum, a control environment with the MuJoCo [99] physics simulator. We set $H = 100$. To demonstrate the robustness, the policy is learned in a clean environment, and is then tested on the perturbed environment. Specifically, during the testing, we set a probability $p$ such that after the agent takes an action, with probability $p$, the action is chosen by an adversary. The adversary follows a fixed policy. A Monte-Carlo method is used to evaluate the accumulated reward of the learned policy on the perturbed environment. We take the average over 100 trajectories.

**Inverted pendulum.** The inverted pendulum experiment is a classic control problem in RL. An inverted pendulum is attached by a pivot point to a cart, which is restricted to linear movement in a plane. The cart can be pushed left or right, and the goal is to balance the inverted pendulum on the top of the cart by applying forces on the cart. A reward of $+1$ is awarded for each time step that the inverted pendulum stand upright within a certain angle limit. The fixed adversarial policy in the inverted pendulum environment is a force of $0.5$ N in the left direction.

**Cliff walking.** The cliff walking experiment is a classic scenario proposed in [94]. The game starts with the player at location $[3, 0]$ of the $4 \times 12$ grid world with the goal located at $[3, 11]$. A cliff runs along $[3, 1 - 10]$. If the player moves to a cliff location, it returns to the start location and receives a reward of $-100$. For every move which does not lead into the cliff, the agent receives a reward of $-1$. The player makes moves until they reach the goal. The fixed adversarial policy in the cliff walking environment is walking a step to the bottom.

To show the robustness, we compare our algorithm with a non-robust RL algorithm that is ORLC (Optimistic Reinforcement Learning with Certificates) in [19]. We set $\rho = 0.2$ for our algorithm, which is the uncertain parameter used during the training. In Figure 6.1, we plot the accumulated reward of both algorithms under different $p$. It can be seen that overall our ARRLC algorithm achieves a much higher reward than the ORLC algorithm. This demonstrates the robustness of our ARRLC algorithm to policy execution uncertainty.

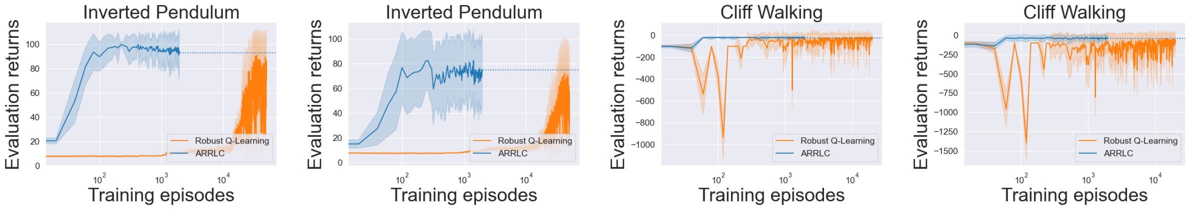


(a) $p = 0.1$ (b) $p = 0.2$ (c) $p = 0.1$ (d) $p = 0.2$

Figure 6.1: ARRLC v.s. ORLC [19]

To show the efficiency, we compare our algorithm with the robust TD algorithm in [45], which can converge to the optimal robust policy but has no theoretical guarantee on sample complexity or regret. We set $\rho = 0.2$. In Figure 6.2, we plot the accumulated reward of both algorithms under different $p$ using a base-10 logarithmic scale on the x-axis and a linear scale on the y-axis. It can be seen that our ARRLC algorithm converges faster than the robust TD algorithm. This demonstrates the efficiency of our ARRLC algorithm to learn optimal policy under policy execution uncertainty.

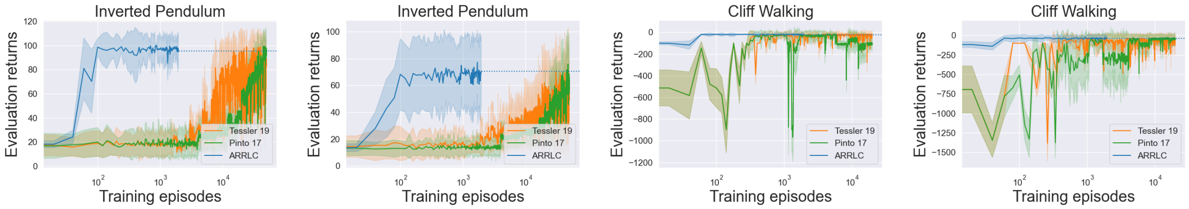
We also compare our algorithm with the approaches in [81, 97] that model the robust



(a) $p = 0.1$ (b) $p = 0.2$ (c) $p = 0.1$ (d) $p = 0.2$

Figure 6.2: ARRLC v.s. Robust TD [45]

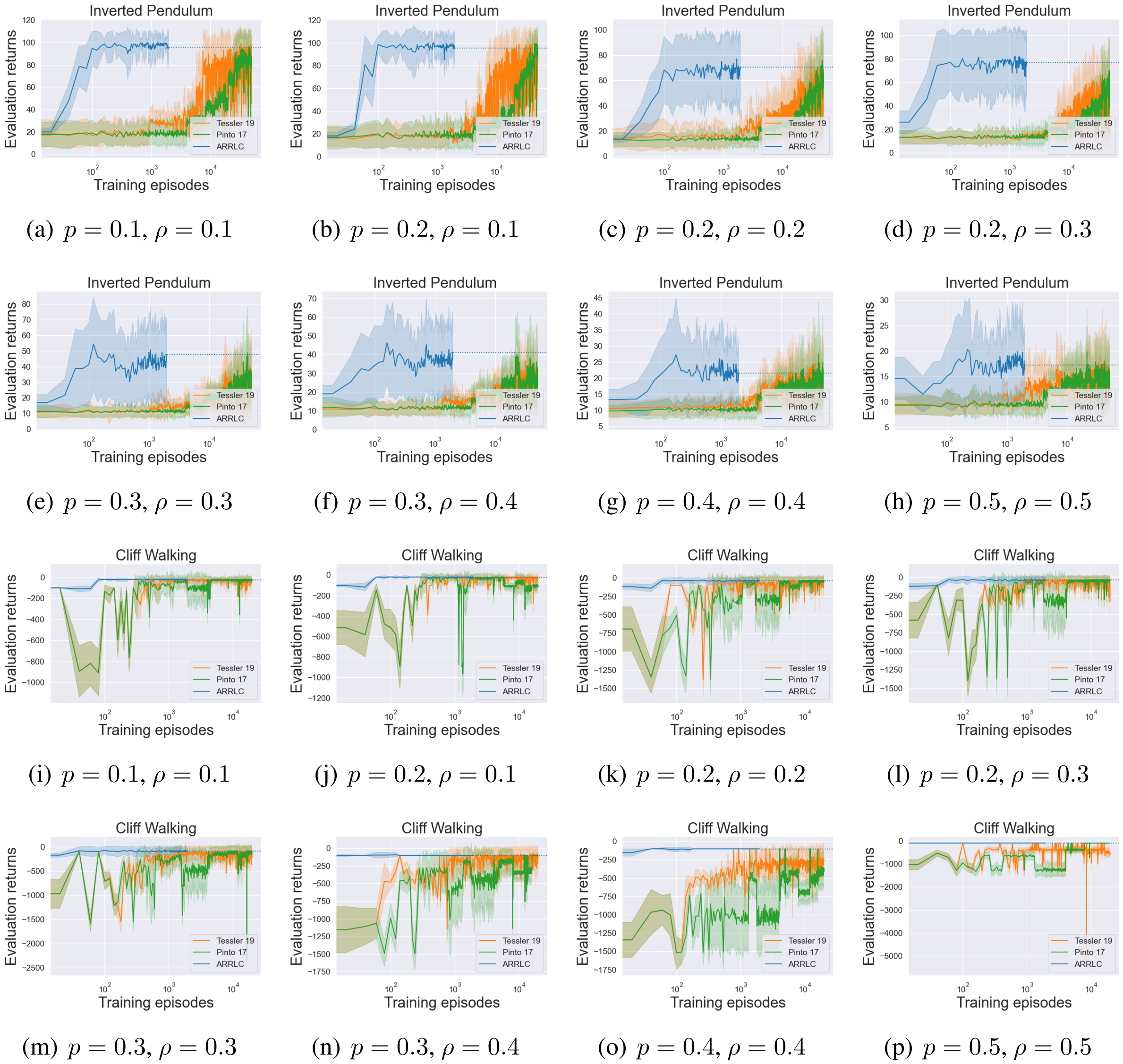
problem as a zero-sum game and alternating update the agent policy and adversary policy. In our implementation, [81] fixes one policy and updates another for 25 episodes, then alternatively updates another in the next 25 episodes. [97] does not alternate the updating until the current policy is converged. Figure 6.3 shows the efficiency of our ARRLC algorithm. ARRLC algorithm is more stable than the other algorithms.



(a) $p = 0.1$ (b) $p = 0.2$ (c) $p = 0.1$ (d) $p = 0.2$

Figure 6.3: ARRLC v.s. PR-PI [97] v.s. RARL [81]

Here, we compare our algorithm with algorithms in [81, 97]. The method in [97] requires an MDP solver to solve the optimal adversarial policy when the agent policy is given and the optimal agent policy when the adversarial policy is given. The white-box MDP solver requires knowledge of the underline MDP so that there is no learning curve and sample complexity discussion in [97]. Thus, we implement the algorithms in [81, 97] with a Q-learning MDP solver, and compared the final evaluation rewards and the learning curve. In addition, we implement the ablation study by setting different $\rho$ and $p$. In our experiments, the policy is learned in a clean environment, and is then tested on the perturbed environment. $\rho$ is the parameter in algorithm when learning the robust policy. $\rho$ can be considered as the agent's guess about the probability of a disturbance occurring. However, $p$ is the probability that the perturb happens in the perturbed environment. In the perturbed environment, with probability $p$, the action is perturbed by an adversarial action.



(a) $p = 0.1$, $\rho = 0.1$ (b) $p = 0.2$, $\rho = 0.1$ (c) $p = 0.2$, $\rho = 0.2$ (d) $p = 0.2$, $\rho = 0.3$

(e) $p = 0.3$, $\rho = 0.3$ (f) $p = 0.3$, $\rho = 0.4$ (g) $p = 0.4$, $\rho = 0.4$ (h) $p = 0.5$, $\rho = 0.5$

(i) $p = 0.1$, $\rho = 0.1$ (j) $p = 0.2$, $\rho = 0.1$ (k) $p = 0.2$, $\rho = 0.2$ (l) $p = 0.2$, $\rho = 0.3$

(m) $p = 0.3$, $\rho = 0.3$ (n) $p = 0.3$, $\rho = 0.4$ (o) $p = 0.4$, $\rho = 0.4$ (p) $p = 0.5$, $\rho = 0.5$

Figure 6.4: ARRLC v.s. RARL v.s. PR-PI

In Figure 6.4, we show the learning curves under different $p$ and $\rho$. It can be seen that our ARRLC algorithm converges faster than the other algorithms. This demonstrates the efficiency of our ARRLC algorithm to learn optimal policy under policy execution uncertainty.

In Figure 6.5, given the agents trained with fixed $\rho$ (rho), we test the agents in different disturbed environments with different $p$. In Figure 6.6, we compare the different agents trained with different

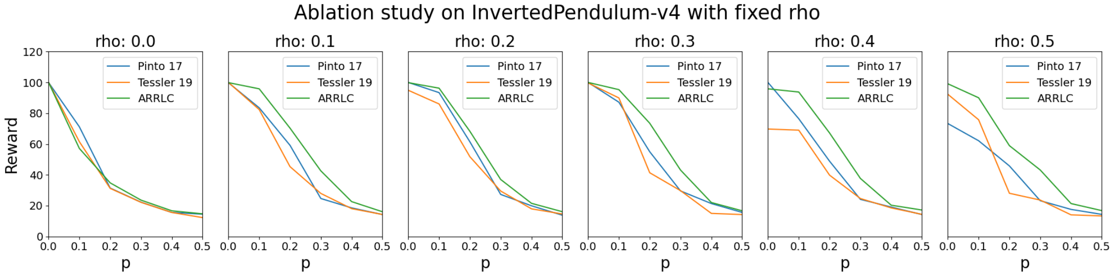


Figure 6.5: Ablation study on InvertedPendulum-v4 with fixed $\rho$.

$\rho$. The x-axis is the different choice of $\rho$ or $p$. The y-axis is the final evaluation rewards.

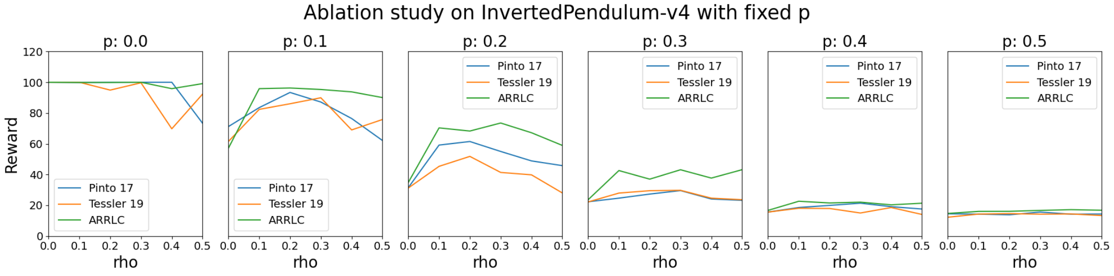
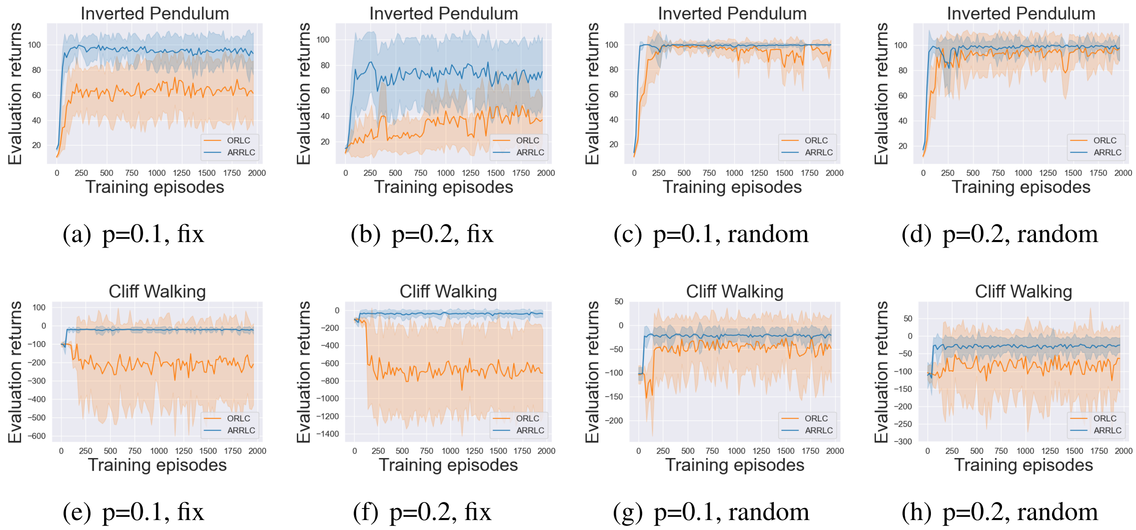


Figure 6.6: Ablation study on InvertedPendulum-v4 with fixed $\rho$.

We also consider different adversary policies include both the fixed policy in the main page and a random adversary policy. After the agent takes an action, with probability $p$, the random adversary will uniformly randomly choose an adversary action to replace the agent's action. In Figure 6.7 and Figure 6.8, ”fix” represents that the actions are perturbed by a fixed adversarial policy during the testing, ”random” represents that the actions are randomly perturbed during the testing, $p$ is the action perturbation probability.

The theoretical guarantee on sample complexity and regret of our algorithm relies on the assumption of known uncertainty parameter. However, in the experimental results shown in Figure 6.5, the parameter can mismatch with the true disturb probability. In Figure 6.7, we test the mismatch of the uncertainty parameter $\rho$ and true uncertainty probability $p$. We train the agent with $\rho = 0.2$, but we use $p = 0.1$ in the test. The proposed robust algorithm still outperforms the non-robust algorithm.

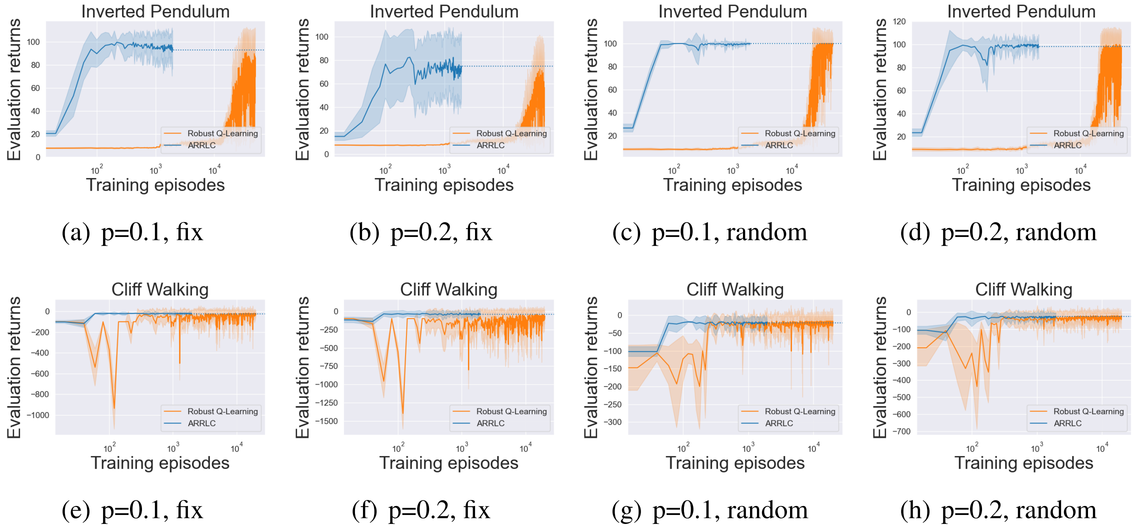
Since we do not know whether the fixed policy or the random policy is the strongest adversary policy against the agent, a more direct comparison is to use the learned worst-case policy in



(a) p=0.1, fix (b) p=0.2, fix (c) p=0.1, random (d) p=0.2, random

(e) p=0.1, fix (f) p=0.2, fix (g) p=0.1, random (h) p=0.2, random

Figure 6.7: ARRLC v.s. ORLC.



(a) p=0.1, fix (b) p=0.2, fix (c) p=0.1, random (d) p=0.2, random

(e) p=0.1, fix (f) p=0.2, fix (g) p=0.1, random (h) p=0.2, random

Figure 6.8: ARRLC v.s. Robust TD

different algorithms to do a cross-comparison. We use the learned worst-case policies to disturb the different robust agents. We report the final evaluation rewards in Table 6.1. We train our method in 2000 episodes and the approaches of [81, 97] in 30000 episodes. We set that $p = \rho = 0.2$. The ARRLC agent performs the best against three different adversaries and the ARRLC adversary impacts the most on three different agents.

Table 6.1: Final rewards under cross-comparison between ARRLC, PR-PI and RAPL

| | ARRLC Adversary | RAPL Adversary | PR-PI Adversary |
|---|---|---|---|
| ARRLC Agent | **72.536** | **81.736** | **89.824** |
| RAPL Agent | 49.936 | 72.216 | 70.6 |
| PR-PI Agent | 52.788 | 63.784 | 86.648 |

## 6.6 Conclusion

In this Chapter, we have developed a novel approach for solving action robust RL problems with probabilistic policy execution uncertainty. We have introduced a model-based algorithm ARRLC and a model-free algorithm ARQ-H. We have theoretically proved the sample complexity bound and the regret bound of the algorithms. The upper bound of the sample complexity and the regret of proposed ARRLC algorithm match the lower bound up to logarithmic factors, which shows the minimax optimality of our algorithm. Moreover, we have carried out numerical experiments to validate our algorithm's robustness and efficiency, revealing that ARRLC surpasses non-robust algorithms and converges more rapidly than the other robust algorithms when faced with action perturbations.

# Chapter 7

# Conclusion

RL has many applicants in a variety of scenarios. The goal of this research is to design RL algorithms that are robust to adversarial attacks. In summary, we have made the following contributions:

Firstly, we have introduced a new class of attacks on stochastic bandits: action-manipulation attacks. We have analyzed the attack against the UCB algorithm and proved that the proposed LCB attack scheme can force the user to almost always pull a non-worst arm with only logarithm effort. To defend against this type of attacks, we have further designed a new bandit algorithm MOUCB that is robust to action-manipulation attacks. We have analyzed the regret of MOUCB under any attack with bounded cost, and have showed that the proposed algorithm is robust to the action-manipulation attacks.

Secondly, we have proposed a class of action poisoning attacks on linear contextual bandits. We have shown that our white-box attack strategy is able to force any linear contextual bandit agent, whose regret scales sublinearly with the total number of rounds, into pulling a target arm chosen by the attacker. We have also shown that our white-box attack strategy can force LinUCB agent into pulling a target arm $T - O(\log^2 T)$ times with attack cost scaled as $O(\log^2 T)$. We have further shown that the proposed blackbox attack strategy can force LinUCB agent into pulling a target arm $T - O(\log^3 T)$ times with attack cost scaled as $T - O(\log^3 T)$.

Thirdly, we have introduced a new class of attacks on RL: action poisoning attacks. We have proposed the $\alpha$-portion white-box attack and the LCB-H black-box attack. We have shown that the $\alpha$-portion white-box attack is able to attack any efficient RL agent and the LCB-H black-box attack nearly matches the performance of the $\alpha$-portion attack. We have analyzed the LCB-H attack against the UCB-H algorithm and proved that the proposed attack scheme can force the agent to almost always follow a particular class of target policy with only logarithm loss and cost.

Fourthly, we have introduced an adversarial attack model on MARL. We have discussed the attack problem in three different settings: the white-box, the gray-box and the black-box settings. We have shown that the power of action poisoning only attacks and reward poisoning only attacks is limited. Even in the white-box setting, there exist some MGs, under which no action poisoning only attack strategy or reward poisoning only attack strategy can be efficient and successful. We have then characterized conditions when action poisoning only attacks or only reward poisoning only attacks can efficiently work. We have further introduced the mixed attack strategy in the gray-box setting that can efficiently attack any sub-linear-regret MARL agents. Finally, we have proposed the approximate mixed attack strategy in the black-box setting and shown its effectiveness on V-learning.

Finally, we have developed a novel approach for solving action robust RL problems with probabilistic policy execution uncertainty. We have theoretically proved the sample complexity bound and the regret bound of the algorithms. The upper bound of the sample complexity and the regret of proposed ARRLC algorithm match the lower bound up to logarithmic factors, which shows the minimax optimality of our algorithm.

# Appendix A

# Appendix of Chapter 2

## A.1 Attack Cost Analysis of LCB attack strategy

### A.1.1 Proof of Lemma 2

The proof is similar with the proof of Lemma 1 that was proved in [43]. Let $\{X_j\}_{j=1}^{\infty}$ be a sequence of $i.i.d$ $\sigma^2$-sub-Gaussian random variables with mean $\mu$. Let $\hat{\mu}^0(t) = \frac{1}{N(t)} \sum_{j=1}^{N(t)} X_j$. By Hoeffding's inequality.

$$\mathbb{P}(|\hat{\mu}^0(t) - \mu| \geq \eta) \leq 2\exp\left(-\frac{N(t)\eta^2}{2\sigma^2}\right). \tag{A.1}$$

In order to ensure that $\mathcal{E}_2$ holds for all arm $i$, all arm $j$ and all pull counts $N = N_{i,j}(t)$, we set $\delta_{i,j,N} := \frac{6\delta}{\pi^2 K^2 N^2}$. We have

$$\mathbb{P}\left(\exists i, \exists j, \exists N : |\hat{\mu}_{i,j}(t) - \mu_j| \geq \sqrt{\frac{2\sigma^2}{N}\log\frac{\pi^2 K^2 N^2}{3\delta}}\right) \leq \sum_{i=1}^{K}\sum_{j=1}^{K}\sum_{N=1}^{\infty}\delta_{i,j,N} = \delta. \tag{A.2}$$

### A.1.2 Proof of Lemma 3

According to event $\mathcal{E}_2$, we have

$$\begin{aligned}
&\left|\hat{\mu}_i(t) - \frac{1}{N_i(t)} \sum_{s\in\tau_i(t)} \mu_{I_s^0}\right| \\
=&\left|\sum_{j=1}^{K} \frac{N_{i,j}(t)}{N_i(t)} \left(\hat{\mu}_{i,j}(t) - \mu_j\right)\right| \\
\leq&\sum_{j=1}^{K} \frac{N_{i,j}(t)}{N_i(t)} \left|\hat{\mu}_{i,j}(t) - \mu_j\right| \\
<&\frac{1}{N_i(t)} \sum_{j=1}^{K} \sqrt{2\sigma^2 N_{i,j}(t) \log \frac{\pi^2 K^2 (N_{i,j}(t))^2}{3\delta}}.
\end{aligned} \tag{A.3}$$

Define a function $f(N) = \sqrt{2\sigma^2 N \log \frac{\pi^2 K^2 N^2}{3\delta}} : (0, +\infty) \to \mathbb{R}$, and we have

$$\begin{aligned}
f''(N) =&\frac{\partial^2}{\partial N^2} \sqrt{2\sigma^2 N \log \frac{\pi^2 K^2 N^2}{3\delta}} \\
=&-\frac{\left(2\sigma^2 \log \frac{\pi^2 K^2 N^2}{3\delta}\right)^2 + 16\sigma^4}{4\left(2\sigma^2 N \log \frac{\pi^2 K^2 N^2}{3\delta}\right)^{\frac{3}{2}}} \\
<&0,
\end{aligned} \tag{A.4}$$

when $N \geq 1$.

Hence $f$ is strictly concave when $N \geq 1$, and according to the property of the concave function,

$$\sum_{j=1}^{K} f(N_{i,j}(t)) < Kf\left(\frac{1}{K}\sum_{j=1}^{K} N_{i,j}(t)\right) = Kf\left(\frac{N_i(t)}{K}\right). \tag{A.5}$$

Thus,

$$
\begin{aligned}
&\left|\hat{\mu}_i(t) - \frac{1}{N_i(t)} \sum_{s\in\tau_i(t)} \mu_{I_s^0}\right| \\
<&\frac{1}{N_i(t)} K \sqrt{2\sigma^2 \frac{N_i(t)}{K} \log \frac{\pi^2 K^2 \left(\frac{N_i(t)}{K}\right)^2}{3\delta}} \\
=&\sqrt{\frac{2\sigma^2 K}{N_i(t)} \log \frac{\pi^2 (N_i(t))^2}{3\delta}}.
\end{aligned}
\tag{A.6}
$$

### A.1.3 Proof of Lemma 4

The LCB attack scheme uses lower confidence bound to exploit the worst arm, so we need to prove that the attacker's pull counts of all non-worst arms should be limited at round $t$.

Consider the case that in round $t+1$, the user chooses a non-target arm $I_{t+1} = i \neq K$ and the attacker changes it to a non-worst arm $I_{t+1}^0 = j \neq i_W$. On one hand, under event $\mathcal{E}_1$, we have

$$
\begin{aligned}
\hat{\mu}_{i_W}^0(t) - \mu_{i_W} &< \mathbf{CB}(N_{i_W}^0(t), \delta), \\
\text{and } \hat{\mu}_j^0(t) - \mu_j &> -\mathbf{CB}(N_j^0(t), \delta).
\end{aligned}
\tag{A.7}
$$

On the other hand, according to the attack scheme, it must be the case that

$$
\hat{\mu}_{i_W}^0(t) - \mathbf{CB}(N_{i_W}^0(t), \delta) > \hat{\mu}_j^0(t) - \mathbf{CB}(N_j^0(t), \delta), \tag{A.8}
$$

which is equivalent to

$$
\mathbf{CB}(N_j^0(t), \delta) > \hat{\mu}_j^0(t) - (\hat{\mu}_{i_W}^0(t) - \mathbf{CB}(N_{i_W}^0(t), \delta)). \tag{A.9}
$$

Combining (A.9) with (A.7), we have

$$\begin{aligned} \mathbf{CB}(N_j^0(t),\delta) &> \mu_j - \mathbf{CB}(N_j^0(t),\delta) - \mu_{i_W}, \\ \text{and } \mathbf{CB}(N_j^0(t),\delta) &> \frac{\Delta_{j,i_W}}{2}. \end{aligned} \tag{A.10}$$

Using the fact that $N_j^0(t) \leq t$ and $N_{i,j}(t) \leq N_j^0(t)$, we have

$$\begin{aligned} \frac{\Delta_{j,i_W}}{2} <& \mathbf{CB}(N_j^0(t),\delta) \\ =& \sqrt{\frac{2\sigma^2}{N_j^0(t)} \log \frac{\pi^2 K (N_j^0(t))^2}{3\delta}} \\ \leq& \sqrt{\frac{2\sigma^2}{N_j^0(t)} \log \frac{\pi^2 K t^2}{3\delta}} \\ \leq& \sqrt{\frac{2\sigma^2}{N_{i,j}(t)} \log \frac{\pi^2 K t^2}{3\delta}}, \end{aligned} \tag{A.11}$$

which is equivalent to

$$N_{i,j}(t) < \frac{8\sigma^2}{\Delta_{j,i_W}^2} \log \frac{\pi^2 K t^2}{3\delta}. \tag{A.12}$$

Hence, under event $\mathcal{E}_2$, we have

$$\begin{aligned} \hat{\mu}_i(t) <& \frac{1}{N_i(t)} \sum_{s\in\tau_i(t)} \mu_{I_s^0} + \sqrt{\frac{2\sigma^2 K}{N_i(t)} \log \frac{\pi^2 (N_i(t))^2}{3\delta}} \\ =& \frac{1}{N_i(t)} \sum_j \sum_{s\in\tau_{i,j}(t)} \mu_{I_s^0} + \sqrt{\frac{2\sigma^2 K}{N_i(t)} \log \frac{\pi^2 (N_i(t))^2}{3\delta}} \\ =& \frac{1}{N_i(t)} \sum_j N_{i,j}(t)\mu_j + \sqrt{\frac{2\sigma^2 K}{N_i(t)} \log \frac{\pi^2 (N_i(t))^2}{3\delta}} \\ =& \sum_j \frac{N_{i,j}(t)}{N_i(t)} (\Delta_{j,i_W} + \mu_{i_W}) + \sqrt{\frac{2\sigma^2 K}{N_i(t)} \log \frac{\pi^2 (N_i(t))^2}{3\delta}} \\ <& \mu_{i_W} + \sqrt{\frac{2\sigma^2 K}{N_i(t)} \log \frac{\pi^2 (N_i(t))^2}{3\delta}} + \frac{1}{N_i(t)} \sum_{j\neq i_W} \frac{8\sigma^2}{\Delta_{j,i_W}} \log \frac{\pi^2 K t^2}{3\delta}. \end{aligned} \tag{A.13}$$

The lemma is proved.

### A.1.4 Proof of Theorem 1

By inferring from Lemma 1, we have that with probability $1 - \frac{\delta}{K}$, $\forall t > K : |\hat{\mu}_K^0(t) - \mu_K| < \mathbf{CB}(N_K^0(t), \delta)$.

Because the LCB attack scheme does not attack the target arm, we can also conclude that with probability $1 - \frac{\delta}{K}$, $\forall t > K : |\hat{\mu}_K(t) - \mu_K| < \mathbf{CB}(N_K(t), \delta)$.

The user relies on the UCB algorithm to choose arms. If at round $t$, the user chooses an arm $I_t = i \neq K$, which is not the target arm, we have

$$\hat{\mu}_i(t-1) + 3\sigma\sqrt{\frac{\log t}{N_i(t-1)}} > \hat{\mu}_K(t-1) + 3\sigma\sqrt{\frac{\log t}{N_K(t-1)}}, \tag{A.14}$$

which is equivalent to

$$3\sigma\sqrt{\frac{\log t}{N_i(t-1)}} > -\hat{\mu}_i(t-1) + \hat{\mu}_K(t-1) + 3\sigma\sqrt{\frac{\log t}{N_K(t-1)}}. \tag{A.15}$$

We need to connect the estimate of arms to the true means. Under event $\mathcal{E}_1$, we have

$$\hat{\mu}_K(t) > \mu_K - \mathbf{CB}(N_K(t), \delta). \tag{A.16}$$

Under event $\mathcal{E}_1 \cap \mathcal{E}_2$, according to Lemma 4, we have

$$\hat{\mu}_i(t) \leq \mu_{i_W} + \sqrt{\frac{2\sigma^2 K}{N_i(t)} \log\frac{\pi^2(N_i(t))^2}{3\delta}} + \frac{1}{N_i(t)}\sum_{j\neq i_W}\frac{8\sigma^2}{\Delta_{j,i_W}}\log\frac{\pi^2 K t^2}{3\delta}. \tag{A.17}$$

Combing the inequalities above,

$$\begin{aligned}3\sigma\sqrt{\frac{\log t}{N_i(t-1)}} > -\mu_{i_W} - &\sqrt{\frac{2\sigma^2 K}{N_i(t-1)}\log\frac{\pi^2(N_i(t-1))^2}{3\delta}} \\ &- \frac{1}{N_i(t-1)}\sum_{j\neq i_W}\frac{8\sigma^2}{\Delta_{j,i_W}}\log\frac{\pi^2 K(t-1)^2}{3\delta} + \\ &\mu_K - \mathbf{CB}(N_K(t-1),\delta) + 3\sigma\sqrt{\frac{\log t}{N_K(t-1)}}.\end{aligned} \tag{A.18}$$

The sum of the last two terms in the RHS of (A.18) is equal or larger than zero. We show it by further bounding the last term as follows: when $t \geq \left(\frac{\pi^2 K}{3\delta}\right)^{\frac{2}{5}}$,

$$\begin{aligned}3\sigma\sqrt{\frac{\log t}{N_K(t-1)}} &\geq \sqrt{4\sigma^2\frac{\log t}{N_K(t-1)} + 5\sigma^2\frac{\log(\frac{\pi^2 K}{3\delta})^{\frac{2}{5}}}{N_K(t-1)}} \\ &\geq \sqrt{2\sigma^2\frac{\log\frac{\pi^2 K t^2}{3\delta}}{N_K(t-1)}} \\ &\geq \sqrt{2\sigma^2\frac{\log\frac{\pi^2 K(N_K(t-1))^2}{3\delta}}{N_K(t-1)}} \\ &= \mathbf{CB}(N_K(t-1),\delta).\end{aligned} \tag{A.19}$$

Now the inequality only depends on $N_i(t-1)$ and some constants:

$$\begin{aligned}&3\sigma\sqrt{\frac{\log t}{N_i(t-1)}} \\ >&\Delta_{K,i_W} - \sqrt{\frac{2\sigma^2 K}{N_i(t-1)}\log\frac{\pi^2(N_i(t-1))^2}{3\delta}} - \frac{1}{N_i(t-1)}\sum_{j\neq i_W}\frac{8\sigma^2}{\Delta_{j,i_W}}\log\frac{\pi^2 K(t-1)^2}{3\delta} \\ >&\Delta_{K,i_W} - \sqrt{\frac{2\sigma^2 K}{N_i(t-1)}\log\frac{\pi^2 t^2}{3\delta}} - \frac{1}{N_i(t-1)}\sum_{j\neq i_W}\frac{8\sigma^2}{\Delta_{j,i_W}}\log\frac{\pi^2 K t^2}{3\delta}.\end{aligned} \tag{A.20}$$

The last inequality is based on the fact that $N_i(t-1) < t$. By solving the inequality above, we have:

$$N_i(t-1) < \frac{1}{4\Delta_{K,i_W}^2}\left(C_1 + \left(C_1^2 + 4\Delta_{K,i_W}\sum_{j\neq i_W}\frac{8\sigma^2}{\Delta_{j,i_W}}\log\frac{\pi^2 K t^2}{3\delta}\right)^{\frac{1}{2}}\right)^2, \tag{A.21}$$

where $C_1 = 3\sigma\sqrt{\log t} + \sqrt{2\sigma^2 K \log\frac{\pi^2 t^2}{3\delta}}$. Since event $\mathcal{E}_1 \cap \mathcal{E}_2$ occurs with probability at least $1-2\delta$, we have that (A.21) holds with probability at least $1-2\delta$. Theorem 1 follows immediately from the definition of the attack cost and (A.21).

### A.1.5 Proof of Theorem 2

Because the target arm is the worst arm, the mean rewards of all arms are larger than or equal to that of the target arm. Thus, for any attack scheme, we have

$$\frac{1}{N_i(t)}\sum_{s\in\tau_i(t)}\mu_{I_s^0} \geq \mu_K. \tag{A.22}$$

If the user pulls arm $K$ at round $t$, according to UCB algorithm, we have for the optimal arm $i_O \neq K$,

$$\hat{\mu}_{i_O}(t-1) + 3\sigma\sqrt{\frac{\log t}{N_{i_O}(t-1)}} < \hat{\mu}_K(t-1) + 3\sigma\sqrt{\frac{\log t}{N_K(t-1)}}. \tag{A.23}$$

Under event $\mathcal{E}_2$, we have Lemma 3 and (A.6) holds for all arm $i$, which implies

$$\hat{\mu}_{i_O}(t-1) > \frac{1}{N_{i_O}(t-1)}\sum_{s\in\tau_{i_O}(t-1)}\mu_{I_s^0} - \sqrt{\frac{2\sigma^2 K}{N_{i_O}(t-1)}\log\frac{\pi^2(N_{i_O}(t-1))^2}{3\delta}}, \tag{A.24}$$

and

$$\hat{\mu}_K(t-1) < \frac{1}{N_K(t-1)}\sum_{s\in\tau_K(t-1)}\mu_{I_s^0} + \sqrt{\frac{2\sigma^2 K}{N_K(t-1)}\log\frac{\pi^2(N_K(t-1))^2}{3\delta}}. \tag{A.25}$$

Noted that for $\delta > \frac{1}{2}$, when $\delta$ is fixed, **CB** $\left(\frac{N}{K}, \frac{\delta}{K}\right) = \sqrt{2\sigma^2 \frac{N}{K} \log \frac{\pi^2 N^2}{3\delta}} : (0, +\infty) \to \mathbb{R}$ is monotonically decreasing in $N \geq 1$.

We aim to prove that the total number of non-target arms pull scales as $T$. We divide the problem into three different cases.

Firstly, if $N_i(t-1) \geq \frac{1}{16} N_K(t-1)$, Theorem 2 holds.

Secondly, if $N_{i_O}(t-1) < \frac{1}{16} N_K(t-1)$ and $N_{i_O}(t-1) < \frac{\sqrt{3\delta}}{\pi} t^{\frac{9}{64K}}$ hold for the optimal arm $i_O$, we have

$$3\sigma \sqrt{\frac{\log t}{N_K(t-1)}} < \frac{3}{4}\sigma \sqrt{\frac{\log t}{N_{i_O}(t-1)}}, \tag{A.26}$$

and

$$\begin{aligned}
&\sqrt{\frac{2\sigma^2 K}{N_K(t-1)} \log \frac{\pi^2 (N_K(t-1))^2}{3\delta}} \\
<&\sqrt{\frac{2\sigma^2 K}{N_{i_O}(t-1)} \log \frac{\pi^2 (N_{i_O}(t-1))^2}{3\delta}} \\
<&\frac{3}{4}\sigma \sqrt{\frac{\log t}{N_{i_O}(t-1)}}.
\end{aligned} \tag{A.27}$$

Combining the inequalities above, we find

$$\begin{aligned}
\frac{1}{N_K(t-1)} \sum_{s \in \tau_K(t-1)} \mu_{I_s^0} - \mu_K &> \frac{3}{4}\sigma \sqrt{\frac{\log t}{N_{i_O}(t-1)}} \\
&> \frac{3}{4}\sigma \sqrt{\frac{\pi \log t}{\sqrt{3\delta} t^{\frac{9}{64K}}}} \\
&> \frac{3}{4}\sigma \sqrt{\frac{\pi \log t}{\sqrt{3\delta} t}}.
\end{aligned} \tag{A.28}$$

The RHS of (A.28) is monotonically decreasing in $t \geq 3$, so $\frac{3}{4}\sigma \sqrt{\frac{\pi \log t}{\sqrt{3\delta} t}} > \frac{3}{4}\sigma \sqrt{\frac{\pi \log T}{\sqrt{3\delta} T}}$.

Since the attack cost is limited by $\mathcal{O}(\log T)$,

$$\frac{1}{N_K(t-1)} \sum_{s \in \tau_K(t-1)} \mu_{I_s^0} - \mu_K = \frac{\mathcal{O}(\log T)}{N_K(t-1)}, \tag{A.29}$$

so

$$N_K(t-1) = O\left(\sqrt{T \log T}\right), \tag{A.30}$$

in which Theorem 2 holds.

Thirdly, if $N_{i_O}(t-1) < \frac{1}{16} N_K(t-1)$ and $N_{i_O}(t-1) \geq \frac{\sqrt{3\delta}}{\pi} t^{\frac{9}{64K}}$ hold for the optimal arm $i_O$, we have

$$3\sigma \sqrt{\frac{\log t}{N_K(t-1)}} < 3\sigma \sqrt{\frac{\log t}{N_{i_O}(t-1)}}, \tag{A.31}$$

and (A.23) is equivalent to

$$\hat{\mu}_{i_O}(t-1) < \hat{\mu}_K(t-1). \tag{A.32}$$

Setting the number of attacks on the optimal arm as $C_{i_O}$ and the number of attacks on the target arm as $C_K$, we have

$$\frac{1}{N_{i_O}(t-1)} \sum_{s \in \tau_{i_O}(t-1)} \mu_{I_s^0} \geq \mu_{i_O} - \frac{C_{i_O}}{N_{i_O}(t-1)} \Delta_{i_O,K}, \tag{A.33}$$

and

$$\frac{1}{N_K(t-1)} \sum_{s \in \tau_K(t-1)} \mu_{I_s^0} \leq \mu_K + \frac{C_K}{N_K(t-1)} \Delta_{i_O,K}, \tag{A.34}$$

Thus, the inequality (A.32) becomes

$$\begin{aligned}&\mu_{i_O} - \frac{C_{i_O}}{N_{i_O}(t-1)}\Delta_{i_O,K} - \textbf{CB}\left(\frac{N_{iO}(t-1)}{K}, \frac{\delta}{K}\right)\\ <&\mu_K + \frac{C_K}{N_K(t-1)}\Delta_{i_O,K} + \textbf{CB}\left(\frac{N_K(t-1)}{K}, \frac{\delta}{K}\right).\end{aligned} \tag{A.35}$$

Because $N_{i_O}(t-1) < \frac{1}{16}N_K(t-1) < N_K(t-1)$, we have $\textbf{CB}\left(\frac{N_{iO}(t-1)}{K}, \frac{\delta}{K}\right) > \textbf{CB}\left(\frac{N_K(t-1)}{K}, \frac{\delta}{K}\right)$.

From (A.35), we have

$$\begin{aligned}&\frac{C_K}{N_K(t-1)}\Delta_{i_O,K}\\ >&\Delta_{i_O,K} - \frac{C_{i_O}}{N_{i_O}(t-1)}\Delta_{i_O,K} - 2\textbf{CB}\left(\frac{N_{i_O}(t-1)}{K}, \frac{\delta}{K}\right)\\ >&\Delta_{i_O,K} - \frac{C_{i_O}}{N_{i_O}(t-1)}\Delta_{i_O,K} - 2\sqrt{\frac{2\sigma^2 K}{N_{i_O}(t-1)}\log\frac{\pi^2(N_{i_O}(t-1))^2}{3\delta}}.\end{aligned} \tag{A.36}$$

Here, based on $N_{i_O}(t-1) \geq \frac{\sqrt{3\delta}}{\pi}t^{\frac{9}{64K}}$ and the fact $t \geq N_K(t-1)$,

$$\begin{aligned}&\frac{C_K}{N_K(t-1)}\Delta_{i_O,K}\\ >&\Delta_{i_O,K} - \frac{C_{i_O}}{\frac{\sqrt{3\delta}}{\pi}t^{\frac{9}{64K}}}\Delta_{i_O,K} - 2\sqrt{\frac{2\sigma^2 K}{\frac{\sqrt{3\delta}}{\pi}t^{\frac{9}{64K}}}\log\frac{\pi^2(\frac{\sqrt{3\delta}}{\pi}t^{\frac{9}{64K}})^2}{3\delta}}\\ >&\Delta_{i_O,K} - \frac{C_{i_O}}{\frac{\sqrt{3\delta}}{\pi}\left(N_K(t-1)\right)^{\frac{9}{64K}}}\Delta_{i_O,K} - 2\sqrt{\frac{2\sigma^2 K}{\frac{\sqrt{3\delta}}{\pi}\left(N_K(t-1)\right)^{\frac{9}{64K}}}\log\frac{\pi^2(\frac{\sqrt{3\delta}}{\pi}t^{\frac{9}{64K}})^2}{3\delta}}.\end{aligned} \tag{A.37}$$

Since the attack cost is limited by $\mathcal{O}(\log T)$,

$$N_K(t-1) = O((\log T)^{\frac{64K}{9}}), \tag{A.38}$$

and Theorem 2 holds.

In summary, all cases show that the user pulls the non-target arm more than $\mathcal{O}(T^{\alpha})$ times, in which $\alpha < 1$. Since event $\mathcal{E}_2$ holds with probability at least $1-\delta$, the conclusion in the theorem holds with probability at least $1-\delta$.

### A.1.6 Proof of Proposition 1

The oracle attack needs to occasionally change the action to the best arm when the user pulls the target arm. Similar to Lemma 3, under event $\mathcal{E}_2$, for arm $K$ and all $t > K$, we have

$$\left| \hat{\mu}_K(t) - \frac{1}{N_K(t)} \sum_{s \in \tau_K(t)} \mu_{I_s^0} \right| < \mathbf{CB}\left( \frac{N_K(t)}{2}, \frac{\delta}{K} \right), \tag{A.39}$$

because when the user pulls the target arm, the rewards the user observes are only drawn from two distributions.

Given the number of rounds that the attacker changes the action to the best arm as $C_K$, we have

$$\frac{1}{N_K(t)} \sum_{s \in \tau_K(t)} \mu_{I_s^0} \le \mu_K + \frac{C_K}{N_K(t)} \Delta_{i_O, K}, \tag{A.40}$$

in which the equality holds when $N_K(t) \ge C_K$.

The user relies on the UCB algorithm to choose arms. We denote the last round when the user chooses the target arm before round $T$ as $t$. At round $t$, the user chooses the target arm $I_t = K$. For any non-target arm $i$, we have

$$\hat{\mu}_i(t-1) + 3\sigma \sqrt{\frac{\log t}{N_i(t-1)}} \le \hat{\mu}_K(t-1) + 3\sigma \sqrt{\frac{\log t}{N_K(t-1)}}. \tag{A.41}$$

We focus on the last term of the RHS of (A.41). When $t \ge \left( \frac{\pi^2 K^2}{12\delta} \right)^4$, we have

$$3\sigma \sqrt{\frac{\log t}{N_K(t)}} \ge \sqrt{\frac{4\sigma^2}{N_K(t)} \log \frac{\pi^2 K^2 t^2}{12\delta}} \ge \mathbf{CB}\left( \frac{N_K(t)}{2}, \frac{\delta}{K} \right). \tag{A.42}$$

Thus, the RHS of (A.41) can be further bounded as:

$$\hat{\mu}_K(t-1) + 3\sigma\sqrt{\frac{\log t}{N_K(t-1)}} \leq \mu_K + \frac{C_K}{N_K(t-1)}\Delta_{i_O,K} + 6\sigma\sqrt{\frac{\log t}{N_K(t-1)}}. \tag{A.43}$$

Similar to (A.42), when $t \geq \frac{\pi^2 K^2}{3\delta}$,

$$\frac{5}{2}\sigma\sqrt{\frac{\log t}{N_i(t)}} > \sqrt{\frac{2\sigma^2}{N_i(t)}\log\frac{\pi^2 K t^2}{3\delta}} \geq \mathbf{CB}\left(N_i(t), \frac{\delta}{K}\right). \tag{A.44}$$

The oracle attack changes every non-target arm to the worst arm. Using Lemma 1, we have that with probability $1 - \frac{\delta(K-1)}{K}$, $\forall t > K$ and $i \neq K : |\hat{\mu}_i(t) - \mu_K| < \mathbf{CB}(N_i(t), \delta)$.

Then, by combing (A.43) and (A.44), (A.41) is equivalent to:

$$\mu_K + \frac{1}{2}\sigma\sqrt{\frac{\log t}{N_i(t-1)}} < \mu_K + \frac{C_K}{N_K(t-1)}\Delta_{i_O,K} + 6\sigma\sqrt{\frac{\log t}{N_K(t-1)}}. \tag{A.45}$$

If the attacker does not attack the target arm, all arms are changed to the worst arm. Thus, at round $t$, the expectation of the target arm pull counts would be $\frac{t}{K}$. Here, we divide the problem into two cases: $N_K(t-1) \geq \frac{T}{K}$ and $N_K(t-1) < \frac{T}{K}$.

If $N_K(t-1) \geq \frac{T}{K}$, from (A.45), we have

$$\frac{1}{2}\sigma\sqrt{\frac{\log t}{N_i(t-1)}} < \frac{KC_K}{T}\Delta_{i_O,K} + 6\sigma\sqrt{\frac{K\log t}{T}}, \tag{A.46}$$

which is equivalent to

$$N_i(t-1) > \frac{\sigma^2 T^2 \log t}{4\left(KC_K\Delta_{i_O,K} + 6\sigma\sqrt{KT\log t}\right)^2}. \tag{A.47}$$

Since equation (A.47) is monotonically increasing in $t \geq 1$ and the fact that $t > \frac{T}{K}$,

$$N_i(t-1) > \frac{\sigma^2 T^2 \log \frac{T}{K}}{4\left(KC_K\Delta_{i_O,K} + 6\sigma\sqrt{KT\log\frac{T}{K}}\right)^2}. \tag{A.48}$$

If $N_K(t-1) < \frac{T}{K}$, the attack cost $|\mathcal{C}| > \frac{T(K-1)}{K} + C_K$ and $\sum_{i \neq K} N_i(t-1) > \frac{T(K-1)}{K}$. Combining the two cases, the proof is completed.

## A.2 Regret Analysis of MOUCB

### A.2.1 Proof of Lemma 5

Note that for $\delta \leq \frac{1}{3}$, $\beta(N) = \sqrt{\frac{2\sigma^2 K}{N}\log\frac{\pi^2 N^2}{3\delta}}$ is monotonically decreasing in $N$, as

$$\frac{\partial}{\partial N}\beta^2(N) = \frac{2\sigma^2 K}{N^2}\left(2 - \log\frac{\pi^2 N^2}{3\delta}\right) \leq \frac{2\sigma^2 K}{N^2}\left(2 - \log\frac{\pi^2}{3\delta}\right) < 0. \tag{A.49}$$

We first prove the first inequality in Lemma 5. Consider the optimal arm $i_O$ and the worst arm $i_W$. Define $C_i := |\{t : t \leq T, I_t^0 \neq I_t = i\}|$. In the action-manipulation setting, when $t > 2AK$, MOUCB algorithm has

$$\begin{aligned} \frac{1}{N_{i_O}(t)} \sum_{s \in \tau_{i_O}(t)} \mu_{I_s^0} &\geq \frac{N_{i_O}(t) - C_{i_O}}{N_{i_O}(t)}\mu_{i_O} + \frac{C_{i_O}}{N_{i_O}(t)}\mu_{i_W} \\ &= \mu_{i_O} - \Delta_{i_O,i_W}\frac{C_{i_O}}{N_{i_O}(t)} \\ &\geq \mu_{i_O} - \Delta_{i_O,i_W}\frac{C_{i_O}}{2A}, \end{aligned} \tag{A.50}$$

and

$$\begin{aligned}\frac{1}{N_{i_W}(t)}\sum_{s\in\tau_{i_W}(t)}\mu_{I_s^0} &\leq \frac{N_{i_W}(t)-C_{i_W}}{N_{i_W}(t)}\mu_{i_W}+\frac{C_{i_W}}{N_{i_W}(t)}\mu_{i_O}\\ &=\mu_{i_W}+\Delta_{i_O,i_W}\frac{C_{i_W}}{N_{i_W}(t)}\\ &\leq \mu_{i_W}+\Delta_{i_O,i_W}\frac{C_{i_W}}{2A}.\end{aligned} \tag{A.51}$$

Combining (A.50) and (A.51), we have

$$\begin{aligned}&\frac{1}{N_{i_O}(t)}\sum_{s\in\tau_{i_O}(t)}\mu_{I_s^0}-\frac{1}{N_{i_W}(t)}\sum_{s\in\tau_{i_W}(t)}\mu_{I_s^0}\\ \geq&\mu_{i_O}-\mu_{i_W}-\Delta_{i_O,i_W}\frac{C_{i_W}}{2A}-\Delta_{i_O,i_W}\frac{C_{i_O}}{2A}\\ \geq&\mu_{i_O}-\mu_{i_W}-\Delta_{i_O,i_W}\frac{A}{2A}\\ =&\frac{\Delta_{i_O,i_W}}{2}.\end{aligned} \tag{A.52}$$

From (A.6), we could find

$$\begin{aligned}&\frac{1}{N_{i_O}(t)}\sum_{s\in\tau_{i_O}(t)}\mu_{I_s^0}-\frac{1}{N_{i_W}(t)}\sum_{s\in\tau_{i_W}(t)}\mu_{I_s^0}\\ \leq&\hat{\mu}_{i_O}(t)+\beta(N_{i_O}(t))-(\hat{\mu}_{i_W}(t)-\beta(N_{i_W}(t)))\\ \leq&\max_{i,j}\left\{\hat{\mu}_i(t)+\beta(N_i(t))-(\hat{\mu}_j(t)-\beta\left(N_j(t)\right))\right\}.\end{aligned} \tag{A.53}$$

We now prove the second inequality in Lemma 5:

$$\begin{aligned}&\max_{i,j}\left\{\hat{\mu}_i(t)+\beta(N_i(t))-(\hat{\mu}_j(t)-\beta(N_j(t)))\right\}\\ \leq&\max_{i,j}\left\{\frac{1}{N_i(t)}\sum_{s\in\tau_i(t)}\mu_{I_s^0}+2\beta(N_i(t))-\left(\frac{1}{N_j(t)}\sum_{s\in\tau_j(t)}\mu_{I_s^0}-2\beta(N_j(t))\right)\right\}\\ \leq&\Delta_{i_O,i_W}+\max_{i,j}\left\{2\beta(N_i(t))+2\beta(N_j(t))\right\}.\end{aligned} \tag{A.54}$$

Recall that for $\delta \le \frac{1}{3}$, $\beta(N) = \sqrt{\frac{2\sigma^2 K}{N}\log\frac{\pi^2 N^2}{3\delta}}$ is monotonically decreasing in $N$. Therefore,

$$\max_{i,j}\left\{2\beta(N_i(t)) + 2\beta(N_j(t))\right\} \le 4\beta(2A). \tag{A.55}$$

### A.2.2 Proof of Theorem 3

MOUCB algorithm first pulls each arm $2A$ times. Then for $t > 2AK$ and under event $\mathcal{E}_2$, if at round $t+1$, MOUCB algorithm choose a non-optimal arm $I_{t+1} = a \neq i_O$, we have

$$\begin{aligned}&\hat{\mu}_a + \beta(N_a(t)) + \frac{2A}{N_a(t)}\max_{i,j}\left\{\hat{\mu}_i - \hat{\mu}_j + \beta(N_i(t)) + \beta(N_j(t))\right\}\\ \ge&\hat{\mu}_{i_O} + \beta(N_{i_O}(t)) + \frac{2A}{N_{i_O}(t)}\max_{i,j}\left\{\hat{\mu}_i - \hat{\mu}_j + \beta(N_i(t)) + \beta(N_j(t))\right\},\end{aligned}$$

which implies

$$\begin{aligned}&\hat{\mu}_a + \frac{A}{N_a(t)}\left(2\Delta_{i_O,i_W} + 8\sqrt{\frac{\sigma^2 K}{A}\log\frac{4\pi^2A^2}{3\delta}}\right) + \beta(N_a(t))\\ \ge&\hat{\mu}_{i_O} + \frac{A}{N_{i_O}(t)}\Delta_{i_O,i_W} + \beta(N_{i_O}(t)),\end{aligned}$$

according to Lemma 5.

From equation (A.6), we could find

$$\begin{aligned}\hat{\mu}_a &\le \frac{1}{N_a(t)}\sum_{s\in\tau_a(t)}\mu_{I_s^0} + \beta(N_a(t))\\ &\le \mu_a + \Delta_{i_O,a}\frac{C_a}{N_a(t)} + \beta(N_a(t))\\ &\le \mu_a + \Delta_{i_O,a}\frac{A}{N_a(t)} + \beta(N_a(t)),\end{aligned}$$

and

$$\begin{aligned}\hat{\mu}_{i_O} &\geq \frac{1}{N_{i_O}(t)} \sum_{s \in \tau_{i_O}(t)} \mu_{I_s^0} - \beta(N_{i_O}(t)) \\ &\geq \mu_{i_O} - \Delta_{i_O,i_W} \frac{C_{i_O}}{N_{i_O}(t)} - \beta(N_{i_O}(t)) \\ &\geq \mu_{i_O} - \Delta_{i_O,i_W} \frac{A}{N_{i_O}(t)} - \beta(N_{i_O}(t)).\end{aligned}$$

By combining the inequalities above, we have

$$\mu_{i_O} \leq \mu_a + \Delta_{i_O,a} \frac{A}{N_a(t)} + 2\beta(N_a(t)) + \frac{A}{N_a(t)} \left( 2\Delta_{i_O,i_W} + 8\sqrt{\frac{\sigma^2 K}{A} \log \frac{4\pi^2 A^2}{3\delta}} \right),$$

which is equivalent to

$$\begin{aligned}\Delta_{i_o,a} \leq & \Delta_{i_O,a} \frac{A}{N_a(t)} + 2\sqrt{\frac{2\sigma^2 K}{N_a(t)} \log \frac{\pi^2 (N_a(t))^2}{3\delta}} + \frac{A}{N_a(t)} \left( 2\Delta_{i_O,i_W} + 8\sqrt{\frac{\sigma^2 K}{A} \log \frac{4\pi^2 A^2}{3\delta}} \right) \\ \leq & 2\sqrt{\frac{2\sigma^2 K}{N_a(t)} \log \frac{\pi^2 t^2}{3\delta}} + \frac{A}{N_a(t)} \left( \Delta_{i_O,a} + 2\Delta_{i_O,i_W} + 8\sqrt{\frac{\sigma^2 K}{A} \log \frac{4\pi^2 A^2}{3\delta}} \right).\end{aligned}$$

Therefore,

$$N_a(t) \leq \max \left\{ \frac{8\sigma^2 K}{\Delta_{i_O,a}^2} \log \frac{\pi^2 t^2}{3\delta}, \frac{A}{\Delta_{i_o,a}} \left( \Delta_{i_O,a} + 2\Delta_{i_O,i_W} + 8\sqrt{\frac{\sigma^2 K}{A} \log \frac{4\pi^2 A^2}{3\delta}} \right) \right\}. \tag{A.56}$$

As event $\mathcal{E}_2$ holds with probability at least $1 - \delta$, (A.21) holds with probability at least $1 - \delta$. Then Theorem 3 follows immediately from the definition of the pseudo-regret in (2.2) and equation (A.56).

# Appendix B

# Appendix of Chapter 3

## B.1 Attack Cost Analysis of White-box Setting

### B.1.1 Proof of Proposition 2

When the agent pulls a non-target arm $I_t \neq K$, the mean reward received by the agent should satisfy $\mathbb{E}[r_{t,I_t^0}|F_{t-1}, I_t] = (1-\alpha)\langle x_t, \theta_K\rangle$. In the observation of the agent, the target arm becomes optimal and the non-target arms are associated with the coefficient vector $(1-\alpha)\theta_K$. In addition, the cumulative pseudo-regret should satisfy $\bar{R}_T = \sum_{t=1}^{T} \mathbb{1}_{\{I_t \neq K\}} \alpha \langle x_t, \theta_K\rangle \leq \sum_{t=1}^{T} \mathbb{1}_{\{I_t \neq K\}} \alpha L S$. If $\bar{R}_T$ is upper bounded by $o(T)$, $\sum_{t=1}^{T} \mathbb{1}_{\{I_t \neq K\}}$ is also upper bounded by $o(T)$.

### B.1.2 Proof of Lemma 6

If the agent computes an estimate of $\theta_i$ by (3.3) and $V_{t,i} = \left(\sum_{k\in\tau_i(t-1)} x_k x_k^T + \lambda\mathbf{I}\right)$, we have

$$\begin{aligned}
&x_t^T\hat{\theta}_{t,i} - x_t^T(1-\alpha)\theta_K \\
=&x_t^T V_{t,i}^{-1}\left(\sum_{k\in\tau_i(t-1)} r_{t,I_k^0} x_k\right) - x_t^T V_{t,i}^{-1} V_{t,i}(1-\alpha)\theta_K \\
=&x_t^T V_{t,i}^{-1}\left(\sum_{k\in\tau_i(t-1)} x_k\left(r_{t,I_k^0} - (1-\alpha)x_k^T\theta_K\right)\right) - \lambda x_t^T V_{t,i}^{-1}(1-\alpha)\theta_K \\
=&\sum_{k\in\tau_i(t-1)} x_t^T V_{t,i}^{-1} x_k\left(x_k^T\theta_{I_k^0} + \eta_k - (1-\alpha)x_k^T\theta_K\right) - \lambda x_t^T V_{t,i}^{-1}(1-\alpha)\theta_K,
\end{aligned} \tag{B.1}$$

and by triangle inequality,

$$\begin{aligned}
&|x_t^T\hat{\theta}_{t,i} - x_t^T(1-\alpha)\theta_K| \\
\leq&\left|\sum_{k\in\tau_i(t-1)} x_t^T V_{t,i}^{-1} x_k\left(x_k^T\theta_{I_k^0} - (1-\alpha)x_k^T\theta_K\right)\right| \\
&+\left|\sum_{k\in\tau_i(t-1)} x_t^T V_{t,i}^{-1} x_k\eta_k\right| + \left|\lambda x_t^T V_{t,i}^{-1}(1-\alpha)\theta_K\right|.
\end{aligned} \tag{B.2}$$

In our model, the mean reward is bounded by $0 < \langle x_t, \theta_i\rangle \leq \|x_t\|_2^2\|\theta_i\|_2^2 = LS$. Since the mean rewards are bounded and the rewards are generated independently, we have $0 \leq \left|x_k^T\theta_{I_k^0} - (1-\alpha)x_k^T\theta_K\right| \leq LS$ and $\mathbb{E}[x_k^T\theta_{I_k^0}|F_{k-1}] = (1-\alpha)x_k^T\theta_K$. Thus, $\left\{x_t^T V_{t,i}^{-1} x_k\left(x_k^T\theta_{I_k^0} - (1-\alpha)x_k^T\theta_K\right)\right\}_{k\in\tau_i(t-1)}$ is a bounded martingale difference sequence w.r.t the filtration $\{F_k\}_{k\in\tau_i(t-1)}$.

Then, by Azuma's inequality,

$$\begin{aligned}&\mathbb{P}\left(\left|\sum_{k\in\tau_i(t-1)} x_t^T V_{t,i}^{-1} x_k \left(x_k^T\theta_{I_k^0} - (1-\alpha)x_k^T\theta_K\right)\right| \geq B\right)\\ \leq &2\exp\left(\frac{-2B^2}{\sum_{k\in\tau_i(t-1)}(x_t^T V_{t,i}^{-1} x_k LS)^2}\right)\\ =&P_{t,i},\end{aligned} \tag{B.3}$$

where $B$ represents confidence bound. In order to ensure the confidence bounds hold for all arms and all round $t$ simultaneously, we set $P_{t,i} = \frac{\delta}{KT}$ so

$$\begin{aligned}B &= LS\sqrt{\frac{1}{2}\log\left(\frac{2KT}{\delta}\right)\sum_{k\in\tau_i(t-1)}(x_t^T V_{t,i}^{-1} x_k)^2}\\ &\leq LS\sqrt{\frac{1}{2}\log\left(\frac{2KT}{\delta}\right)}\|x_t\|_{V_{t,i}^{-1}},\end{aligned} \tag{B.4}$$

where the last inequality is obtained from the fact that

$$\begin{aligned}\|x_t\|_{V_{t,i}^{-1}}^2 &= x_t^T V_{t,i}^{-1}\left(\sum_{k\in\tau_i(t-1)} x_k x_k^T + \lambda\mathbf{I}\right)V_{t,i}^{-1}x_t\\ &\geq x_t^T V_{t,i}^{-1}\left(\sum_{k\in\tau_i(t-1)} x_k x_k^T\right)V_{t,i}^{-1}x_t\\ &= \sum_{k\in\tau_i(t-1)}(x_t^T V_{t,i}^{-1} x_k)^2.\end{aligned} \tag{B.5}$$

In other words, with probability $1-\delta$, we have

$$\begin{aligned}&\left|\sum_{k\in\tau_i(t-1)} x_t^T V_{t,i}^{-1} x_k \left(x_k^T\theta_{I_k^0} - (1-\alpha)x_k^T\theta_K\right)\right|\\ \leq &LS\sqrt{\frac{1}{2}\log\left(\frac{2KT}{\delta}\right)}\|x_t\|_{V_{t,i}^{-1}},\end{aligned} \tag{B.6}$$

for all arms and all $t$.

Note that $V_{t,i} = \sum_{k\in\tau_i(t-1)} x_k x_k^T + \lambda\mathbf{I}$ is positive definite. We define $\langle x, y\rangle_V = x^T V y$ as the weighted inner-product. According to Cauchy-Schwarz inequality, we have

$$\left|\sum_{k\in\tau_i(t-1)} x_t^T V_{t,i}^{-1} x_k \eta_k\right| \leq \|x_t\|_{V_{t,i}^{-1}} \left\|\sum_{k\in\tau_i(t-1)} x_k\eta_k\right\|_{V_{t,i}^{-1}}. \tag{B.7}$$

Assume that $\lambda \geq L$. From Theorem 1 and Lemma 11 in [1], we know that for any $\delta > 0$, with probability at least $1-\delta$

$$\begin{aligned}&\left\|\sum_{k\in\tau_i(t-1)} x_k\eta_k\right\|_{V_{t,i}^{-1}}^2\\ \leq& 2R^2\log\left(\frac{K\det(V_{t,i})^{1/2}\det(\lambda\mathbf{I})^{-1/2}}{\delta}\right)\\ \leq& R\sqrt{2\log\frac{K}{\delta} + d\log\left(1+\frac{L^2N_i(t)}{\lambda d}\right)},\end{aligned} \tag{B.8}$$

for all arms and all $t > 0$.

For the third part of the right hand side of (B.10),

$$|\lambda x_t^T V_{t,i}^{-1}(1-\alpha)\theta_K| \leq \|(1-\alpha)\lambda\theta_K\|_{V_{t,i}^{-1}} \|x_t\|_{V_{t,i}^{-1}}. \tag{B.9}$$

Since $V_{t,i} \succeq \lambda\mathbf{I}$, the maximum eigenvalue of $V_{t,i}^{-1}$ is smaller or equal to $1/\lambda$. Thus, $\|(1-\alpha)\lambda\theta_K\|_{V_{t,i}^{-1}}^2 \leq \frac{1}{\lambda}\|(1-\alpha)\lambda\theta_K\|_2^2 \leq (1-\alpha)^2\lambda S^2$.

In summary,

$$\begin{aligned}&|x_t^T\hat{\theta}_{t,i} - x_t^T(1-\alpha)\theta_K|\\ \leq& \left((1-\alpha)\sqrt{\lambda}S + LS\sqrt{\frac{1}{2}\log\left(\frac{2KT}{\delta}\right)} + R\sqrt{2\log\frac{K}{\delta} + d\log\left(1+\frac{L^2N_i(t)}{\lambda d}\right)}\right)\|x_t\|_{V_{t,i}^{-1}}.\end{aligned} \tag{B.10}$$

### B.1.3 Proof of Theorem 4

For round $t$ and context $x_t$, if LinUCB pulls arm $i \neq K$, we have

$$x_t^T \hat{\theta}_{t,K} + \beta_{t,K}\sqrt{x_t^T V_{t,K}^{-1} x_t} \leq x_t^T \hat{\theta}_{t,i} + \beta_{t,i}\sqrt{x_t^T V_{t,i}^{-1} x_t}.$$

Recall $\beta_{t,i} = \sqrt{\lambda} S + R\sqrt{2 \log \frac{K}{\delta} + d \log\left(1 + \frac{L^2 N_i(t)}{\lambda d}\right)}$.

Since the attacker does not attack the target arm, the confidence bound of arm $K$ does not change and $x_t^T \theta_K \leq x_t^T \hat{\theta}_{t,K} + \beta_{t,K}\sqrt{x_t^T V_{t,K}^{-1} x_t}$ holds with probability $1 - \frac{\delta}{K}$.

Thus, by Lemma 1,

$$\begin{aligned} x_t^T \theta_K \leq & x_t^T \hat{\theta}_{t,i} + \beta_{t,i}\sqrt{x_t^T V_{t,i}^{-1} x_t} \\ \leq & x_t^T (1-\alpha)\theta_K + \beta_{t,i}\|x_t\|_{V_{t,i}^{-1}} + LS\sqrt{\frac{1}{2}\log\left(\frac{2KT}{\delta}\right)}\omega\left(N_i(t)\right)\|x_t\|_{V_{t,i}^{-1}}. \end{aligned} \tag{B.11}$$

By multiplying both sides $\mathbb{1}_{\{I_t = i\}}$ and summing over rounds, we have

$$\begin{aligned} & \sum_{k=1}^{t} \mathbb{1}_{\{I_k = i\}} \alpha x_k^T \theta_K \\ \leq & \sum_{k=1}^{t} \mathbb{1}_{\{I_k = i\}} \left(\beta_{k,i} + \sqrt{\lambda} S + LS\sqrt{\frac{1}{2}\log\left(\frac{2KT}{\delta}\right)} \right. \\ & \left. + R\sqrt{2\log\frac{K}{\delta} + d\log\left(1 + \frac{L^2 N_i(k)}{\lambda d}\right)}\right)\|x_k\|_{V_{k,i}^{-1}}. \end{aligned} \tag{B.12}$$

Here, we use Lemma 11 from [1] and obtain

$$\begin{aligned} \sum_{k=1}^{t} \mathbb{1}_{\{I_k = i\}} \|x_k\|_{V_{k,i}^{-1}}^2 & \leq 2d\log(1 + \frac{N_i(t)L^2}{d\lambda}) \\ & \leq 2d\log\left(1 + \frac{tL^2}{d\lambda}\right). \end{aligned} \tag{B.13}$$

According to $\sum_{k=1}^{t} \mathbb{1}_{\{I_k=i\}} \|x_k\|_{V_{k,i}^{-1}} \leq \sqrt{N_i(t) \sum_{k=1}^{t} \mathbb{1}_{\{I_k=i\}} \|x_k\|^2_{V_{k,i}^{-1}}}$, we have

$$\sum_{k=1}^{t} \mathbb{1}_{\{I_k=i\}} \|x_k\|^2_{V_{k,i}^{-1}} \leq \sqrt{N_i(t) 2d \log\left(1 + \frac{tL^2}{d\lambda}\right)}. \tag{B.14}$$

Thus, we have

$$\begin{aligned}\sum_{k=1}^{t} \mathbb{1}_{\{I_k=i\}} \alpha x_k^T \theta_K \leq & \sqrt{N_i(t) 2d \log\left(1 + \frac{tL^2}{d\lambda}\right)} \left( LS\sqrt{\frac{1}{2}\log\left(\frac{2KT}{\delta}\right)} \right. \\ & \left. + 2\sqrt{\lambda}S + 2R\sqrt{2\log\frac{K}{\delta} + d\log\left(1 + \frac{tL^2}{\lambda d}\right)} \right),\end{aligned} \tag{B.15}$$

and

$$\begin{aligned}N_i(t) = \sum_{k=1}^{t} \mathbb{1}_{\{I_k=i\}} \leq & \frac{2d}{(\alpha\gamma)^2} \log\left(1 + \frac{tL^2}{d\lambda}\right) \left( 2\sqrt{\lambda}S + LS\sqrt{\frac{1}{2}\log\left(\frac{2KT}{\delta}\right)} \right. \\ & \left. + 2R\sqrt{2\log\frac{K}{\delta} + d\log\left(1 + \frac{tL^2}{\lambda d}\right)} \right)^2,\end{aligned} \tag{B.16}$$

where $\gamma = \min_{x \in \mathcal{D}} \langle x, \theta_K \rangle$.

## B.2 Attack Cost Analysis of Black-box Setting

### B.2.1 Proof of Lemma 7

Since the estimate of $\theta_i$ obtained by the agent satisfies

$$\hat{\theta}^0_{t,i} = \left(V^0_{t,i}\right)^{-1} \left( \sum_{k \in \tau_i^{\dagger}(t-1)} w_{k,i} r_{k,I_k^0} x_k \right), \tag{B.17}$$

we have

$$
\begin{aligned}
& x_t^T \hat{\theta}^0_{t,i} - x_t^T \theta_i \\
=& x_t^T \left(V^0_{t,i}\right)^{-1} \left( \sum_{k \in \tau_i^\dagger(t-1)} w_{k,i} r_{k,I_k^0} x_k \right) - x_t^T \left(V^0_{t,i}\right)^{-1} V^0_{t,i} \theta_i \\
=& x_t^T \left(V^0_{t,i}\right)^{-1} \left( \sum_{k \in \tau_i^\dagger(t-1)} (w_{k,i} r_{k,I_k^0} - x_k^T \theta_i) x_k \right) - \lambda x_t^T \left(V^0_{t,i}\right)^{-1} \theta_i \\
=& x_t^T \left(V^0_{t,i}\right)^{-1} \left( \sum_{k \in \tau_i^\dagger(t-1)} (w_{k,i} x_k^T \theta_{I_k^0} - x_k^T \theta_i) x_k \right) \\
& + x_t^T \left(V^0_{t,i}\right)^{-1} \left( \sum_{k \in \tau_i^\dagger(t-1)} w_{k,i} \eta_k \right) - \lambda x_t^T \left(V^0_{t,i}\right)^{-1} \theta_i .
\end{aligned}
$$

We have $0 \leq \left| w_{k,i} x_k^T \theta_{I_k^0} - x_k^T \theta_i \right| \leq w_{k,i} LS$ and $\mathbb{E}[w_{k,i} x_k^T \theta_{I_k^0} | F_{k-1}] = x_k^T \theta_i$. In addition, by the definition of $w_{k,i}$, we have that $w_{k,i} \leq 1/\alpha$ if $i \neq K$, and $w_{k,i} \leq 2$ if $i = K$. Thus, $\left\{ x_t^T \left(V^0_{t,i}\right)^{-1} \left( \sum_{k \in \tau_i^\dagger(t-1)} (w_{k,i} x_k^T \theta_{I_k^0} - x_k^T \theta_i) x_k \right) \right\}_{k \in \tau_i(t-1)}$ is also a bounded martingale difference sequence w.r.t the filtration $\{F_k\}_{k \in \tau_i(t-1)}$. By following the steps in Section B.1.2, we have, with probability $1 - \frac{K-1}{K}\delta$, for any arm $i \neq K$ and any round $t$,

$$
\left| x_t^T \left(V^0_{t,i}\right)^{-1} \left( \sum_{k \in \tau_i^\dagger(t-1)} (w_{k,i} x_k^T \theta_{I_k^0} - x_k^T \theta_i) x_k \right) \right| \leq \frac{LS}{\alpha} \sqrt{\frac{1}{2} \log \left( \frac{2KT}{\delta} \right)} \|x_t\|_{\left(V^0_{t,i}\right)^{-1}},
$$

and with probability $1 - \frac{1}{K}\delta$, for arm $K$ and any round $t$,

$$
\left| x_t^T \left(V^0_{t,K}\right)^{-1} \left( \sum_{k \in \tau_i^\dagger(t-1)} (w_{k,K} x_k^T \theta_{I_k^0} - x_k^T \theta_K) x_k \right) \right| \leq 2LS \sqrt{\frac{1}{2} \log \left( \frac{2KT}{\delta} \right)} \|x_t\|_{\left(V^0_{t,K}\right)^{-1}}.
$$

The confidence bound of the second item of the right side hand of (B.18) can be obtained from (B.8). With probability, $1 - \frac{K-1}{K}\delta$, for any arm $i \neq K$ and any round $t$,

$$\left| x_t^T \left(V_{t,i}^0\right)^{-1} \left( \sum_{k \in \tau_i^\dagger(t-1)} w_{k,i}\eta_k \right) \right| \leq \frac{R}{\alpha} \sqrt{2 \log \frac{K}{\delta} + d \log \left(1 + \frac{L^2 N_i^\dagger(t)}{\lambda d}\right)} \|x_t\|_{\left(V_{t,i}^0\right)^{-1}}. \tag{B.18}$$

With probability, $1 - \frac{1}{K}\delta$, for arm $K$ and any round $t$,

$$\left| x_t^T \left(V_{t,i}^0\right)^{-1} \left( \sum_{k \in \tau_i^\dagger(t-1)} w_{k,i}\eta_k \right) \right| \leq 2R \sqrt{2 \log \frac{K}{\delta} + d \log \left(1 + \frac{L^2 N_K^\dagger(t)}{\lambda d}\right)} \|x_t\|_{\left(V_{t,K}^0\right)^{-1}}. \tag{B.19}$$

In summary,

$$\begin{aligned} & |x_t^T \hat{\theta}_{t,i}^0 - x_t^T \theta_i| \\ \leq & \phi_i \left( \sqrt{\lambda} S + LS \sqrt{\frac{1}{2} \log \left(\frac{2KT}{\delta}\right)} + R \sqrt{2 \log \frac{K}{\delta} + d \log \left(1 + \frac{L^2 N_i^\dagger(t)}{\lambda d}\right)} \right) \|x_t\|_{\left(V_{t,K}^0\right)^{-1}}, \end{aligned} \tag{B.20}$$

where $\phi_i = 1/\alpha$ when $i \neq K$ and $\phi_K = 2$.

### B.2.2 Proof of Lemma 8

Recall the definition of $\epsilon_t$:

$$\epsilon_t = \text{clip}\left( \frac{1}{2}, \frac{(1-\alpha)\langle x_t, \hat{\theta}_{t,K}^0 \rangle - \langle x_t, \hat{\theta}_{t,I_t^\dagger}^0 \rangle}{\langle x_t, \hat{\theta}_{t,K}^0 \rangle - \langle x_t, \hat{\theta}_{t,I_t^\dagger}^0 \rangle}, 1 - \alpha \right), \tag{B.21}$$

and the definition of $I_t^\dagger$:

$$I_t^\dagger = \arg\min_{i \neq K} \left( \langle x_t, \hat{\theta}_{t,i}^0 \rangle - \beta_{t,i}^0 \|x_t\|_{(V_{t,i}^0)^{-1}} \right). \tag{B.22}$$

By Lemma 7, $\langle x_t, \hat{\theta}_{t,I_t^\dagger}^0 \rangle - \beta_{t,I_t^\dagger}^0 \|x_t\|_{(V_{t,I_t^\dagger}^0)^{-1}} \leq \min_i \langle x_t, \theta_i \rangle$ with probability $1 - 2\delta$.

Because $\epsilon_t$ is bounded by $[1/2, 1-\alpha]$, we can analyze $\mathbb{E}[r_{t,I_t^0} | F_{t-1}, I_t]$ in four cases.

**Case 1**: when $\langle x_t, \hat{\theta}^0_{t,K}\rangle < \langle x_t, \hat{\theta}^0_{t,I_t^\dagger}\rangle$ and $\epsilon_t = 1-\alpha$, we have

$$\mathbb{E}[r_{t,I_t^0}|F_{t-1}, I_t] = (1-\alpha)\langle x_t, \theta_K\rangle + \alpha\langle x_t, \theta_{I_t^\dagger}\rangle. \tag{B.23}$$

Then, by Lemma 7,

$$\begin{aligned}
&(1-\alpha)x_t^T\theta_K + \alpha x_t^T\theta_{I_t^\dagger} - (1-\alpha)x_t^T\theta_K \\
\leq&(1-\alpha)\left(x_t^T\hat{\theta}^0_{t,K} + \beta^0_{t,K}\|x_t\|_{\left(V^0_{t,K}\right)^{-1}}\right) + \alpha\left(x_t^T\hat{\theta}^0_{t,I_t^\dagger} + \beta^0_{t,I_t^\dagger}\|x_t\|_{\left(V^0_{t,I_t^\dagger}\right)^{-1}}\right) - (1-\alpha)x_t^T\theta_K \\
\leq& x_t^T\hat{\theta}^0_{t,I_t^\dagger} - (1-\alpha)x_t^T\theta_K + (1-\alpha)\beta^0_{t,K}\|x_t\|_{\left(V^0_{t,K}\right)^{-1}} + \alpha\beta^0_{t,I_t^\dagger}\|x_t\|_{\left(V^0_{t,I_t^\dagger}\right)^{-1}} \\
\leq&(1-\alpha)\beta^0_{t,K}\|x_t\|_{\left(V^0_{t,K}\right)^{-1}} + (1+\alpha)\beta^0_{t,I_t^\dagger}\|x_t\|_{\left(V^0_{t,I_t^\dagger}\right)^{-1}},
\end{aligned} \tag{B.24}$$

where the last inequality is obtained by $x_t^T\hat{\theta}^0_{t,I_t^\dagger} - \beta^0_{t,I_t^\dagger}\|x_t\|_{\left(V^0_{t,I_t^\dagger}\right)^{-1}} \leq \min_i\langle x_t, \theta_i\rangle$ and Assumption 1. We also have

$$(1-\alpha)x_t^T\theta_K + \alpha x_t^T\theta_{I_t^\dagger} - (1-\alpha)x_t^T\theta_K = \alpha x_t^T\theta_{I_t^\dagger} \geq 0. \tag{B.25}$$

**Case 2**: when $\langle x_t, \hat{\theta}^0_{t,K}\rangle \geq \langle x_t, \hat{\theta}^0_{t,I_t^\dagger}\rangle > (1-2\alpha)\langle x_t, \hat{\theta}^0_{t,K}\rangle$ and $\epsilon_t = 1/2$, we have

$$\mathbb{E}[r_{t,I_t^0}|F_{t-1}, I_t] = \frac{1}{2}\langle x_t, \theta_K\rangle + \frac{1}{2}\langle x_t, \theta_{I_t^\dagger}\rangle. \tag{B.26}$$

Then, by Lemma 7,

$$\begin{aligned}
&\frac{1}{2}(x_t^T\theta_K + x_t^T\theta_{I_t^\dagger}) - (1-\alpha)x_t^T\theta_K \\
=&\frac{1}{2}\left(x_t^T\theta_{I_t^\dagger} - (1-2\alpha)x_t^T\theta_K\right) \\
\leq&\frac{1}{2}\left(x_t^T\hat{\theta}^0_{t,I_t^\dagger} + \beta^0_{t,I_t^\dagger}\|x_t\|_{\left(V^0_{t,I_t^\dagger}\right)^{-1}} - (1-2\alpha)x_t^T\theta_K\right) \\
\leq&\beta^0_{t,I_t^\dagger}\|x_t\|_{\left(V^0_{t,I_t^\dagger}\right)^{-1}}
\end{aligned}$$

where the last inequality is obtained by $x_t^T\hat{\theta}^0_{t,I_t^\dagger} - \beta^0_{t,I_t^\dagger}\|x_t\|_{\left(V^0_{t,I_t^\dagger}\right)^{-1}} \leq \min_i\langle x_t, \theta_i\rangle$ and Assumption 1.

In addition, by Lemma 7,

$$\begin{aligned}
&\frac{1}{2}(x_t^T\theta_K + x_t^T\theta_{I_t^\dagger}) - (1-\alpha)x_t^T\theta_K \\
\geq&\frac{1}{2}\left(x_t^T\hat{\theta}^0_{t,I_t^\dagger} - \beta^0_{t,I_t^\dagger}\|x_t\|_{\left(V^0_{t,I_t^\dagger}\right)^{-1}}\right) - \frac{1}{2}(1-2\alpha)\left(x_t^T\hat{\theta}^0_{t,K} + \beta^0_{t,K}\|x_t\|_{(V^0_{t,K})^{-1}}\right) \qquad \text{(B.27)} \\
\geq&-\frac{1}{2}\beta^0_{t,I_t^\dagger}\|x_t\|_{\left(V^0_{t,I_t^\dagger}\right)^{-1}} - \frac{1}{2}(1-2\alpha)\beta^0_{t,K}\|x_t\|_{(V^0_{t,K})^{-1}}.
\end{aligned}$$

**Case 3**: when $0 \leq \langle x_t, \hat{\theta}^0_{t,I_t^\dagger}\rangle \leq (1-2\alpha)\langle x_t, \hat{\theta}^0_{t,K}\rangle$ and $1/2 \leq \epsilon_t \leq 1-\alpha$, we have

$$\mathbb{E}[r_{t,I_t^0}|F_{t-1}, I_t] = \epsilon_t\langle x_t, \theta_K\rangle + (1-\epsilon_t)\langle x_t, \theta_{I_t^\dagger}\rangle. \qquad \text{(B.28)}$$

We can find that

$$\begin{aligned}
&\epsilon_t\langle x_t,\theta_K\rangle + (1-\epsilon_t)\langle x_t,\theta_{I_t^\dagger}\rangle - (1-\alpha)\langle x_t,\theta_K\rangle \\
=&\epsilon_t(\langle x_t,\theta_K\rangle - \langle x_t,\theta_{I_t^\dagger}\rangle) + \langle x_t,\theta_{I_t^\dagger}\rangle - (1-\alpha)\langle x_t,\theta_K\rangle \\
=&\epsilon_t(\langle x_t,\hat{\theta}^0_{t,K}\rangle - \langle x_t,\hat{\theta}^0_{t,I_t^\dagger}\rangle) + \langle x_t,\theta_{I_t^\dagger}\rangle - (1-\alpha)\langle x_t,\theta_K\rangle \\
&+\epsilon_t(\langle x_t,\hat{\theta}^0_{t,I_t^\dagger}\rangle - \langle x_t,\theta_{I_t^\dagger}\rangle) + \epsilon_t(\langle x_t,\theta_K\rangle - \langle x_t,\hat{\theta}^0_{t,K}\rangle) \\
=&(1-\alpha)\langle x_t,\hat{\theta}^0_{t,K}\rangle - \langle x_t,\hat{\theta}^0_{t,I_t^\dagger}\rangle + \langle x_t,\theta_{I_t^\dagger}\rangle - (1-\alpha)\langle x_t,\theta_K\rangle \\
&+\epsilon_t(\langle x_t,\hat{\theta}^0_{t,I_t^\dagger}\rangle - \langle x_t,\theta_{I_t^\dagger}\rangle) + \epsilon_t(\langle x_t,\theta_K\rangle - \langle x_t,\hat{\theta}^0_{t,K}\rangle) \\
=&(1-\alpha-\epsilon_t)\left(\langle x_t,\hat{\theta}^0_{t,K}\rangle - \langle x_t,\theta_K\rangle\right) + (1-\epsilon_t)\left(\langle x_t,\hat{\theta}^0_{t,I_t^\dagger}\rangle - \langle x_t,\theta_{I_t^\dagger}\rangle\right),
\end{aligned} \tag{B.29}$$

which is equivalent to

$$\begin{aligned}
&\left|\mathbb{E}[r_{t,I_t^0}|F_{t-1},I_t] - (1-\alpha)\langle x_t,\theta_K\rangle\right| \\
\leq&(1-\alpha-\epsilon_t)\beta^0_{t,K}\|x_t\|_{\left(V^0_{t,K}\right)^{-1}} + (1-\epsilon_t)\beta^0_{t,I_t^\dagger}\|x_t\|_{\left(V^0_{t,I_t^\dagger}\right)^{-1}}.
\end{aligned} \tag{B.30}$$

**Case 4**: when $\langle x_t,\hat{\theta}^0_{t,I_t^\dagger}\rangle < 0$ and $\epsilon_t = 1-\alpha$, we have

$$\mathbb{E}[r_{t,I_t^0}|F_{t-1},I_t] = (1-\alpha)\langle x_t,\theta_K\rangle + \alpha\langle x_t,\theta_{I_t^\dagger}\rangle. \tag{B.31}$$

Then, by Lemma 7,

$$\begin{aligned}
&(1-\alpha)x_t^T\theta_K + \alpha x_t^T\theta_{I_t^\dagger} - (1-\alpha)x_t^T\theta_K \\
=&\alpha x_t^T\theta_{I_t^\dagger} \\
\leq&\alpha x_t^T\hat{\theta}^0_{t,I_t^\dagger} + \alpha\beta^0_{t,I_t^\dagger}\|x_t\|_{\left(V^0_{t,I_t^\dagger}\right)^{-1}} \\
\leq&\alpha\beta^0_{t,I_t^\dagger}\|x_t\|_{\left(V^0_{t,I_t^\dagger}\right)^{-1}},
\end{aligned} \tag{B.32}$$

where the last inequality is obtained by $x_t^T\hat{\theta}^0_{t,I_t^\dagger} - \beta^0_{t,I_t^\dagger}\|x_t\|_{\left(V^0_{t,I_t^\dagger}\right)^{-1}} \leq \min_i\langle x_t,\theta_i\rangle$ and

Assumption 1. We also have

$$(1-\alpha)x_t^T\theta_K + \alpha x_t^T\theta_{I_t^\dagger} - (1-\alpha)x_t^T\theta_K = \alpha x_t^T\theta_{I_t^\dagger} \geq 0. \tag{B.33}$$

Combining these four cases, we have

$$\begin{aligned} &\left|\mathbb{E}[r_{t,I_t^0}|F_{t-1}, I_t] - (1-\alpha)\langle x_t, \theta_K\rangle\right| \\ \leq &(1-\alpha)\beta_{t,K}^0\|x_t\|_{\left(V_{t,K}^0\right)^{-1}} + (1+\alpha)\beta_{t,I_t^\dagger}^0\|x_t\|_{\left(V^0_{t,I_t^\dagger}\right)^{-1}}. \end{aligned} \tag{B.34}$$

### B.2.3 Proof of Lemma 9

From Section B.1.2, we have, for any arm $i \neq K$,

$$\begin{aligned} &|x_t^T\hat{\theta}_{t,i} - x_t^T(1-\alpha)\theta_K| \\ \leq &\left|\sum_{k\in\tau_i(t-1)} x_t^T V_{t,i}^{-1}x_k\left(x_k^T\theta_{I_k^0} - (1-\alpha)x_k^T\theta_K\right)\right| \\ &+\left|\sum_{k\in\tau_i(t-1)} x_t^T V_{t,i}^{-1}x_k\eta_k\right| + \left|\lambda x_t^T V_{t,i}^{-1}(1-\alpha)\theta_K\right| \\ \leq &\left|\sum_{k\in\tau_i(t-1)} x_t^T V_{t,i}^{-1}x_k\left(x_k^T\theta_{I_k^0} - \epsilon_k\langle x_k, \theta_K\rangle - (1-\epsilon_k)\langle x_k, \theta_{I_k^\dagger}\rangle\right)\right| \\ &+\left|\sum_{k\in\tau_i(t-1)} x_t^T V_{t,i}^{-1}x_k\left(\epsilon_k\langle x_k, \theta_K\rangle + (1-\epsilon_k)\langle x_k, \theta_{I_k^\dagger}\rangle - (1-\alpha)x_k^T\theta_K\right)\right| \\ &+\left|\sum_{k\in\tau_i(t-1)} x_t^T V_{t,i}^{-1}x_k\eta_k\right| + \left|\lambda x_t^T V_{t,i}^{-1}(1-\alpha)\theta_K\right|. \end{aligned} \tag{B.35}$$

Since the mean rewards are bounded and the rewards are generated independently, we have $0 \leq \left|x_k^T\theta_{I_k^0} - \epsilon_k\langle x_k, \theta_K\rangle - (1-\epsilon_k)\langle x_k, \theta_{I_k^\dagger}\rangle\right| \leq LS$ and $\mathbb{E}[x_k^T\theta_{I_k^0}|F_{k-1}] = \epsilon_k\langle x_k, \theta_K\rangle + (1-\epsilon_k)\langle x_k, \theta_{I_k^\dagger}\rangle$.

Then $\left\{x_t^T V_{t,i}^{-1}x_k\left(x_k^T\theta_{I_k^0} - \mathbb{E}[x_k^T\theta_{I_k^0}|F_{k-1}]\right)\right\}_{k\in\tau_i(t-1)}$ is also a bounded martingale difference sequence w.r.t the filtration $\{F_k\}_{k\in\tau_i(t-1)}$. By following the steps in Section B.1.2, we have, with probability $1-\delta$, for any arm $i$ and any round $t$,

$$\left|\sum_{k\in\tau_i(t-1)} x_t^T V_{t,i}^{-1} x_k \left(x_k^T\theta_{I_k^0} - \mathbb{E}[x_k^T\theta_{I_k^0}|F_{k-1}]\right)\right| \le LS\sqrt{\frac{1}{2}\log\left(\frac{2KT}{\delta}\right)}\|x_t\|_{V_{t,i}^{-1}}. \tag{B.36}$$

From (B.5) in Section B.1.2, we have

$$\|x_t\|_{V_{t,i}^{-1}}^2 \ge \sum_{k\in\tau_i(t-1)} (x_t^T V_{t,i}^{-1} x_k)^2. \tag{B.37}$$

Then, the second item of the right hand side of (B.35) can be upper bounded by

$$\begin{aligned}
&\left|\sum_{k\in\tau_i(t-1)} x_t^T V_{t,i}^{-1} x_k \left(\epsilon_k\langle x_k,\theta_K\rangle + (1-\epsilon_k)\langle x_t,\theta_{I_k^\dagger}\rangle - (1-\alpha)x_k^T\theta_K\right)\right| \\
\le&\sqrt{\sum_{k\in\tau_i(t-1)}\left(\mathbb{E}[r_{k,I_k^0}|F_{k-1},I_k] - (1-\alpha)x_k^T\theta_K\right)^2}\sqrt{\sum_{k\in\tau_i(t-1)} (x_t^T V_{t,i}^{-1} x_k)^2} \\
\le&\left(\sum_{k\in\tau_i(t-1)}\left((1-\alpha)\beta_{k,K}^0\|x_k\|_{\left(V_{k,K}^0\right)^{-1}} + (1+\alpha)\beta_{k,I_k^\dagger}^0\|x_k\|_{\left(V_{k,I_k^\dagger}^0\right)^{-1}}\right)^2\right)^{\frac{1}{2}}\|x_t\|_{V_{t,i}^{-1}},
\end{aligned} \tag{B.38}$$

where the first inequality is obtained from Cauchy-Schwarz inequality, the second inequality is obtained from Lemma 8 and (B.5).

In addition, by the fact that $(a+b)^2 \le 2a^2+2b^2$ for any real number, we have

$$\begin{aligned}
&\sum_{k\in\tau_i(t-1)}\left((1-\alpha)\beta_{k,K}^0\|x_k\|_{\left(V_{k,K}^0\right)^{-1}} + (1+\alpha)\beta_{k,I_k^\dagger}^0\|x_k\|_{\left(V_{k,I_k^\dagger}^0\right)^{-1}}\right)^2 \\
\le&\sum_{k\in\tau_i(t-1)} 2\left((1-\alpha)\beta_{k,K}^0\|x_k\|_{\left(V_{k,K}^0\right)^{-1}}\right)^2 + \sum_{k\in\tau_i(t-1)} 2\left((1+\alpha)\beta_{k,I_k^\dagger}^0\|x_k\|_{\left(V_{k,I_k^\dagger}^0\right)^{-1}}\right)^2.
\end{aligned} \tag{B.39}$$

Here, we use Lemma 11 from [1] and get, for any arm $i$,

$$\begin{aligned}\sum_{k\in\tau_i^\dagger(t-1)}\|x_k\|^2_{(V^0_{k,i})^{-1}} &\leq 2d\log\left(1+\frac{N_i(t)L^2}{d\lambda}\right)\\ &\leq 2d\log\left(1+\frac{tL^2}{d\lambda}\right).\end{aligned}\tag{B.40}$$

By the fact that $\sum_i \tau_i(t-1)=\tau_K^\dagger(t-1)$, and $\sum_{i\neq K}\tau_i(t-1)=\sum_{i\neq K}\tau_i^\dagger(t-1)$, we have, for any arm $i$, $\tau_i(t-1)\subseteq\tau_K^\dagger(t-1)$, and $\tau_i(t-1)\subseteq\sum_{j\neq K}\tau_j^\dagger(t-1)$. Thus,

$$\begin{aligned}\sum_{k\in\tau_i(t-1)}\|x_k\|^2_{\left(V^0_{k,K}\right)^{-1}} &\leq \sum_{k\in\tau_K^\dagger(t-1)}\|x_k\|^2_{(V^0_{k,K})^{-1}}\\ &\leq 2d\log\left(1+\frac{tL^2}{d\lambda}\right),\end{aligned}\tag{B.41}$$

and

$$\begin{aligned}\sum_{k\in\tau_i(t-1)}\|x_k\|^2_{\left(V^0_{k,I_k^\dagger}\right)^{-1}} &\leq \sum_{i\neq K}\sum_{k\in\tau_i^\dagger(t-1)}\|x_k\|^2_{(V^0_{k,i})^{-1}}\\ &\leq 2(K-1)d\log\left(1+\frac{tL^2}{d\lambda}\right).\end{aligned}\tag{B.42}$$

By combining (3.10), (B.39), (B.41) and (B.42), we have

$$\begin{aligned}&\sum_{k\in\tau_i(t-1)}\left((1-\alpha)\beta^0_{k,K}\|x_k\|_{\left(V^0_{k,K}\right)^{-1}}+(1+\alpha)\beta^0_{k,I_k^\dagger}\|x_k\|_{\left(V^0_{k,I_k^\dagger}\right)^{-1}}\right)^2\\ \leq&\sum_{k\in\tau_i(t-1)}2\left(\beta^0_{k,K}\|x_k\|_{\left(V^0_{k,K}\right)^{-1}}\right)^2+\sum_{k\in\tau_i(t-1)}2\left(2\beta^0_{k,I_k^\dagger}\|x_k\|_{\left(V^0_{k,I_k^\dagger}\right)^{-1}}\right)^2\\ \leq&16d^2\left(\omega(t)+LS\sqrt{\frac{1}{2}\log\left(\frac{2KT}{\delta}\right)}\right)^2\log\left(1+\frac{tL^2}{d\lambda}\right)\\ &+\frac{16d^2(K-1)}{\alpha^2}\left(\omega(t)+LS\sqrt{\frac{1}{2}\log\left(\frac{2KT}{\delta}\right)}\right)^2\log\left(1+\frac{tL^2}{d\lambda}\right)\\ \leq&\frac{16d^2K}{\alpha^2}\left(\omega(t)+LS\sqrt{\frac{1}{2}\log\left(\frac{2KT}{\delta}\right)}\right)^2\log\left(1+\frac{tL^2}{d\lambda}\right).\end{aligned}\tag{B.43}$$

In summary, we have

$$\begin{aligned}&|x_t^T\hat{\theta}_{t,i} - x_t^T(1-\alpha)\theta_K| \\ &\leq \left(1+\frac{4d}{\alpha}\sqrt{K\log\left(1+\frac{tL^2}{d\lambda}\right)}\right)\left(\omega(N_i(t)) + LS\sqrt{\frac{1}{2}\log\left(\frac{2KT}{\delta}\right)}\right)\|x_t\|_{V_{t,i}^{-1}}.\end{aligned} \tag{B.44}$$

### B.2.4 Proof of Theorem 5

For round $t$ and context $x_t$, if LinUCB pulls arm $i \neq K$, we have

$$x_t^T\hat{\theta}_{t,K} + \beta_{t,K}\sqrt{x_t^T V_{t,K}^{-1} x_t} \leq x_t^T\hat{\theta}_{t,i} + \beta_{t,i}\sqrt{x_t^T V_{t,i}^{-1} x_t}.$$

In this case, $\beta_{t,i} = \omega(N_i(t)) = \sqrt{\lambda}S + R\sqrt{2\log\frac{K}{\delta} + d\log\left(1+\frac{L^2N_i(t)}{\lambda d}\right)}$.

Since the attacker does not attack the target arm, the confidence bound of arm $K$ does not change and $x_t^T\theta_K \leq x_t^T\hat{\theta}_{t,K} + \beta_{t,K}\sqrt{x_t^T V_{t,K}^{-1} x_t}$ holds with probability $1-\frac{\delta}{K}$.

Thus, by Lemma 2,

$$\begin{aligned}x_t^T\theta_K \leq& x_t^T\hat{\theta}_{t,i} + \beta_{t,i}\sqrt{x_t^T V_{t,i}^{-1} x_t} \\ \leq& x_t^T(1-\alpha)\theta_K + \omega(N_i(t))\|x_t\|_{V_{t,i}^{-1}} \\ &+ \left(1+\frac{4d}{\alpha}\sqrt{K\log\left(1+\frac{tL^2}{d\lambda}\right)}\right)\left(\omega(N_i(t)) + LS\sqrt{\frac{1}{2}\log\left(\frac{2KT}{\delta}\right)}\right)\|x_t\|_{V_{t,i}^{-1}}.\end{aligned} \tag{B.45}$$

By multiplying both sides by $\mathbb{1}_{\{I_t=i\}}$ and summing over rounds, we have

$$\begin{aligned}&\sum_{k=1}^{t}\mathbb{1}_{\{I_k=i\}}\alpha x_k^T\theta_K \\ \leq& \sum_{k=1}^{t}\mathbb{1}_{\{I_k=i\}}\left(2+\frac{4d}{\alpha}\sqrt{K\log\left(1+\frac{kL^2}{d\lambda}\right)}\right)\left(\omega(N_k(t)) + LS\sqrt{\frac{1}{2}\log\left(\frac{2KT}{\delta}\right)}\right)\|x_k\|_{V_{k,i}^{-1}}.\end{aligned} \tag{B.46}$$

Here, we use Lemma 11 from [1] and get

$$\begin{aligned}\sum_{k=1}^{t} \mathbb{1}_{\{I_k=i\}} \|x_k\|^2_{V_{k,i}^{-1}} &\leq 2d \log\left(1 + \frac{N_i(t)L^2}{d\lambda}\right) \\ &\leq 2d \log\left(1 + \frac{tL^2}{d\lambda}\right).\end{aligned} \tag{B.47}$$

According to $\sum_{k=1}^{t} \mathbb{1}_{\{I_k=i\}} \|x_k\|_{V_{k,i}^{-1}} \leq \sqrt{N_i(t) \sum_{k=1}^{t} \mathbb{1}_{\{I_k=i\}} \|x_k\|^2_{V_{k,i}^{-1}}}$, we have

$$\sum_{k=1}^{t} \mathbb{1}_{\{I_k=i\}} \|x_k\|^2_{V_{k,i}^{-1}} \leq \sqrt{N_i(t) 2d \log\left(1 + \frac{tL^2}{d\lambda}\right)}. \tag{B.48}$$

Thus, we have

$$\begin{aligned}&\sum_{k=1}^{t} \mathbb{1}_{\{I_k=i\}} \alpha x_k^T \theta_K \\ \leq& \sqrt{N_i(t) 2d \log\left(1 + \frac{tL^2}{d\lambda}\right)} \left(2 + \frac{4d}{\alpha} \sqrt{K \log\left(1 + \frac{tL^2}{d\lambda}\right)}\right) \left(\omega(t) + LS\sqrt{\frac{1}{2}\log\left(\frac{2KT}{\delta}\right)}\right),\end{aligned} \tag{B.49}$$

and

$$\begin{aligned}N_i(t) =& \sum_{k=1}^{t} \mathbb{1}_{\{I_k=i\}} \\ \leq& \frac{2d}{(\alpha\gamma)^2} \log\left(1 + \frac{tL^2}{d\lambda}\right) \left(2 + \frac{4d}{\alpha} \sqrt{K \log\left(1 + \frac{tL^2}{d\lambda}\right)}\right)^2 \left(\omega(t) + LS\sqrt{\frac{1}{2}\log\left(\frac{2KT}{\delta}\right)}\right)^2,\end{aligned} \tag{B.50}$$

where $\gamma = \min_{x \in \mathcal{D}} \langle x, \theta_K \rangle$.

## B.3 Proof of Generalized Linear Model

### B.3.1 Proof of Lemma 10

The maximum-likelihood estimation can be written as the solution to the following equation

$$\sum_{n\in\tau_i(t-1)} (r_n - \mu(x_n^T\hat{\theta}_{t,i}))x_n = 0. \tag{B.51}$$

Define $g_{t,i}(\theta) = \sum_{n\in\tau_i(t-1)} \mu(x_n^T\theta))x_n$. $g_{t,i}(\hat{\theta}_{t,i}) = \sum_{n\in\tau_i(t-1)} r_n x_n$. Since $\mu$ is continuously twice differentiable, $\nabla g_{t,i}$ is continuous, and for any $\theta \in \Theta$, $\nabla g_{t,i}(\theta) = \sum_{n\in\tau_i(t-1)} x_n x_n^T \dot{\mu}(x_n^T\theta))$. $\nabla g_{t,i}(\theta)$ denotes the Jacobian matrix of $g_{t,i}$ at $\theta$. By the Fundamental Theorem of Calculus,

$$g_{t,i}(\hat{\theta}_{t,i}) - g_{t,i}((1-\alpha)\theta_K) = G_{t,i}(\hat{\theta}_{t,i} - (1-\alpha)\theta_K), \tag{B.52}$$

where

$$G_{t,i} = \int_0^1 \nabla g_{t,i}\left(s\hat{\theta}_{t,i} + (1-s)(1-\alpha)\theta_K\right)\,\mathrm{d}s. \tag{B.53}$$

Note that $\nabla g_{t,i}(\theta) = \sum_{n\in\tau_i(t-1)} x_n x_n^T \dot{\mu}(x_n^T\theta))$. According to the assumption that $c_\mu = \inf_{\theta\in\Theta,x\in\mathcal{D}} \dot{\mu}(x^T\theta) > 0$, we have $G_{t,i} \succeq c_\mu V_{t,i} \succeq c_\mu V_{KJ,i} \succeq \lambda_0 I \succ 0$, where in the last two step we used the assumption that the minimal eigenvalue of $V_i$ is greater or equal to $\lambda_0$ after playing arm $i$ $J$ times. Thus, $G_{t,i}$ is positive definite and non-singular. Therefore,

$$\hat{\theta}_i - (1-\alpha)\theta_K = G_{t,i}^{-1}\left(g_{t,i}(\hat{\theta}_i) - g_{t,i}((1-\alpha)\theta_K)\right). \tag{B.54}$$

For arm $K$, $g_{t,i}(\hat{\theta}_i) - g_{t,i}(\theta_K) = \sum_{n\in\tau_i(t-1)} \eta_n x_n$.

For all arm $i \neq K$, the right hand side of (B.54) is equivalent to

$$\begin{aligned} & g_{t,i}(\hat{\theta}_i) - g_{t,i}((1-\alpha)\theta_K) \\ = & \sum_{n \in \tau_i(t-1)} (r_n - \mu((1-\alpha)x_n^T\theta_K))x_n \\ = & \sum_{n \in \tau_i(t-1)} (\mu(x_n^T\theta_{I_n^0}) - \mu((1-\alpha)x_n^T\theta_K))x_n + \sum_{n \in \tau_i(t-1)} \eta_n x_n. \end{aligned} \tag{B.55}$$

We set $Z_1 = \sum_{n \in \tau_i(t-1)}(\mu(x_n^T\theta_{I_n^0}) - \mu((1-\alpha)x_n^T\theta_K))x_n$ and $Z_2 = \sum_{n \in \tau_i(t-1)} \eta_n x_n$. We have $g_{t,i}(\hat{\theta}_i) - g_{t,i}((1-\alpha)\theta_K) = Z_1 + Z_2$ and

$$x_t^T(\hat{\theta}_i - (1-\alpha)\theta_K) = x_t^T G_{t,i}^{-1}(Z_1 + Z_2). \tag{B.56}$$

For any context $x \in \mathcal{D}$ and arm $i \neq K$, we have

$$\begin{aligned} & |x^T(\hat{\theta}_{t,i} - (1-\alpha)\theta_K)| \\ = & |x^T G_{t,i}^{-1}(Z_1 + Z_2)| \\ \leq & |x^T G_{t,i}^{-1} Z_1| + |x^T G_{t,i}^{-1} Z_2|. \end{aligned} \tag{B.57}$$

We first bound $|x^T G_{t,i}^{-1} Z_2|$. Since $G_{t,i}$ is positive definite and $G_{t,i}^{-1}$ is also positive definite, $|x^T G_{t,i}^{-1} Z_2| \leq \|x\|_{V_{t,i}^{-1}} \|Z_2\|_{G_{t,i}^{-1}}$.

Since $G_{t,i} \succeq c_\mu V_{t,i}$ implies that $G_{t,i}^{-1} \preceq c_\mu^{-1} \bar{V}_{t,i}^{-1}$, we have $\|x\|_{G_{t,i}^{-1}} \leq \frac{1}{\sqrt{c_\mu}} \|x\|_{\bar{V}_{t,i}^{-1}}$ holds for any $x \in \mathbb{R}^d$. Thus,

$$|x^T G_{t,i}^{-1} Z_2| \leq \frac{1}{c_\mu} \|x\|_{\bar{V}_{t,i}^{-1}} \|Z_2\|_{\bar{V}_{t,i}^{-1}}. \tag{B.58}$$

Note that $V_{t,i} = \bar{V}_{t,i} + \lambda \mathbf{I}$. Hence, for all vector $x \in \mathbb{R}^d$

$$\|x\|_{\bar{V}_{t,i}^{-1}}^2 = \|x\|_{V_{t,i}^{-1}}^2 + x^T(\bar{V}_{t,i}^{-1} - V_{t,i}^{-1})x. \tag{B.59}$$

Since $(A+B)^{-1} = A^{-1} - A^{-1}B(A+B)^{-1}$,

$$V_{t,i}^{-1} = \bar{V}_{t,i}^{-1} - \lambda \bar{V}_{t,i}^{-1} V_{t,i}^{-1}. \tag{B.60}$$

The above implies that

$$\begin{aligned} 0 \leq & x^T(\bar{V}_{t,i}^{-1} - V_{t,i}^{-1})x \\ = & x^T \left( \lambda \bar{V}_{t,i}^{-1} V_{t,i}^{-1} \right) x \\ \leq & \frac{\lambda}{\lambda_0} \|x\|_{V_{t,i}^{-1}}^2. \end{aligned} \tag{B.61}$$

and $\|x\|_{\bar{V}_{t,i}^{-1}}^2 \leq (1 + \frac{\lambda}{\lambda_0}) \|x\|_{V_{t,i}^{-1}}^2$.

From Theorem 1 and Lemma 11 in [1], we know that for any $\delta > 0$, with probability at least $1-\delta$

$$\begin{aligned} & \left\| \sum_{k \in \tau_i(t-1)} x_k \eta_k \right\|_{V_{t,i}^{-1}}^2 \\ \leq & 2R^2 \log \left( \frac{K \det(V_{t,i})^{1/2} \det(\lambda \mathbf{I})^{-1/2}}{\delta} \right) \\ \leq & 2R^2 \left( \log \frac{K}{\delta} + d \log \left( 1 + \frac{L^2 N_i(t)}{\lambda d} \right) \right), \end{aligned} \tag{B.62}$$

for all arms and all $t > 0$.

Set $\lambda = \lambda_0$, we have

$$\|Z_2\|_{\bar{V}_{t,i}^{-1}} \leq 2R \sqrt{\log \frac{K}{\delta} + d \log \left( 1 + \frac{L^2 N_i(t)}{\lambda_0 d} \right)}. \tag{B.63}$$

Now we bound $|x^T G_{t,i}^{-1} Z_1|$. Similarly,

$$|x^T G_{t,i}^{-1} Z_1| \leq \frac{1}{c_\mu} \|x\|_{\bar{V}_{t,i}^{-1}} \|Z_1\|_{\bar{V}_{t,i}^{-1}}. \tag{B.64}$$

In our model, we have $0 < \langle x_t, \theta_i \rangle \leq \|x_t\|_2^2 \|\theta_i\|_2^2 = LS$. Further,

$$\begin{aligned} 0 &\leq \left| \mu(x_k^T \theta_{I_k^0}) - \mu((1-\alpha) x_k^T \theta_K) \right| \\ &\leq k_\mu \left| x_k^T \theta_{I_k^0} - (1-\alpha) x_k^T \theta_K \right| \\ &\leq k_\mu LS. \end{aligned} \tag{B.65}$$

Since we have $\mathbb{E}[\mu(x_k^T \theta_{I_k^0}) | F_{k-1}] = \mu((1-\alpha) x_k^T \theta_K)$, $\left\{ \mu(x_k^T \theta_{I_k^0}) - \mu\left((1-\alpha) x_k^T \theta_K\right) \right\}_{k \in \tau_i(t-1)}$ is a bounded martingale difference sequence w.r.t the filtration $\{F_k\}_{k \in \tau_i(t-1)}$ and is also $k_\mu LS$-sub-Gaussian-sub-Gaussian.

From Theorem 1 and Lemma 11 in [1], we know that for any $\delta > 0$, with probability at least $1 - \frac{K-1}{K}\delta$

$$\|Z_1\|_{\bar{V}_{t,i}^{-1}} \leq 2 k_\mu LS \sqrt{\log \frac{K}{\delta} + d \log \left( 1 + \frac{L^2 N_i(t)}{\lambda_0 d} \right)}, \tag{B.66}$$

for any arm $i \neq K$ and all $t > 0$.

In summary, for all arm $i \neq K$,

$$|x^T (\hat{\theta}_i - (1-\alpha)\theta_K)| \leq \frac{2 k_\mu LS + 2R}{c_\mu} \sqrt{\log \frac{K}{\delta} + d \log \left( 1 + \frac{L^2 N_i(t)}{\lambda_0 d} \right)} \|x\|_{\bar{V}_{t,i}^{-1}}. \tag{B.67}$$

### B.3.2 Proof of Theorem 6

For round $t$ and context $x_t$, if GLM-UCB pulls arm $i \neq K$, we have

$$x_t^T \hat{\theta}_{t,K} + \beta_{t,K} \sqrt{x_t^T \bar{V}_{t,K}^{-1} x_t} \leq x_t^T \hat{\theta}_{t,i} + \beta_{t,i} \sqrt{x_t^T \bar{V}_{t,i}^{-1} x_t}.$$

Recall $\beta_{t,i} = \frac{4R}{c_\mu} \sqrt{\log \frac{K}{\delta} + d \log \left( 1 + \frac{L^2 N_i(t)}{\lambda_0 d} \right)}$.

Since the attacker does not attack the target arm, the confidence bound of arm $K$ does not change and $x_t^T \theta_K \leq x_t^T \hat{\theta}_{t,K} + \beta_{t,K} \sqrt{x_t^T V_{t,K}^{-1} x_t}$ holds with probability $1 - \frac{\delta}{K}$.

Thus, by Lemma (10),

$$\begin{aligned} x_t^T\theta_K \leq & x_t^T\hat{\theta}_{t,i} + \beta_{t,i}\sqrt{x_t^T V_{t,i}^{-1} x_t} \\ \leq & x_t^T(1-\alpha)\theta_K + \frac{k_\mu LS + 2R}{R}\beta_{t,i}\|x_t\|_{\bar{V}_{t,i}^{-1}}. \end{aligned} \tag{B.68}$$

By multiplying both sides $\mathbb{1}_{\{I_t=i\}}$ and summing over rounds, we have

$$\begin{aligned} & \sum_{k=1}^{t}\mathbb{1}_{\{I_k=i\}}\alpha x_k^T\theta_K \\ \leq & \sum_{k=1}^{t}\mathbb{1}_{\{I_k=i\}}\frac{k_\mu LS + 2R}{R}\beta_{t,i}\|x_t\|_{\bar{V}_{t,i}^{-1}}. \end{aligned} \tag{B.69}$$

Here, we use Lemma 11 from [1] and obtain

$$\begin{aligned} \sum_{k=1}^{t}\mathbb{1}_{\{I_k=i\}}\|x_k\|^2_{V_{k,i}^{-1}} & \leq 2d\log(1+\frac{N_i(t)L^2}{d\lambda}) \\ & \leq 2d\log\left(1+\frac{tL^2}{d\lambda}\right). \end{aligned} \tag{B.70}$$

According to $\sum_{k=1}^{t}\mathbb{1}_{\{I_k=i\}}\|x_k\|_{V_{k,i}^{-1}} \leq \sqrt{N_i(t)\sum_{k=1}^{t}\mathbb{1}_{\{I_k=i\}}\|x_k\|^2_{V_{k,i}^{-1}}}$, we have

$$\sum_{k=1}^{t}\mathbb{1}_{\{I_k=i\}}\|x_k\|_{V_{k,i}^{-1}} \leq \sqrt{N_i(t)2d\log\left(1+\frac{tL^2}{d\lambda}\right)}. \tag{B.71}$$

Set $\lambda=\lambda_0$, we have $\|x\|^2_{\bar{V}_{t,i}^{-1}} \leq (1+\frac{\lambda}{\lambda_0})\|x\|^2_{V_{t,i}^{-1}} \leq 2\|x\|^2_{V_{t,i}^{-1}}$. Thus, we have

$$\sum_{k=1}^{t}\mathbb{1}_{\{I_k=i\}}\alpha x_k^T\theta_K \leq \frac{k_\mu LS + 2R}{R}\beta_{t,i}\sqrt{4N_i(t)d\log\left(1+\frac{tL^2}{d\lambda_0}\right)}, \tag{B.72}$$

and

$$N_i(t) = \sum_{k=1}^{t}\mathbb{1}_{\{I_k=i\}} \leq \frac{4d}{(\alpha\gamma)^2}\log\left(1+\frac{tL^2}{d\lambda_0}\right)\left(\frac{k_\mu LS + 2R}{R}\beta_{t,i}\right)^2, \tag{B.73}$$

where $\gamma = \min_{x\in\mathcal{D}}\langle x, \theta_K\rangle$.

### B.3.3 Proof of Lemma 11

The attacker calculate the maximum-likelihood estimator $\hat{\theta}^0_{t,i}$ by solving the equation

$$\sum_{n\in\tau_i(t-1)^\dagger} (w_{n,i}r_n - \mu(x_n^T\hat{\theta}_i))x_n = 0. \tag{B.74}$$

Note that $g^0_{t,i}(\theta) = \sum_{n\in\tau_i^\dagger(t-1)} \mu(x_n^T\theta))x_n$. $g^0_{t,i}(\hat{\theta}^0_{t,i}) = \sum_{n\in\tau_i^\dagger(t-1)} w_{n,i}r_nx_n$.

For all arm $i$,

$$\begin{aligned}
&g^0_{t,i}(\hat{\theta}^0_{t,i}) - g^0_{t,i}(\theta_i)\\
=&\sum_{n\in\tau_i^\dagger(t-1)} (w_{n,i}r_n - \mu(x_n^T\theta_i))x_n\\
=&\sum_{n\in\tau_i^\dagger(t-1)} (w_{n,i}\mu(x_n^T\theta_{I_n^0}) - \mu(x_n^T\theta_i))x_n + \sum_{n\in\tau_i^\dagger(t-1)} w_{n,i}\eta_nx_n.
\end{aligned} \tag{B.75}$$

Similarly, we set $Z_3 = \sum_{n\in\tau_i^\dagger(t-1)} w_{n,i}\eta_nx_n$ and $Z_4 = \sum_{n\in\tau_i^\dagger(t-1)}(w_{n,i}\mu(x_n^T\theta_{I_n^0}) - \mu((1-\alpha)x_n^T\theta_K))x_n$ .

We have $g^0_{t,i}(\hat{\theta}^0_{t,i}) - g^0_{t,i}(\theta_i) = Z_3 + Z_4$ and

$$x_t^T(\hat{\theta}^0_{t,i} - \theta_i) = x_t^T(G^0_{t,i})^{-1}(Z_3 + Z_4), \tag{B.76}$$

where

$$G^0_{t,i} = \int_0^1 \nabla g^0_{t,i}\left(s\hat{\theta}^0_{t,i} + (1-s)\theta_i\right)\,\mathrm{d}s. \tag{B.77}$$

For any context $x \in \mathcal{D}$, we have

$$\begin{aligned}
&|x^T(\hat{\theta}^0_{t,i} - \theta_i)|\\
=&|x^T(G^0_{t,i})^{-1}(Z_3 + Z_4)|\\
\le&|x^T(G^0_{t,i})^{-1}Z_3| + |x^T(G^0_{t,i})^{-1}Z_4|.
\end{aligned} \tag{B.78}$$

We first bound $|x^T(G^0_{t,i})^{-1}Z_3|$. We have

$$|x^T(G^0_{t,i})^{-1}Z_3| \leq \frac{1}{c_\mu}\|x\|_{(\bar{V}^0_{t,i})^{-1}}\|Z_3\|_{(\bar{V}^0_{t,i})^{-1}}, \tag{B.79}$$

where $\bar{V}^0_{t,i} = \sum_{n \in \tau_i(t-1)^\dagger} x_n x_n^T$

Note that $V_{t,i}{=}\bar{V}^0_{t,i} + \lambda \mathbf{I}$. Hence,

$$\begin{aligned} \|Z_3\|^2_{(\bar{V}^0_{t,i})^{-1}} =& \|Z_3\|^2_{(V^0_{t,i})^{-1}} + Z_3^T((\bar{V}^0_{t,i})^{-1} - (V^0_{t,i})^{-1})Z_3 \\ \leq& (1+\frac{\lambda}{\lambda_0})\|Z_3\|^2_{(V^0_{t,i})^{-1}}. \end{aligned} \tag{B.80}$$

From Theorem 1 and Lemma 11 in [1], we know that for any $\delta > 0$, with probability at least $1-\delta$

$$\begin{aligned} &\left\| \sum_{k \in \tau_i(t-1)} x_k \eta_k \right\|^2_{V^{-1}_{t,i}} \\ \leq& 2R^2 \log\left(\frac{K \det(V_{t,i})^{1/2}\det(\lambda \mathbf{I})^{-1/2}}{\delta}\right) \\ \leq& 2R^2\left(\log\frac{K}{\delta} + d\log\left(1+\frac{L^2 N_i(t)}{\lambda d}\right)\right), \end{aligned} \tag{B.81}$$

for all arms and all $t > 0$.

Set $\lambda = \lambda_0$, we have

$$\|Z_3\|^2_{(\bar{V}^0_{t,i})^{-1}} \leq 2\phi_i R\sqrt{\log\frac{K}{\delta} + d\log\left(1+\frac{L^2 N^0_i(t)}{\lambda_0 d}\right)}. \tag{B.82}$$

Now we bound $|x^T(G^0_{t,i})^{-1}Z_4|$. Similarly,

$$|x^T(G^0_{t,i})^{-1}Z_4| \leq \frac{1}{c_\mu}\|x\|_{(\bar{V}^0_{t,i})^{-1}}\|Z_4\|_{(\bar{V}^0_{t,i})^{-1}}. \tag{B.83}$$

In our model, we have $0 < \langle x_t, \theta_i \rangle \leq \|x_t\|_2^2 \|\theta_i\|_2^2 = LS$. Further,

$$0 \leq \left| w_{k,i}\mu(x_k^T \theta_{I^0_k}) - \mu((1-\alpha)x_k^T\theta_K)\right| \leq \phi_i k_\mu LS. \tag{B.84}$$

Since we have $\mathbb{E}[w_{k,i}\mu(x_k^T\theta_{I_k^0})|F_{k-1}] = \mu(x_k^T\theta_i)$, $\left\{w_{k,i}\mu(x_k^T\theta_{I_k^0}) - \mu\left(x_k^T\theta_i\right)\right\}_{k\in\tau_i(t-1)}$ is a bounded martingale difference sequence w.r.t the filtration $\{F_k\}_{k\in\tau_i(t-1)}$ and is also $\phi_i k_\mu LS$-sub-Gaussian-sub-Gaussian.

From Theorem 1 and Lemma 11 in [1], we know that for any $\delta > 0$, with probability at least $1-\delta$

$$\|Z_4\|^2_{(\bar{V}^0_{t,i})^{-1}} \le 2\phi_i k_\mu LS\sqrt{\log\frac{K}{\delta} + d\log\left(1+\frac{L^2N_i^0(t)}{\lambda_0 d}\right)}. \tag{B.85}$$

for any arm $i \neq K$ and all $t > 0$.

In summary, for all arm $i \neq K$,

$$|x^T(\hat{\theta}_i - (1-\alpha)\theta_K)| \le 2\phi_i\frac{k_\mu LS + R}{c_\mu}\sqrt{\log\frac{K}{\delta} + d\log\left(1+\frac{L^2N_i(t)}{\lambda_0 d}\right)}\|x\|_{(\bar{V}^0_{t,i})^{-1}}. \tag{B.86}$$

### B.3.4 Proof of Lemma 12

Recall the definition of $\epsilon_t$:

$$\epsilon_t = \text{clip}\left(\frac{c_\mu}{c_\mu + k_\mu}, \frac{\mu((1-\alpha)x_t^T\hat{\theta}^0_{t,K}) - \mu(x_t^T\hat{\theta}^0_{t,I_t^\dagger})}{\mu(x_t^T\hat{\theta}^0_{t,K}) - \mu(x_t^T\hat{\theta}^0_{t,I_t^\dagger})}, 1-\alpha\frac{c_\mu}{k_\mu}\right), \tag{B.87}$$

and the definition of $I_t^\dagger$:

$$I_t^\dagger = \arg\min_{i\neq K}\left(\langle x_t, \hat{\theta}^0_{t,i}\rangle - \beta^0_{t,i}\|x_t\|_{(\bar{V}^0_{t,i})^{-1}}\right). \tag{B.88}$$

By Lemma 11, $\langle x_t, \hat{\theta}^0_{t,I_t^\dagger}\rangle - \beta^0_{t,I_t^\dagger}\|x_t\|_{(\bar{V}^0_{t,I_t^\dagger})^{-1}} \le \min_i\langle x_t, \theta_i\rangle$ with probability $1-2\delta$. Thus, with probability $1-2\delta$, $\mu(x_t^T\hat{\theta}^0_{t,I_t^\dagger}) - \min_i \mu(x_t^T\theta_i) \le k_\mu\beta^0_{t,I_t^\dagger}\|x_t\|_{(\bar{V}^0_{t,I_t^\dagger})^{-1}}$.

Because $\epsilon_t$ is bounded by $[1/2, 1-\alpha]$, we can analyze $\mathbb{E}[r_{t,I_t^0}|F_{t-1}, I_t]$ in four cases.

**Case 1**: when $\langle x_t, \hat{\theta}^0_{t,K}\rangle < \langle x_t, \hat{\theta}^0_{t,I_t^\dagger}\rangle$, we have $\epsilon_t = 1-\alpha\frac{c_\mu}{k_\mu}$ and $\mu(\langle x_t, \hat{\theta}^0_{t,K}\rangle) < \mu(\langle x_t, \hat{\theta}^0_{t,I_t^\dagger}\rangle)$. Thus,

$$\mathbb{E}[r_{t,I_t^0}|F_{t-1}, I_t] = (1-\alpha\frac{c_\mu}{k_\mu})\mu(x_t^T\theta_K) + \alpha\frac{c_\mu}{k_\mu}\mu(x_t^T\theta_{I_t^\dagger}). \tag{B.89}$$

Then, by Lemma 11,

$$\begin{aligned}
&(1-\alpha\frac{c_\mu}{k_\mu})\mu(x_t^T\theta_K)+\alpha\frac{c_\mu}{k_\mu}\mu(x_t^T\theta_{I_t^\dagger})-\mu((1-\alpha)x_t^T\theta_K)\\
\le&(1-\alpha\frac{c_\mu}{k_\mu})\left(\mu(x_t^T\hat{\theta}^0_{t,K})+k_\mu\beta^0_{t,K}\|x_t\|_{(\bar{V}^0_{t,K})^{-1}}\right)\\
&+\alpha\frac{c_\mu}{k_\mu}\left(\mu(x_t^T\hat{\theta}^0_{t,I_t^\dagger})+k_\mu\beta^0_{t,I_t^\dagger}\|x_t\|_{\left(\bar{V}^0_{t,I_t^\dagger}\right)^{-1}}\right)-\mu((1-\alpha)x_t^T\theta_K)\\
\le&\mu(x_t^T\hat{\theta}^0_{t,I_t^\dagger})-\mu((1-\alpha)x_t^T\theta_K)+\alpha c_\mu\beta^0_{t,I_t^\dagger}\|x_t\|_{\left(\bar{V}^0_{t,I_t^\dagger}\right)^{-1}}+(1-\alpha\frac{c_\mu}{k_\mu})k_\mu\beta^0_{t,K}\|x_t\|_{(\bar{V}^0_{t,K})^{-1}}\\
\le&(k_\mu-\alpha c_\mu)\beta^0_{t,K}\|x_t\|_{(\bar{V}^0_{t,K})^{-1}}+(k_\mu+\alpha c_\mu)\beta^0_{t,I_t^\dagger}\|x_t\|_{\left(\bar{V}^0_{t,I_t^\dagger}\right)^{-1}},
\end{aligned} \tag{B.90}$$

where the second inequality is obtains by $\mu(\langle x_t,\hat{\theta}^0_{t,K}\rangle)<\mu(\langle x_t,\hat{\theta}^0_{t,I_t^\dagger}\rangle)$ and the last inequality is obtained by $\mu(x_t^T\hat{\theta}^0_{t,I_t^\dagger})-\min_i\mu(x_t^T\theta_i)\le k_\mu\beta^0_{t,I_t^\dagger}\|x_t\|_{(\bar{V}^0_{t,I_t^\dagger})^{-1}}$ and Assumption 1. We also have

$$\begin{aligned}
&(1-\alpha\frac{c_\mu}{k_\mu})\mu(x_t^T\theta_K)+\alpha\frac{c_\mu}{k_\mu}\mu(x_t^T\theta_{I_t^\dagger})-\mu((1-\alpha)x_t^T\theta_K)\\
\ge&(1-\alpha\frac{c_\mu}{k_\mu})\left(\mu(x_t^T\hat{\theta}^0_{t,K})-k_\mu\beta^0_{t,K}\|x_t\|_{(\bar{V}^0_{t,K})^{-1}}\right)\\
&+\alpha\frac{c_\mu}{k_\mu}\left(\mu(x_t^T\hat{\theta}^0_{t,I_t^\dagger})-k_\mu\beta^0_{t,I_t^\dagger}\|x_t\|_{\left(\bar{V}^0_{t,I_t^\dagger}\right)^{-1}}\right)-\mu((1-\alpha)x_t^T\theta_K)\\
\ge&\mu(x_t^T\hat{\theta}^0_{t,K})-\mu((1-\alpha)x_t^T\theta_K)-\alpha c_\mu\beta^0_{t,I_t^\dagger}\|x_t\|_{\left(\bar{V}^0_{t,I_t^\dagger}\right)^{-1}}-(1-\alpha\frac{c_\mu}{k_\mu})k_\mu\beta^0_{t,K}\|x_t\|_{(\bar{V}^0_{t,K})^{-1}}\\
\ge&\mu(x_t^T\theta_K)-\mu((1-\alpha)x_t^T\theta_K)-\alpha c_\mu\beta^0_{t,I_t^\dagger}\|x_t\|_{\left(\bar{V}^0_{t,I_t^\dagger}\right)^{-1}}-(2-\alpha\frac{c_\mu}{k_\mu})k_\mu\beta^0_{t,K}\|x_t\|_{(\bar{V}^0_{t,K})^{-1}}\\
\ge&-(2k_\mu-\alpha c_\mu)\beta^0_{t,K}\|x_t\|_{(\bar{V}^0_{t,K})^{-1}}-\alpha c_\mu\beta^0_{t,I_t^\dagger}\|x_t\|_{\left(\bar{V}^0_{t,I_t^\dagger}\right)^{-1}}.
\end{aligned} \tag{B.91}$$

**Case 2**: when $\mu(x_t^T\hat{\theta}^0_{t,K})\ge\mu(x_t^T\hat{\theta}^0_{t,I_t^\dagger})>(1+\frac{c_\mu}{k_\mu})\mu((1-\alpha)x_t^T\hat{\theta}^0_{t,K})-\frac{c_\mu}{k_\mu}\mu(x_t^T\hat{\theta}^0_{t,K})$ and $\epsilon_t=$

$\frac{c_\mu}{c_\mu+k_\mu}$, we have

$$\mathbb{E}[r_{t,I_t^0}|F_{t-1},I_t]=\frac{c_\mu}{c_\mu+k_\mu}\mu(x_t^T\theta_K)+\frac{k_\mu}{c_\mu+k_\mu}\mu(x_t^T\theta_{I_t^\dagger}). \tag{B.92}$$

By Lemma 11, with probability $1-2\delta$, $\mu(x_t^T\hat{\theta}^0_{t,I_t^\dagger})-\min_i\mu(x_t^T\theta_i)\le k_\mu\beta^0_{t,I_t^\dagger}\|x_t\|_{(\bar{V}^0_{t,I_t^\dagger})^{-1}}$. Since $\min_i x_t^T\theta_i\le(1-2\alpha)x_t^T\theta_K$, we have $\mu(x_t^T\hat{\theta}^0_{t,I_t^\dagger})\le\mu((1-2\alpha)x_t^T\theta_K)+k_\mu\beta^0_{t,I_t^\dagger}\|x_t\|_{(\bar{V}^0_{t,I_t^\dagger})^{-1}}$ and then $\mu(x_t^T\theta_{I_t^\dagger})\le\mu((1-2\alpha)x_t^T\theta_K)+2k_\mu\beta^0_{t,I_t^\dagger}\|x_t\|_{(\bar{V}^0_{t,I_t^\dagger})^{-1}}$. Thus,

$$\begin{aligned}&\frac{c_\mu\mu(x_t^T\theta_K)}{c_\mu+k_\mu}+\frac{k_\mu\mu(x_t^T\theta_{I_t^\dagger})}{c_\mu+k_\mu}-\mu((1-\alpha)x_t^T\theta_K)\\=&\frac{c_\mu}{c_\mu+k_\mu}\mu(x_t^T\theta_K)-\frac{c_\mu}{c_\mu+k_\mu}\mu((1-\alpha)x_t^T\theta_K)\\&+\frac{k_\mu}{c_\mu+k_\mu}\mu(x_t^T\theta_{I_t^\dagger})-\frac{k_\mu}{c_\mu+k_\mu}\mu((1-\alpha)x_t^T\theta_K)\\\le&\frac{c_\mu}{c_\mu+k_\mu}\big(\mu(x_t^T\theta_K)-\mu((1-\alpha)x_t^T\theta_K)\big)\\&+\frac{k_\mu}{c_\mu+k_\mu}\big(\mu((1-2\alpha)x_t^T\theta_K)-\mu((1-\alpha)x_t^T\theta_K)\big)\\&+\frac{2k_\mu^2}{c_\mu+k_\mu}\beta^0_{t,I_t^\dagger}\|x_t\|_{(\bar{V}^0_{t,I_t^\dagger})^{-1}}\end{aligned} \tag{B.93}$$

According to the definition of $k_\mu$ and $c_\mu$ and Lemma 11,

$$\begin{aligned}&\frac{c_\mu\mu(x_t^T\theta_K)}{c_\mu+k_\mu}+\frac{k_\mu\mu(x_t^T\theta_{I_t^\dagger})}{c_\mu+k_\mu}-\mu((1-\alpha)x_t^T\theta_K)\\\le&\frac{c_\mu}{c_\mu+k_\mu}k_\mu\big(x_t^T\theta_K-(1-\alpha)x_t^T\theta_K\big)\\&+\frac{k_\mu}{c_\mu+k_\mu}c_\mu\big((1-2\alpha)x_t^T\theta_K-(1-\alpha)x_t^T\theta_K\big)\\&+\frac{2k_\mu^2}{c_\mu+k_\mu}\beta^0_{t,I_t^\dagger}\|x_t\|_{(\bar{V}^0_{t,I_t^\dagger})^{-1}}\\=&\frac{2k_\mu^2}{c_\mu+k_\mu}\beta^0_{t,I_t^\dagger}\|x_t\|_{(\bar{V}^0_{t,I_t^\dagger})^{-1}}.\end{aligned} \tag{B.94}$$

In addition, by Lemma 11,

$$\begin{aligned}
&\frac{c_\mu \mu(x_t^T\theta_K)}{c_\mu+k_\mu}+\frac{k_\mu \mu(x_t^T\theta_{I_t^\dagger})}{c_\mu+k_\mu}-\mu((1-\alpha)x_t^T\theta_K)\\
=&\frac{c_\mu \mu(x_t^T\theta_K)}{c_\mu+k_\mu}+\frac{k_\mu \mu(x_t^T\theta_{I_t^\dagger})}{c_\mu+k_\mu}-\mu((1-\alpha)x_t^T\hat{\theta}^0_{t,K})\\
&+\mu((1-\alpha)x_t^T\hat{\theta}^0_{t,K})-\mu((1-\alpha)x_t^T\theta_K)\\
\geq&\frac{c_\mu}{c_\mu+k_\mu}\mu(x_t^T\theta_K)+\frac{k_\mu}{c_\mu+k_\mu}\mu(x_t^T\theta_{I_t^\dagger})\\
&-\frac{c_\mu}{c_\mu+k_\mu}\mu(x_t^T\hat{\theta}^0_{t,K})-\frac{k_\mu}{c_\mu+k_\mu}\mu(x_t^T\hat{\theta}^0_{t,I_t^\dagger})\\
&+\mu((1-\alpha)x_t^T\hat{\theta}^0_{t,K})-\mu((1-\alpha)x_t^T\theta_K)\\
\geq&-(1-\alpha+\frac{c_\mu}{c_\mu+k_\mu})k_\mu\beta^0_{t,K}\|x_t\|_{\left(\bar{V}^0_{t,K}\right)^{-1}}\\
&-\frac{k_\mu}{c_\mu+k_\mu}k_\mu\beta^0_{t,I_t^\dagger}\|x_t\|_{\left(\bar{V}^0_{t,I_t^\dagger}\right)^{-1}},
\end{aligned} \tag{B.95}$$

where the first inequality is obtained by the condition of case 2: $\mu(x_t^T\hat{\theta}^0_{t,I_t^\dagger}) > (1+\frac{c_\mu}{k_\mu})\mu((1-\alpha)x_t^T\hat{\theta}^0_{t,K})-\frac{c_\mu}{k_\mu}\mu(x_t^T\hat{\theta}^0_{t,K})$ which is equivalent to $\frac{c_\mu\mu(x_t^T\hat{\theta}^0_{t,K})}{c_\mu+k_\mu}+\frac{k_\mu\mu(x_t^T\hat{\theta}^0_{t,I_t^\dagger})}{c_\mu+k_\mu} > \mu((1-\alpha)x_t^T\hat{\theta}^0_{t,K})$.

**Case 3**: when the attacker's estimates satisfy

$$\begin{aligned}
&\frac{k_\mu}{\alpha c_\mu}\mu((1-\alpha)x_t^T\hat{\theta}^0_{t,K})-(\frac{k_\mu}{\alpha c_\mu}-1)\mu(x_t^T\hat{\theta}^0_{t,K})\\
\leq&\mu(x_t^T\hat{\theta}^0_{t,I_t^\dagger})\\
\leq&(1+\frac{c_\mu}{k_\mu})\mu((1-\alpha)x_t^T\hat{\theta}^0_{t,K})-\frac{c_\mu}{k_\mu}\mu(x_t^T\hat{\theta}^0_{t,K})
\end{aligned} \tag{B.96}$$

and hence $\frac{c_\mu}{c_\mu+k_\mu}\leq\epsilon_t\leq 1-\alpha\frac{c_\mu}{k_\mu}$, we have

$$\mathbb{E}[r_{t,I_t^0}|F_{t-1},I_t]=\epsilon_t\mu(x_t^T\theta_K)+(1-\epsilon_t)\mu(x_t^T\theta_{I_t^\dagger}). \tag{B.97}$$

We can find that

$$\begin{aligned}
&\epsilon_t\mu(x_t^T\theta_K)+(1-\epsilon_t)\mu(x_t^T\theta_{I_t^\dagger})-\mu((1-\alpha)x_t^T\theta_K)\\
=&\epsilon_t(\mu(x_t^T\theta_K)-\mu(x_t^T\theta_{I_t^\dagger}))+\mu(x_t^T\theta_{I_t^\dagger})-\mu((1-\alpha)x_t^T\theta_K)\\
=&\epsilon_t(\mu(x_t^T\hat\theta^0_{t,K})-\mu(x_t^T\hat\theta^0_{t,I_t^\dagger}))+\epsilon_t(\mu(x_t^T\hat\theta^0_{t,I_t^\dagger})-\mu(x_t^T\theta_{I_t^\dagger}))\\
&+\epsilon_t(\mu(x_t^T\theta_K)-\mu(x_t^T\hat\theta^0_{t,K}))+\mu(x_t^T\theta_{I_t^\dagger})-\mu((1-\alpha)x_t^T\theta_K)\\
=&\mu((1-\alpha)x_t^T\hat\theta^0_{t,K})-\mu(x_t^T\hat\theta^0_{t,I_t^\dagger})+\epsilon_t(\mu(x_t^T\hat\theta^0_{t,I_t^\dagger})-\mu(x_t^T\theta_{I_t^\dagger}))\\
&+\epsilon_t(\mu(x_t^T\theta_K)-\mu(x_t^T\hat\theta^0_{t,K}))+\mu(x_t^T\theta_{I_t^\dagger})-\mu((1-\alpha)x_t^T\theta_K)\\
=&\mu((1-\alpha)x_t^T\hat\theta^0_{t,K})-\mu((1-\alpha)x_t^T\theta_K)+\\
&\epsilon_t(\mu(x_t^T\theta_K)-\mu(x_t^T\hat\theta^0_{t,K}))+(1-\epsilon_t)\left(\mu(x_t^T\theta_{I_t^\dagger})-\mu(x_t^T\hat\theta^0_{t,I_t^\dagger})\right),
\end{aligned}\tag{B.98}$$

From Lemma 11,

$$\begin{aligned}
&\left|\mathbb{E}[r_{t,I_t^0}|F_{t-1},I_t]-\mu((1-\alpha)x_t^T\theta_K\right|\\
\le&(1-\alpha+\epsilon_t)k_\mu\beta^0_{t,K}\|x_t\|_{(\bar V^0_{t,K})^{-1}}+(1-\epsilon_t)k_\mu\beta^0_{t,I_t^\dagger}\|x_t\|_{\left(\bar V^0_{t,I_t^\dagger}\right)^{-1}}.
\end{aligned}\tag{B.99}$$

**Case 4**: when $\mu(x_t^T\hat\theta^0_{t,I_t^\dagger})<\frac{k_\mu}{\alpha c_\mu}\mu((1-\alpha)x_t^T\hat\theta^0_{t,K})-(\frac{k_\mu}{\alpha c_\mu}-1)\mu(x_t^T\hat\theta^0_{t,K})$ and $\epsilon_t=1-\alpha\frac{c_\mu}{k_\mu}$, we have

$$\mathbb{E}[r_{t,I_t^0}|F_{t-1},I_t]=(1-\alpha\frac{c_\mu}{k_\mu})\mu(x_t^T\theta_K)+\alpha\frac{c_\mu}{k_\mu}\mu(x_t^T\theta_{I_t^\dagger}).\tag{B.100}$$

Then, by Lemma 7,

$$\begin{aligned}
&(1-\alpha\frac{c_\mu}{k_\mu})\mu(x_t^T\theta_K)+\alpha\frac{c_\mu}{k_\mu}\mu(x_t^T\theta_{I_t^\dagger})-\mu((1-\alpha)x_t^T\theta_K)\\
\leq&(1-\alpha\frac{c_\mu}{k_\mu})\mu(x_t^T\theta_K)-\mu((1-\alpha)x_t^T\theta_K)+\alpha\frac{c_\mu}{k_\mu}\mu(x_t^T\hat{\theta}^0_{t,I_t^\dagger})+\alpha c_\mu\beta^0_{t,I_t^\dagger}\|x_t\|_{(\bar{V}^0_{t,I_t^\dagger})^{-1}}\\
<&(1-\alpha\frac{c_\mu}{k_\mu})\mu(x_t^T\theta_K)-\mu((1-\alpha)x_t^T\theta_K)+\mu((1-\alpha)x_t^T\hat{\theta}^0_{t,K})-(1-\alpha\frac{c_\mu}{k_\mu})\mu(x_t^T\hat{\theta}^0_{t,K}) \qquad \text{(B.101)}\\
&+\alpha c_\mu\beta^0_{t,I_t^\dagger}\|x_t\|_{(\bar{V}^0_{t,I_t^\dagger})^{-1}}\\
\leq&(k_\mu-c_\mu)\beta^0_{t,K}\|x_t\|_{\left(\bar{V}^0_{t,K}\right)^{-1}}+\alpha c_\mu\beta^0_{t,I_t^\dagger}\|x_t\|_{(\bar{V}^0_{t,I_t^\dagger})^{-1}}.
\end{aligned}$$

Since $x_t^T\theta_{I_t^\dagger}>0$, $\mu(x_t^T\theta_K)-\mu(x_t^T\theta_{I_t^\dagger})\leq k_\mu x_t^T\theta_K$. Hence, we also have

$$\begin{aligned}
&(1-\alpha\frac{c_\mu}{k_\mu})\mu(x_t^T\theta_K)+\alpha\frac{c_\mu}{k_\mu}\mu(x_t^T\theta_{I_t^\dagger})-\mu((1-\alpha)x_t^T\theta_K)\\
=&\mu(x_t^T\theta_K)-\mu((1-\alpha)x_t^T\theta_K)-\alpha\frac{c_\mu}{k_\mu}\left(\mu(x_t^T\theta_K)-\mu(x_t^T\theta_{I_t^\dagger})\right)\\
\geq&c_\mu\alpha x_t^T\theta_K-\alpha\frac{c_\mu}{k_\mu}k_\mu x_t^T\theta_K=0.
\end{aligned}\tag{B.102}$$

Combining these four cases, we have

$$\left|\mathbb{E}[r_{t,I_t^0}|F_{t-1},I_t]-\mu((1-\alpha)x_t^T\theta_K)\right|\leq 2k_\mu\beta^0_{t,K}\|x_t\|_{\left(\bar{V}^0_{t,K}\right)^{-1}}+2k_\mu\beta^0_{t,I_t^\dagger}\|x_t\|_{\left(\bar{V}^0_{t,I_t^\dagger}\right)^{-1}}.\tag{B.103}$$

### B.3.5 Proof of Lemma 13

The agent's maximum-likelihood estimation can be written as the solution to the following equation

$$\sum_{n\in\tau_i(t-1)}(r_n-\mu(x_n^T\hat{\theta}_{t,i}))x_n=0.\tag{B.104}$$

As described in the section B.3.1, we have $g_{t,i}(\hat{\theta}_i) - g_{t,i}((1-\alpha)\theta_K) = Z_1 + Z_2$ and

$$x_t^T(\hat{\theta}_i - (1-\alpha)\theta_K) = x_t^T G_{t,i}^{-1}(Z_1 + Z_2). \tag{B.105}$$

We set $Z_1 = \sum_{n\in\tau_i(t-1)}(\mu(x_n^T\theta_{I_n^0}) - \mu((1-\alpha)x_n^T\theta_K))x_n$ and $Z_2 = \sum_{n\in\tau_i(t-1)}\eta_n x_n$.

In the white-box attack case, we have $\mathbb{E}[\mu(x_k^T\theta_{I_k^0})|F_{k-1}] = \mu((1-\alpha)x_k^T\theta_K)$ and hence $\mathbb{E}[Z_1|F_{k-1}] = \mathbf{0}$. Under the proposed black-box attack, $\mathbb{E}[Z_1|F_{k-1}] \neq \mathbf{0}$ but

$$\begin{aligned}&\left|\mathbb{E}[\mu(x_t^T\theta_{I_t^0})|F_{t-1}, I_t] - (1-\alpha)\langle x_t, \theta_K\rangle\right| \\ \le & 2k_\mu\beta_{t,K}^0\|x_t\|_{\left(\bar{V}_{t,K}^0\right)^{-1}} + 2k_\mu\beta_{t,I_t^\dagger}^0\|x_t\|_{\left(\bar{V}_{t,I_t^\dagger}^0\right)^{-1}}.\end{aligned} \tag{B.106}$$

We set $Z_1 = Z_5 + Z_6$, where

$$Z_5 = \sum_{n\in\tau_i(t-1)}(\mu(x_n^T\theta_{I_n^0}) - \mathbb{E}[\mu(x_t^T\theta_{I_t^0})|F_{t-1}, I_t])x_n$$

and

$$Z_6 = \sum_{n\in\tau_i(t-1)}(\mathbb{E}[\mu(x_t^T\theta_{I_t^0})|F_{t-1}, I_t] - \mu((1-\alpha)x_n^T\theta_K))x_n.$$

For any context $x \in \mathcal{D}$ and arm $i \neq K$, we have

$$|x^T(\hat{\theta}_{t,i} - (1-\alpha)\theta_K)| \le |x^T G_{t,i}^{-1}Z_2| + |x^T G_{t,i}^{-1}Z_5| + |x^T G_{t,i}^{-1}Z_6|. \tag{B.107}$$

Since we have $\left\{\mu(x_k^T\theta_{I_k^0}) - \mathbb{E}[\mu(x_k^T\theta_{I_k^0})|F_{k-1}, I_k]\right\}_{k\in\tau_i(t-1)}$ is a bounded martingale difference sequence w.r.t the filtration $\{F_k, I_k\}_{k\in\tau_i(t-1)}$ and is also $k_\mu LS$-sub-Gaussian-sub-Gaussian.

From Theorem 1 and Lemma 11 in [1], we know that for any $\delta > 0$, with probability at least $1 - \frac{K-1}{K}\delta$

$$\|Z_5\|_{\bar{V}_{t,i}^{-1}} \le 2k_\mu LS\sqrt{\log\frac{K}{\delta} + d\log\left(1 + \frac{L^2N_i(t)}{\lambda_0 d}\right)}, \tag{B.108}$$

for any arm $i \neq K$ and all $t > 0$.

Similar with the equation (B.5) in Section B.1.2, we have

$$\begin{aligned}\|x_t\|^2_{G_{t,i}^{-1}} &= x_t^T G_{t,i}^{-1} G_{t,i} G_{t,i}^{-1} x_t \\ &\geq c_\mu x_t^T V_{t,i}^{-1} \left( \sum_{k \in \tau_i(t-1)} x_k x_k^T \right) V_{t,i}^{-1} x_t \\ &= c_\mu \sum_{k \in \tau_i(t-1)} (x_t^T G_{t,i}^{-1} x_k)^2.\end{aligned} \tag{B.109}$$

and hence $\sum_{k \in \tau_i(t-1)} (x_t^T G_{t,i}^{-1} x_k)^2 \leq \frac{1}{c_\mu^2} \|x_t\|^2_{\bar{V}_{t,i}^{-1}}$.

Then, $|x^T G_{t,i}^{-1} Z_6|$ can be upper bounded by

$$\begin{aligned}&|x^T G_{t,i}^{-1} Z_6| \\ \leq &\sqrt{\sum_{k \in \tau_i(t-1)} \left( \mathbb{E}[r_{k,I_k^0} | F_{k-1}, I_k] - \mu((1-\alpha) x_k^T \theta_K) \right)^2} \sqrt{\sum_{k \in \tau_i(t-1)} (x_t^T G_{t,i}^{-1} x_k)^2} \\ \leq &\left( \sum_{k \in \tau_i(t-1)} \left( 2k_\mu \beta^0_{k,K} \|x_k\|_{\left(\bar{V}^0_{k,K}\right)^{-1}} + 2k_\mu \beta^0_{k,I_k^\dagger} \|x_k\|_{\left(\bar{V}^0_{k,I_k^\dagger}\right)^{-1}} \right)^2 \right)^{\frac{1}{2}} \frac{1}{c_\mu} \|x_t\|_{\bar{V}_{t,i}^{-1}},\end{aligned} \tag{B.110}$$

where the first inequality is obtained from Cauchy-Schwarz inequality, the second inequality is obtained from Lemma 8 and (B.5).

In addition, by the fact that $(a+b)^2 \leq 2a^2 + 2b^2$ for any real number, we have

$$\begin{aligned}&\sum_{k \in \tau_i(t-1)} \left( 2k_\mu \beta^0_{k,K} \|x_k\|_{\left(\bar{V}^0_{k,K}\right)^{-1}} + 2k_\mu \beta^0_{k,I_k^\dagger} \|x_k\|_{\left(\bar{V}^0_{k,I_k^\dagger}\right)^{-1}} \right)^2 \\ \leq &\sum_{k \in \tau_i(t-1)} 2 \left( 2k_\mu \beta^0_{k,K} \|x_k\|_{\left(\bar{V}^0_{k,K}\right)^{-1}} \right)^2 + \sum_{k \in \tau_i(t-1)} 2 \left( 2k_\mu \beta^0_{k,I_k^\dagger} \|x_k\|_{\left(\bar{V}^0_{k,I_k^\dagger}\right)^{-1}} \right)^2.\end{aligned} \tag{B.111}$$

Here, we use Lemma 11 from [1] and get, for any arm $i$,

$$\sum_{k\in\tau_i^\dagger(t-1)} \|x_k\|^2_{(V^0_{k,i})^{-1}} \leq 2d\log\left(1+\frac{N_i(t)L^2}{d\lambda}\right)$$
$$\leq 2d\log\left(1+\frac{tL^2}{d\lambda}\right). \tag{B.112}$$

Set $\lambda = \lambda_0$, we have $\|x\|^2_{\bar{V}_{t,i}^{-1}} \leq (1+\frac{\lambda}{\lambda_0})\|x\|^2_{V_{t,i}^{-1}} \leq 2\|x\|^2_{V_{t,i}^{-1}}$.

By the fact that $\sum_i \tau_i(t-1) = \tau_K^\dagger(t-1)$, and $\sum_{i\neq K}\tau_i(t-1) = \sum_{i\neq K}\tau_i^\dagger(t-1)$, we have, for any arm $i$, $\tau_i(t-1) \subseteq \tau_K^\dagger(t-1)$, and $\tau_i(t-1) \subseteq \sum_{j\neq K}\tau_j^\dagger(t-1)$. Thus,

$$\sum_{k\in\tau_i(t-1)} \|x_k\|^2_{\left(\bar{V}^0_{k,K}\right)^{-1}} \leq \sum_{k\in\tau_K^\dagger(t-1)} \|x_k\|^2_{(\bar{V}^0_{k,K})^{-1}}$$
$$\leq 4d\log\left(1+\frac{tL^2}{d\lambda_0}\right), \tag{B.113}$$

and

$$\sum_{k\in\tau_i(t-1)} \|x_k\|^2_{\left(\bar{V}^0_{k,I_k^\dagger}\right)^{-1}} \leq \sum_{i\neq K}\sum_{k\in\tau_i^\dagger(t-1)} \|x_k\|^2_{(\bar{V}^0_{k,i})^{-1}}$$
$$\leq 4(K-1)d\log\left(1+\frac{tL^2}{d\lambda_0}\right). \tag{B.114}$$

By combining (B.111), (B.113) and (B.114) and when $K \geq 3$, we have

$$
\begin{aligned}
&\sum_{k\in\tau_i(t-1)} \left( 2k_\mu \beta^0_{k,K} \|x_k\|_{\left(V^0_{k,K}\right)^{-1}} + 2k_\mu \beta^0_{k,I_k^\dagger} \|x_k\|_{\left(V^0_{k,I_k^\dagger}\right)^{-1}} \right)^2 \\
\leq &\sum_{k\in\tau_i(t-1)} 2\left( 2k_\mu \beta^0_{k,K} \|x_k\|_{\left(V^0_{k,K}\right)^{-1}} \right)^2 + \sum_{k\in\tau_i(t-1)} 2\left( 2k_\mu \beta^0_{k,I_k^\dagger} \|x_k\|_{\left(V^0_{k,I_k^\dagger}\right)^{-1}} \right)^2 \\
\leq &\left( 2\phi_K \frac{k_\mu LS + R}{c_\mu} \right)^2 \left( \log\frac{K}{\delta} + d\log\left(1 + \frac{L^2 t}{\lambda_0 d}\right) \right) \\
&\times 8k_\mu^2 \times 16d^2 \log\left(1 + \frac{tL^2}{d\lambda_0}\right) \\
&+ \left( 2\frac{K_\mu}{c_\mu \alpha} \frac{k_\mu LS + R}{c_\mu} \right)^2 \left( \log\frac{K}{\delta} + d\log\left(1 + \frac{L^2 t}{\lambda_0 d}\right) \right) \qquad \text{(B.115)} \\
&\times 8k_\mu^2 \times 16d^2 (K-1) \log\left(1 + \frac{tL^2}{d\lambda_0}\right) \\
\leq &128 k_\mu^2 d^2 \left( \frac{2k_\mu LS + 2R}{c_\mu} \right)^2 \left( \log\frac{K}{\delta} + d\log\left(1 + \frac{L^2 t}{\lambda_0 d}\right) \right) \\
&\times \log\left(1 + \frac{tL^2}{d\lambda_0}\right) \times \left( (K-1)\frac{k_\mu^2}{c_\mu^2 \alpha^2} + (1 + \frac{k_\mu}{c_\mu})^2 \right) \\
\leq &256 k_\mu^2 d^2 \left( \frac{2k_\mu LS + 2R}{c_\mu} \right)^2 \left( \log\frac{K}{\delta} + d\log\left(1 + \frac{L^2 t}{\lambda_0 d}\right) \right) \\
&\times \log\left(1 + \frac{tL^2}{d\lambda_0}\right) \times K\frac{k_\mu^2}{c_\mu^2 \alpha^2}
\end{aligned}
$$

In summary, we have

$$
\begin{aligned}
&|x_t^T \hat{\theta}_{t,i} - x_t^T (1-\alpha)\theta_K| \\
\leq &\left( 1 + \frac{16k_\mu^2 d}{c_\mu \alpha} \sqrt{K \log\left(1 + \frac{tL^2}{d\lambda_0}\right)} \right) \frac{2k_\mu LS + 2R}{c_\mu} \sqrt{\log\frac{K}{\delta} + d\log\left(1 + \frac{L^2 t}{\lambda_0 d}\right)} \|x\|_{\bar{V}_{t,i}^{-1}}.
\end{aligned}
\tag{B.116}
$$

### B.3.6 Proof of Theorem 7

For round $t$ and context $x_t$, if GLM-UCB pulls arm $i \neq K$, we have

$$x_t^T \hat{\theta}_{t,K} + \beta_{t,K} \sqrt{x_t^T \bar{V}_{t,K}^{-1} x_t} \leq x_t^T \hat{\theta}_{t,i} + \beta_{t,i} \sqrt{x_t^T \bar{V}_{t,i}^{-1} x_t}.$$

Recall $\beta_{t,i} = \frac{4R}{c_\mu} \sqrt{\log \frac{K}{\delta} + d \log \left(1 + \frac{L^2 N_i(t)}{\lambda_0 d}\right)}$.

Since the attacker does not attack the target arm, the confidence bound of arm $K$ does not change and $x_t^T \theta_K \leq x_t^T \hat{\theta}_{t,K} + \beta_{t,K} \sqrt{x_t^T V_{t,K}^{-1} x_t}$ holds with probability $1 - \frac{\delta}{K}$.

Thus, by Lemma (13),

$$\begin{aligned} & x_t^T \theta_K \\ \leq & x_t^T \hat{\theta}_{t,i} + \beta_{t,i} \sqrt{x_t^T V_{t,i}^{-1} x_t} \\ \leq & x_t^T (1-\alpha) \theta_K + \frac{2k_\mu L S + 2R}{c_\mu} \left(1 + \frac{16 k_\mu^2 d}{c_\mu \alpha} \sqrt{K \log \left(1 + \frac{tL^2}{d\lambda_0}\right)}\right) \sqrt{\log \frac{K}{\delta} + d \log \left(1 + \frac{L^2 t}{\lambda_0 d}\right)} \|x\|_{\bar{V}_{t,i}^{-1}}. \end{aligned} \tag{B.117}$$

By multiplying both sides $\mathbb{1}_{\{I_t = i\}}$ and summing over rounds, we have

$$\begin{aligned} & \sum_{k=1}^{t} \mathbb{1}_{\{I_k = i\}} \alpha x_k^T \theta_K \\ \leq & \sum_{k=1}^{t} \mathbb{1}_{\{I_k = i\}} \left(1 + \frac{16 k_\mu^2 d}{c_\mu \alpha} \sqrt{K \log \left(1 + \frac{tL^2}{d\lambda_0}\right)}\right) \frac{2k_\mu L S + 2R}{c_\mu} \sqrt{\log \frac{K}{\delta} + d \log \left(1 + \frac{L^2 t}{\lambda_0 d}\right)} \|x\|_{\bar{V}_{t,i}^{-1}}. \end{aligned} \tag{B.118}$$

Here, we use Lemma 11 from [1] and obtain

$$\begin{aligned} \sum_{k=1}^{t} \mathbb{1}_{\{I_k = i\}} \|x_k\|_{V_{k,i}^{-1}}^2 & \leq 2d \log(1 + \frac{N_i(t) L^2}{d\lambda}) \\ & \leq 2d \log \left(1 + \frac{tL^2}{d\lambda}\right). \end{aligned} \tag{B.119}$$

According to $\sum_{k=1}^{t} \mathbb{1}_{\{I_k=i\}}\|x_k\|_{V_{k,i}^{-1}} \le \sqrt{N_i(t)\sum_{k=1}^{t} \mathbb{1}_{\{I_k=i\}}\|x_k\|^2_{V_{k,i}^{-1}}}$, we have

$$\sum_{k=1}^{t} \mathbb{1}_{\{I_k=i\}}\|x_k\|_{V_{k,i}^{-1}} \le \sqrt{N_i(t)2d\log\left(1+\frac{tL^2}{d\lambda}\right)}. \tag{B.120}$$

Set $\lambda = \lambda_0$, we have $\|x\|^2_{\bar{V}_{t,i}^{-1}} \le (1+\frac{\lambda}{\lambda_0})\|x\|^2_{V_{t,i}^{-1}} \le 2\|x\|^2_{V_{t,i}^{-1}}$. Thus, we have

$$\begin{aligned}
&\sum_{k=1}^{t} \mathbb{1}_{\{I_k=i\}}\alpha x_k^T\theta_K \\
\le&\sum_{k=1}^{t} \mathbb{1}_{\{I_k=i\}}\left(1+\frac{16k_\mu^2 d}{c_\mu\alpha}\sqrt{K\log\left(1+\frac{tL^2}{d\lambda_0}\right)}\right)\times \\
&\frac{2k_\mu LS+2R}{c_\mu}\sqrt{\log\frac{K}{\delta}+d\log\left(1+\frac{L^2t}{\lambda_0 d}\right)}\sqrt{4N_i(t)d\log\left(1+\frac{tL^2}{d\lambda_0}\right)},
\end{aligned} \tag{B.121}$$

and

$$\begin{aligned}
N_i(t)=&\sum_{k=1}^{t} \mathbb{1}_{\{I_k=i\}} \\
\le&\frac{4d}{(\alpha\gamma)^2}\left(\frac{2k_\mu LS+2R}{c_\mu}\right)^2\log\left(1+\frac{tL^2}{d\lambda_0}\right)\times \\
&\left(\log\frac{K}{\delta}+d\log\left(1+\frac{L^2t}{\lambda_0 d}\right)\right)\left(1+\frac{16k_\mu^2 d}{c_\mu\alpha}\sqrt{K\log\left(1+\frac{tL^2}{d\lambda_0}\right)}\right)^2,
\end{aligned} \tag{B.122}$$

where $\gamma = \min_{x\in\mathcal{D}}\langle x, \theta_K\rangle$.

# Appendix C

# Appendix of Chapter 4

## C.1 Proofs for the white-box attack

### C.1.1 Proof of Lemma 1

We assume that the agent does not know the attacker's manipulations and the presence of the attacker. We can consider the combination of the attack and the environment as a new environment, and the RL agent interacts with the new environment in the attack setting. We define $\overline{Q}$ and $\overline{V}$ as the $Q$-values and value functions of the new environment that the RL agent observes. The optimal policy can be given from the the Bellman optimality equations. Suppose the target policy $\pi^{\dagger}$ is optimal at step $h+1$ in the observation of the agent. Then, $\overline{V}^{*}_{h+1}(s) = \overline{V}^{\dagger}_{h+1}(s)$ for all state $s$, where $\overline{V}$ represents the value function in the observation of the agent. Similarly, we set $\overline{Q}$ as the $Q$-values in the observation of the agent. As the attacker does not attack when the agent pick the target action, $\overline{V}^{\dagger}_{h+1} = V^{\dagger}_{h+1}$. For any $a \neq \pi^{\dagger}_{h}(s)$, from the equation (4.3), (4.4) and (4.6), $\overline{Q}^{*}_{h}$ is given by

$$\begin{aligned}\overline{Q}_h^*(s,a) =&(1-\alpha)(R_h(s,\pi_h^\dagger(s)) + P_h\overline{V}_{h+1}^*(s,\pi_h^\dagger(s)))\\&+\alpha(R_h(s,\pi_h^-(s)) + P_h\overline{V}_{h+1}^*(s,\pi_h^-(s)))\\=&(1-\alpha)Q_h^\dagger(s,\pi_h^\dagger(s)) + \alpha Q_h^\dagger(s,\pi_h^-(s))\\<&Q_h^\dagger(s,\pi_h^\dagger(s)) = V_h^\dagger(s) = \overline{Q}_h^\dagger(s,\pi_h^\dagger(s)).\end{aligned} \tag{C.1}$$

We can conclude that if the target policy $\pi^\dagger$ is optimal at step $h+1$ in the observation of the agent, the target policy $\pi^\dagger$ is also optimal at step $h$ in the observation of the agent. Since $V_{H+1}^\pi = \mathbf{0}$ and $Q_{H+1}^\pi = \mathbf{0}$, the target policy $\pi^\dagger$ is the optimal policy, from induction on $h = H, H-1, \cdots, 1$.

### C.1.2 Proof of Theorem 8

Here, we follows the idea of error decomposition proposed in [35, 111]. We first decomposed the expected regret Regret$(K)$ into the gap of $Q$-values. Denote by $\Delta_h^k = V_h^\dagger(s_h^k) - \min_{a\in\mathcal{A}} Q_h^\dagger(s_h^k, a)$ and $\overline{\Delta}_h^k = \overline{V}_h^\dagger(s_h^k) - \overline{Q}_h^\dagger(s_h^k, a_h^k)$.

As shown in Lemma 1, the target policy $\pi^\dagger$ is optimal in the observation of the agent. Thus,

$$\text{Regret}(K) = \sum_{k=1}^K [\overline{V}_1^*(s_1^k) - \overline{V}_1^{\pi^k}(s_1^k)] = \sum_{k=1}^K [\overline{V}_1^\dagger(s_1^k) - \overline{V}_1^{\pi^k}(s_1^k)]. \tag{C.2}$$

For episode $k$,

$$\begin{aligned}
&\overline{V}_1^{\dagger}(s_1^k) - \overline{V}_1^{\pi^k}(s_1^k) \\
=&\overline{V}_1^{\dagger}(s_1^k) - \mathbb{E}_{a\sim\pi_1^k(\cdot|s_1^k)}[\overline{Q}_1^{\dagger}(s_1^k, a)|\mathcal{F}_1^k] + \mathbb{E}_{a\sim\pi_1^k(\cdot|s_1^k)}[\overline{Q}_1^{\dagger}(s_1^k, a)|\mathcal{F}_1^k] - \overline{V}_1^{\pi^k}(s_1^k) \\
=&\mathbb{E}[\overline{\Delta}_1^k|\mathcal{F}_1^k] + \mathbb{E}_{s'\sim P_1(\cdot|s_1^k, a\sim\pi_1^k(\cdot|s_1^k))}[(\overline{V}_2^{\dagger} - \overline{V}_2^{\pi^k})(s')] \\
=&\cdots = \mathbb{E}[\sum_{h=1}^{H}\overline{\Delta}_h^k|\mathcal{F}_1^k] \qquad \text{(C.3)} \\
\overset{①}{=}&\mathbb{E}\left[\sum_{h=1}^{H}\alpha\Delta_h^k\mathbb{1}(a_h^k \neq \pi_h^{\dagger}(s_h^k))|\mathcal{F}_1^k\right] \\
\geq&\alpha\Delta_{min}\mathbb{E}\left[\sum_{h=1}^{H}\mathbb{1}\left(\widetilde{a}_h^k \neq a_h^k\right)\right],
\end{aligned}$$

where $\mathcal{F}_h^k$ represents the $\sigma$-field generated by all the random variables until episode $k$, step $h$ begins, and the equation ① holds due to $\overline{Q}_h^{\dagger}(s_h^k, a_h^k) = (1-\alpha)Q_h^{\dagger}(s_h^k, \pi_h^{\dagger}(s_h^k)) + \alpha Q_h^{\dagger}(s_h^k, \pi_h^{-}(s_h^k))$ when $a_h^k \neq \pi_h^{\dagger}(s_h^k)$, and $\overline{V}_h^{\dagger}(s_h^k) = V_h^{\dagger}(s_h^k)$.

In the $\alpha$-portion attack, the attacker attacks only when the agent picks a non-target arm. Thus, $\mathbb{1}\left(\widetilde{a}_h^k \neq a_h^k\right) \leq \mathbb{1}\left(a_h^k \neq \pi_h^{\dagger}(s_h^k)\right)$ and Cost$(K, H) \leq$ Loss$(K, H)$.

We can conclude that

$$\mathbb{E}[\text{Cost}(K, H)] \leq \mathbb{E}[\text{Loss}(K, H)] \leq \frac{\text{Regret}(K)}{\alpha\Delta_{min}}. \qquad \text{(C.4)}$$

Before the proof of the upper bound on the loss and the cost, we first introduce an important lemma, which shows the connections between the expected regret to the loss and the cost.

**Lemma 3.** For any MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, H, P, R)$ and any $p \in (0, 1)$, with probability at least $1 - p$, we have

$$\sum_{k=1}^{K}\sum_{h=1}^{H}\overline{\Delta}_h^k \leq \sum_{k=1}^{K}\left(\overline{V}_h^{\dagger}(s_h^k) - \overline{V}_h^{\pi^k}(s_h^k)\right) + 2H^2\sqrt{\log(1/p)\sum_{k=1}^{K}\left(\overline{V}_h^{\dagger}(s_h^k) - \overline{V}_h^{\pi^k}(s_h^k)\right)}. \qquad \text{(C.5)}$$

The proof of Lemma 3 is based on the Freedman inequality [25, 100]. Since $\mathbb{E}[\sum_{h=1}^{H}\overline{\Delta}_h^k|\mathcal{F}_1^k] =$

$\overline{V}_1^{\dagger}(s_1^k) - \overline{V}_1^{\pi^k}(s_1^k)$, denote by $X_k = \sum_{h=1}^{H} \overline{\Delta}_h^k - \left(\overline{V}_h^{\dagger}(s_h^k) - \overline{V}_h^{\pi^k}(s_h^k)\right)$, then $\{X_k\}_{k=1}^K$ is a martingale difference sequence w.r.t the filtration $\{\mathcal{F}_1^k\}_{k\geq 1}$. The difference sequence is uniformly bounded by $|X_k^2| \leq H^2$. Define the predictable quadratic variation process of the martingale $W_K := \sum_{k=1}^{K} \mathbb{E}[X_k^2|\mathcal{F}_1^k]$, which is bounded by

$$W_K \leq \sum_{k=1}^{K} \mathbb{E}\left[\left(\overline{\Delta}_h^k\right)^2 |\mathcal{F}_1^k\right] \leq \sum_{k=1}^{K} H^2 \mathbb{E}\left[\overline{\Delta}_h^k|\mathcal{F}_1^k\right] = \sum_{k=1}^{K} H^2\left(\overline{V}_h^{\dagger}(s_h^k) - \overline{V}_h^{\pi^k}(s_h^k)\right). \tag{C.6}$$

By the Freedman's inequality, we have

$$\begin{aligned}
&\mathbb{P}\left(\sum_{k=1}^{K} X_k > 2H^2\sqrt{\log(1/p)\sum_{k=1}^{K}\left(\overline{V}_h^{\dagger}(s_h^k) - \overline{V}_h^{\pi^k}(s_h^k)\right)}\right) \\
&\leq \exp\left\{\frac{-2H^4\log(1/p)\sum_{k=1}^{K}\left(\overline{V}_h^{\dagger}(s_h^k) - \overline{V}_h^{\pi^k}(s_h^k)\right)}{W_K + H^2 * 2H^2\sqrt{\log(1/p)\sum_{k=1}^{K}\left(\overline{V}_h^{\dagger}(s_h^k) - \overline{V}_h^{\pi^k}(s_h^k)\right)}/3}\right\} \\
&\leq \exp\left\{-\log(1/p)\right\} = p.
\end{aligned} \tag{C.7}$$

Theorem 8 is directly from Lemma 3 and $\overline{\Delta}_h^k \geq \alpha\Delta_{min}\mathbb{1}(\pi_h^{\dagger}(s_h^k) \neq \pi_h^k(s_h^k))$.

## C.2 Proofs for LCB-H attack

### C.2.1 Proof of Lemma 2

At the beginning of the episode $k$, for any step $h \in [H]$ and any $(s,a) \in \mathcal{S} \times \mathcal{A}$ with $N_h^k(s,a) \neq 0$, according to Algorithm 2.1, the estimate of $Q$-values under the target policy $\pi^{\dagger}$ are given by

$$\hat{Q}_{h,k}^{\dagger}(s,a) = \frac{1}{N_h^k(s,a)} \sum_{i=1}^{k-1} \mathbb{1}\left((\widetilde{a}_h^k, s_h^k) = (s,a)\right)\left(r_h^k + \rho_{h+1:H+1}^k G_{h+1:H+1}^k\right). \tag{C.8}$$

Note that for any $\big((\widetilde{a}_h^k, s_h^k) = (s,a)\big)$, we have

$$\mathbb{E}[r_h^k + \rho_{h+1:H+1}^k G_{h+1:H+1}^k | \widetilde{a}_h^k, s_h^k] = R_h(s,a) + \mathbb{E}_{s' \sim P_h(\cdot|s,a)}[V_{h+1}^{\dagger}(s')] = Q_h^{\dagger}(s,a). \tag{C.9}$$

Thus, we can apply Hoeffding's inequality here to bound $|\hat{Q}_{h,k}^{\dagger}(s,a) - Q_h^{\dagger}(s,a)|$. The cumulative reward is bounded by $0 \leq G_{h+1:H+1}^k \leq H - h$ and the important sampling ratio is bounded by $0 \leq \rho_{h+1:H+1}^k \leq e$ because

$$\rho_{h+1:H+1}^k \leq \left(\frac{1}{(1-\frac{1}{H})}\right)^{H-h} \leq \left(\frac{1}{(1-\frac{1}{H})}\right)^{H-1} \leq e. \tag{C.10}$$

By Hoeffding's inequality, since $|r_h^k + \rho_{h+1:H+1}^k G_{h+1:H+1}^k| \leq e(H-h)+1$, we have

$$\mathbb{P}\left(|\hat{Q}_{h,k}^{\dagger}(s,a) - Q_h^{\dagger}(s,a)| > \eta\right) \leq 2\exp\left(-\frac{\eta^2}{2N_h^k(s,a)\left(\frac{H-h+1}{N_h^k(s,a)}\right)^2}\right). \tag{C.11}$$

To hold a high-probability confidence bound for any state $s$, any action $a$, any step $h$ and any episode $k$, set the right hand side of the above inequality to $p/SAT$. Then, we have $\eta = (e(H-h)+1)\sqrt{\frac{2\iota}{N_h^k(s,a)}}$ and $\iota = \log(2SAT/p)$.

### C.2.2 Proof of Theorem 9

From Lemma 3, for any MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, H, P, R)$ and any $p \in (0,1)$, with probability at least $1-p$, we have

$$\begin{aligned}\sum_{k=1}^{K}\sum_{h=1}^{H}\overline{\Delta}_h^k \leq & \sum_{k=1}^{K}\left(\overline{V}_h^{k,\dagger}(s_h^k) - \overline{V}_h^{k,\pi^k}(s_h^k)\right) + 2H^2\sqrt{\log(1/p)\sum_{k=1}^{K}\left(\overline{V}_h^{\dagger}(s_h^k) - \overline{V}_h^{k,\pi^k}(s_h^k)\right)} \\ \leq & \sum_{k=1}^{K}\left(\overline{V}_h^{k,*}(s_h^k) - \overline{V}_h^{k,\pi^k}(s_h^k)\right) + 2H^2\sqrt{\log(1/p)\sum_{k=1}^{K}\left(\overline{V}_h^{k,*}(s_h^k) - \overline{V}_h^{k,\pi^k}(s_h^k)\right)} \\ = & \text{D-Regret}(K) + 2H^2\sqrt{\log(1/p)\text{D-Regret}(K)}.\end{aligned} \tag{C.12}$$

Since the LCB-H attacker dose not attack the target action, $\overline{V}_h^{k,\dagger}(s_h^k) = V_h^\dagger(s_h^k)$. Thus, we have $\overline{\Delta}_h^k = \overline{V}_h^{k,\dagger}(s_h^k) - \overline{Q}_h^{k,\dagger}(s_h^k, a_h^k) = V_h^\dagger(s_h^k) - \overline{Q}_h^{k,\dagger}(s_h^k, a_h^k)$. When the agent picks a target action $a_h^k = \pi^\dagger(s_h^k)$, the attacker does not attack and $\overline{Q}_h^{k,\dagger}(s_h^k, a_h^k) = \overline{V}_h^{k,\dagger}(s_h^k) = V_h^\dagger(s_h^k)$. Thus, the left hand side of the equation (C.12) can be written as

$$\sum_{k=1}^{K}\sum_{h=1}^{H}\overline{\Delta}_h^k = \sum_{k=1}^{K}\sum_{h=1}^{H}\mathbb{1}(a_h^k \neq \pi_h^\dagger(s_h^k))\overline{\Delta}_h^k = \sum_{(k,h)\in\tau}\overline{\Delta}_h^k, \tag{C.13}$$

where $\tau = \{(k,h) \in [K]\times[H] | a_h^k \neq \pi_h^\dagger(s_h^k)\}$.

At episode $k$ and step $h$, after the agent picks an action, since the attack scheme is given, we have $\overline{Q}_h^{k,\dagger}(s_h^k, a_h^k) = \mathbb{E}[Q_h^\dagger(s_h^k, \widetilde{a}_h^k)|\mathcal{F}_1^k, s_h^k, a_h^k]$. Furthermore, $\mathbb{E}[V_h^\dagger(s_h^k) - Q_h^\dagger(s_h^k, \widetilde{a}_h^k)|\mathcal{F}_1^k, s_h^k, a_h^k] = \overline{\Delta}_h^k$. By the Hoeffding inequality, since $|V_h^\dagger(s_h^k) - Q_h^\dagger(s_h^k, \widetilde{a}_h^k)| \leq H$, we have

$$\mathbb{P}\left(\sum_{(k,h)\in\tau}\left(V_h^\dagger(s_h^k) - Q_h^\dagger(s_h^k, \widetilde{a}_h^k) - \overline{\Delta}_h^k\right) > \eta\right) \leq \exp\left(-\frac{\eta^2}{2|\tau|H^2}\right). \tag{C.14}$$

Set the left hand side of the above inequality to $p$. With probability $1-p$, we have,

$$\sum_{k=1}^{K}\sum_{h=1}^{H}\overline{\Delta}_h^k \geq \sum_{(k,h)\in\tau}\left(V_h^\dagger(s_h^k) - Q_h^\dagger(s_h^k,\widetilde{a}_h^k)\right) - H\sqrt{2|\tau|\log(1/p)}. \tag{C.15}$$

If $\widetilde{a}_h^k \neq \pi_h^\dagger(s)$ holds, the attacker attacked the agent, and from Lemma 2, we have with probability $1-p$,

$$Q_h^\dagger(s,\pi_h^-(s)) \geq L_h^k(s,\pi_h^-(s)) \geq L_h^k(s,\widetilde{a}_h^k) \geq Q_h^\dagger(s,\widetilde{a}_h^k) - 2(e(H-h)+1)\sqrt{\frac{2\iota}{N_h^k(s_h^k,\widetilde{a}_h^k)}}, \tag{C.16}$$

and $0 \leq Q_h^\dagger(s,\widetilde{a}_h^k) - Q_h^\dagger(s,\pi_h^-(s)) \leq 2(e(H-h)+1)\sqrt{\frac{2\iota}{N_h^k(s_h^k,\widetilde{a}_h^k)}}$. If $\widetilde{a}_h^k \neq \pi_h^\dagger(s)$ holds, $V_h^\dagger(s_h^k) = Q_h^\dagger(s_h^k,\widetilde{a}_h^k)$. For the second item in the right hand side of inequality (C.15), we have with probability $1-p$,

$$\begin{aligned}&\sum_{(k,h)\in\tau}\left(V_h^\dagger(s_h^k) - Q_h^\dagger(s_h^k,\widetilde{a}_h^k)\right)\\ \geq &\sum_{(k,h)\in\tau}\mathbb{1}\left(\widetilde{a}_h^k \neq \pi_h^\dagger(s)\right)\left(\Delta_h^k - 2(e(H-h)+1)\sqrt{\frac{2\iota}{N_h^k(s_h^k,\widetilde{a}_h^k)}}\right).\end{aligned} \tag{C.17}$$

For $(k,h)\in\tau$, $\mathbb{E}[\mathbb{1}(\widetilde{a}_h^k \neq \pi_h^\dagger(s))|\mathcal{F}_h^k,(k,h)\in\tau] = 1/H$. By the Hoeffding inequality, we have with probability $1-p$,

$$\sum_{(k,h)\in\tau}\left|\mathbb{1}\left(\widetilde{a}_h^k \neq \pi_h^\dagger(s)\right) - 1/H\right| \leq \sqrt{2|\tau|\log(2/p)}. \tag{C.18}$$

We regroup the right hand side of inequality (C.17) in a different way and further

$$\begin{aligned}
&\sum_{(k,h)\in\tau} \mathbb{1}\left(\widetilde{a}_h^k \neq \pi_h^\dagger(s)\right)\sqrt{1/N_h^k(s_h^k,\widetilde{a}_h^k)} \\
=&\sum_{h\in[H]}\sum_{s\in\mathcal{S}}\sum_{a\neq\pi_h^\dagger(s)}\sum_{n=1}^{N_h^{K+1}(s,a)}\sqrt{1/n} \\
\leq&\sum_{h\in[H]}\sum_{s\in\mathcal{S}}\sum_{a\neq\pi_h^\dagger(s)}\left(1+\int_{n=1}^{N_h^{K+1}(s,a)}\sqrt{1/n}dn\right) \\
\leq&\sum_{h\in[H]}\sum_{s\in\mathcal{S}}\sum_{a\neq\pi_h^\dagger(s)} 2\sqrt{N_h^{K+1}(s,a)} \\
\overset{①}{\leq}&2SAH\sqrt{\frac{\sum_{h\in[H]}\sum_{s\in\mathcal{S}}\sum_{a\neq\pi_h^\dagger(s)}N_h^{K+1}(s,a)}{SAH}} \\
=&2\sqrt{SAH\sum_{(k,h)\in\tau}\mathbb{1}\left(\widetilde{a}_h^k\neq\pi_h^\dagger(s)\right)} \\
\overset{②}{\leq}&2\sqrt{SAH\left(|\tau|/H+\sqrt{2|\tau|\log(2/p)}\right)} \\
\leq&2\sqrt{SA|\tau|}+2\sqrt{2SAH|\tau|\log(2/p)},
\end{aligned} \tag{C.19}$$

where ① holds due to the property of the concave function $\sqrt{n}$ and ② holds due to the inequality (C.18).

In addition,

$$\sum_{(k,h)\in\tau}\mathbb{1}\left(\widetilde{a}_h^k\neq\pi_h^\dagger(s)\right)\Delta_h^k \geq \left(|\tau|/H-\sqrt{2|\tau|\log(2/p)}\right)\Delta_{min}. \tag{C.20}$$

Combing (C.12), (C.15), (C.17), (C.19) and (C.20), we have

$$\begin{aligned}
\Delta_{min}|\tau|/H \leq &\text{D-Regret}(K)+2H^2\sqrt{\log(1/p)\text{D-Regret}(K)}+(H+\Delta_{min})\sqrt{2|\tau|\log(1/p)} \\
&2(e(H-h)+1)\sqrt{2\iota}\left(2\sqrt{SA|\tau|}+2\sqrt{2SAH|\tau|\log(2/p)}\right),
\end{aligned} \tag{C.21}$$

which is equivalent to

$$|\tau| \leq \frac{H}{\Delta_{min}} \left(\text{D-Regret}(K) + 2H^2\sqrt{\log(1/p)\text{D-Regret}(K)}\right) + \frac{307SAH^4\iota}{\Delta_{min}^2}. \tag{C.22}$$

In addition, $\text{Cost}(K, H) \leq \text{Loss}(K, H) = \sum_{k=1}^{K}\sum_{h=1}^{H} \mathbb{1}(a_h^k \neq \pi_h^\dagger(s_h^k)) = |\tau|$. The proof is completed.

## C.3 Proof of LCB-H attacks on UCB-H

For completeness, we describe the main steps of UCB-H algorithm in Algorithm C.1.

**Algorithm C.1** Q-learning with UCB-Hoeffding [41]

```
Initialize $Q_h(s,a) = 0$ and $N_h(s,a) = 0$ for all state $s \in \mathcal{S}$, all action $a \in \mathcal{A}$ and all step $h \in [H]$.
Define $\alpha_t = \frac{H+1}{H+t}$, $\iota = log(2SAT/p)$, and set a constant $c$.
for episode $k = 1, 2, \ldots, K$ do
    Receive $s_1$.
    for step $h = 1, 2, \ldots, H$ do
        Take action $a_h \leftarrow \arg\max_{a'} Q_h(s_h, a')$, and observe $s_{h+1}$ and $r_h$.
        $t = N_h(s_h, a_h) \leftarrow N_h(s_h, a_h) + 1$; $b_t = c\sqrt{H^3\iota/t}$.
        $Q_h(s_h, a_h) = (1 - \alpha_t)Q_h(s_h, a_h) + \alpha_t[r_h + V_{h+1}(s_{h+1}) + b_t]$.
        $V_h(s_h) \leftarrow \min\{H, \max_{a'} Q_h(s_h, a')\}$.
    end for
end for
```

Before the proof of Theorem 10, we first introduce our main technical lemma.

We denote by $\overline{Q}_h^k$, $\overline{V}_h^k$, $\overline{N}_h^k$ the observations of UCB-H agent at the beginning of episode $k$. The lemma below is our main technical lemma that shows the difference between the agent's observations $\overline{Q}_h^k$ and the true $Q$-values $Q_h^\dagger$ can be bounded by quantities from the next step.

**Lemma 4.** Assume the attacker follows the LCB-H attack strategy on the UCB-H agent. Suppose the constant $c$ in UCB-H algorithm satisfies $c > 0$. Let $\beta_h(t) = (cH + 2(H-h) + 2)\sqrt{H\iota/t}$ when $t > 0$ and $\beta_h(0) = 0$ for any step $h$, and let $B_h(t) = (e(H-h)+1)\sqrt{\frac{2\iota}{t}}$ when $t > 0$ and $B_h(0) = H$ for any step $h$. For any $p \in (0, 1)$, with probability at least $1 - 3p$, the following confidence bounds hold simultaneously for all $(s, a, h, k) \in \mathcal{S} \times \mathcal{A} \times [H] \times [k]$:

$$\begin{aligned} &\sum_{i=1}^{t} \alpha_t^i \left( \overline{V}_{h+1}^{k_i}(s_{h+1}^{k_i}) - V_{h+1}^{\dagger}(s_{h+1}^{k_i}) \right) \leq \overline{Q}_h^k(s, \pi_h^{\dagger}(s)) - Q_h^{\dagger}(s, \pi_h^{\dagger}(s)) \\ &\leq \alpha_t^0 H + \sum_{i=1}^{t} \alpha_t^i \left( \overline{V}_{h+1}^{k_i}(s_{h+1}^{k_i}) - V_{h+1}^{\dagger}(s_{h+1}^{k_i}) \right) + \beta_h(t), \end{aligned} \tag{C.23}$$

and

$$\begin{aligned} &\overline{Q}_h^k(s, a) - Q_h^{\dagger}(s, \pi_h^{\dagger}(s)) = \overline{Q}_h^k(s, a) - Q_h^{\dagger}(s, \pi_h^{-}(s)) - \Delta_h(s) \\ \leq &\alpha_t^0 H + \sum_{i=1}^{t} \alpha_t^i \left( \overline{V}_{h+1}^{k_i}(s_{h+1}^{k_i}) - V_{h+1}^{\dagger}(s_{h+1}^{k_i}) \right) + \beta_h(t) \\ &+ \sum_{i=1}^{t} \alpha_t^i \mathbb{1}\left( \widetilde{a}_h^{k_i} \neq \pi_h^{\dagger}(s) \right) \left( 2B_h \left( N_h^{k_i}(s, \widetilde{a}_h^{k_i}) \right) - \Delta_h(s) \right), \end{aligned} \tag{C.24}$$

where $t = \overline{N}_h^k(s, a)$, $\Delta_h(s) := Q_h^{\dagger}(s, \pi_h^{\dagger}(s)) - Q_h^{\dagger}(s, \pi_h^{-}(s))$, and $k_1, k_2, \ldots, k_t < k$ are the episodes in which $(s, a)$ was previously taken by the agent at step $h$.

By recursing the results in Lemma 4, we can obtained Theorem 10.

### C.3.1 Proof of Lemma 4

Lemma 4 shows the result of the LCB-H attacks on the UCB-H algorithm. Thus, we need to refer the readers to some settings and the Lemma 4.1 in [41]. Note that UCB-H chooses the learning rate as $\alpha_t := \frac{H+1}{H+t}$. For notational convenience, define $\alpha_t^0 := \prod_{j=1}^{t}(1-\alpha_t)$ and $\alpha_t^i := \alpha_i \prod_{j=i+1}^{t}(1-\alpha_t)$. Here, we introduce some useful properties of $\alpha_t^i$ which were proved in [41]:

(1) $\sum_{i=1}^{t} \alpha_t^i = 1$ and $\alpha_t^0 = 0$ for $t \geq 1$;

(2) $\sum_{i=1}^{t} \alpha_t^i = 0$ and $\alpha_t^0 = 1$ for $t = 0$;

(3) $\frac{1}{\sqrt{t}} \leq \sum_{i=1}^{t} \frac{\alpha_t^i}{\sqrt{t}} \leq \frac{2}{\sqrt{t}}$ for every $t \geq 1$;

(4) $\sum_{i=1}^{t} (\alpha_t^i)^2 \leq \frac{2H}{t}$ for every $t \geq 1$;

(5) $\sum_{t=i}^{\infty} \alpha_t^i \leq (1 + \frac{1}{H})$ for every $i \geq 1$.

As shown in [41], at any $(s, a, h, k) \in \mathcal{S} \times \mathcal{A} \times [H] \times [K]$, let $t = \overline{N}_h^k(s, a)$ and suppose $(s, a)$

was previously taken by the agent at step $h$ of episodes $k_1, k_2, \ldots, k_t < k$. By the update equations in the UCB-H Algorithm and the definition of $\alpha_t^i$, we have

$$\overline{Q}_h^k(s,a) = \alpha_t^0 H + \sum_{i=1}^{t} \alpha_t^i \left( r_h^{k_i} + \overline{V}_{h+1}^{k_i}(s_{h+1}^{k_i}) + b_i \right). \tag{C.25}$$

Then we can bound the difference between $\overline{Q}_h^k$ and $Q_h^\dagger$.

$$\begin{aligned}
&\overline{Q}_h^k(s,a) - Q_h^\dagger(s, \pi_h^-(s)) \\
=&\alpha_t^0 \left( H - Q_h^\dagger(s, \pi_h^-(s)) \right) + \sum_{i=1}^{t} \alpha_t^i \left( r_h^{k_i} + \overline{V}_{h+1}^{k_i}(s_{h+1}^{k_i}) + b_i - Q_h^\dagger(s, \pi_h^-(s)) \right) \\
=&\alpha_t^0 (H - Q_h^\dagger(s, \pi_h^-(s))) + \sum_{i=1}^{t} \alpha_t^i \left( r_h^{k_i} - r_h(s, \widetilde{a}_h^{k_i}) + b_i \right) \\
&+ \sum_{i=1}^{t} \alpha_t^i \left( r_h(s, \widetilde{a}_h^{k_i}) + \overline{V}_{h+1}^{k_i}(s_{h+1}^{k_i}). - Q_h^\dagger(s, \pi_h^-(s)) \right).
\end{aligned} \tag{C.26}$$

We can rewrite the third term in the RHS of (C.26) as follows

$$\begin{aligned}
&r_h(s, \widetilde{a}_h^{k_i}) + \overline{V}_{h+1}^{k_i}(s_{h+1}^{k_i}) - Q_h^\dagger(s, \pi_h^-(s)) \\
=&r_h(s, \widetilde{a}_h^{k_i}) + \overline{V}_{h+1}^{k_i}(s_{h+1}^{k_i}) - Q_h^\dagger(s, \widetilde{a}_h^{k_i}) + Q_h^\dagger(s, \widetilde{a}_h^{k_i}) - Q_h^\dagger(s, \pi_h^-(s)) \\
=&\overline{V}_{h+1}^{k_i}(s_{h+1}^{k_i}) - P_h V_{h+1}^\dagger(s, \widetilde{a}_h^{k_i}) + Q_h^\dagger(s, \widetilde{a}_h^{k_i}) - Q_h^\dagger(s, \pi_h^-(s)) \\
=&\overline{V}_{h+1}^{k_i}(s_{h+1}^{k_i}) - V_{h+1}^\dagger(s_{h+1}^{k_i}) + V_{h+1}^\dagger(s_{h+1}^{k_i}) - P_h V_{h+1}^\dagger(s, \widetilde{a}_h^{k_i}) + Q_h^\dagger(s, \widetilde{a}_h^{k_i}) - Q_h^\dagger(s, \pi_h^-(s)).
\end{aligned} \tag{C.27}$$

As the result, the difference between $\overline{Q}_h^k$ and $Q_h^\dagger$ can be rewritten as

$$\begin{aligned}
&\overline{Q}_h^k(s,a) - Q_h^\dagger(s,\pi_h^-(s)) \\
=&\alpha_t^0(H - Q_h^\dagger(s,\pi_h^-(s)) + \sum_{i=1}^t \alpha_t^i \left(\overline{V}_{h+1}^{k_i}(s_{h+1}^{k_i}) - V_{h+1}^\dagger(s_{h+1}^{k_i})\right) \\
&+ \sum_{i=1}^t \alpha_t^i \left(r_h^{k_i} - r_h(s,\widetilde{a}_h^{k_i}) + V_{h+1}^\dagger(s_{h+1}^{k_i}) - P_h V_{h+1}^\dagger(s,\widetilde{a}_h^{k_i}) + b_i\right) \\
&+ \sum_{i=1}^t \alpha_t^i \left(Q_h^\dagger(s,\widetilde{a}_h^{k_i}) - Q_h^\dagger(s,\pi_h^-(s))\right).
\end{aligned} \tag{C.28}$$

Since $\mathbb{E}[V_{h+1}^\dagger(s_{h+1}^k)|\mathcal{F}_h^k \cup \{s_h^k, a_h^k\}] = \mathbb{E}[V_{h+1}^\dagger(s_{h+1}^k)|s_h^k, a_h^k] = P_h V_{h+1}^\dagger(s,a)$ for any state-action pair $(s_h^k, a_h^k) = (s,a)$, $\sum_{i=1}^t \alpha_t^i \left(V_{h+1}^\dagger(s_{h+1}^{k_i}) - P_h V_{h+1}^\dagger(s,\widetilde{a}_h^{k_i})\right)$ is the weighted sum of a martingale difference sequence w.r.t the filtration $\{\mathcal{F}_h^{k_i}\}_{i\geq 1}$. By Azuma-Hoeffding inequality, we have

$$\mathbb{P}\left(\left|\sum_{i=1}^t \alpha_t^i \left(V_{h+1}^\dagger(s_{h+1}^{k_i}) - P_h V_{h+1}^\dagger(s,\widetilde{a}_h^{k_i})\right)\right| \geq \eta\right) \leq 2\exp\left(-\frac{\eta^2}{2(H-h)^2\frac{2H}{t}}\right), \tag{C.29}$$

where we used $\sum_{i=1}^t (\alpha_t^i)^2 \leq \frac{2H}{t}$ a property of $\alpha_t^i$ . By setting the right hand side of the above equation to $p/(SAT)$ and $t = \overline{N}_h^k(s,a)$, we have for each fixed state-action pair $(s,a,h) \in \mathcal{S} \times \mathcal{A} \times [H]$, with probability at least $1 - p/(SAH)$, event $\mathcal{E}_1$ holds, where $\mathcal{E}_1$ is defined as

$$\begin{aligned}
\mathcal{E}_1 := &\{\forall k \in [K], \\
&\left|\sum_{i=1}^t \alpha_t^i \left(V_{h+1}^\dagger(s_{h+1}^{k_i}) - P_h V_{h+1}^\dagger(s,\widetilde{a}_h^{k_i})\right)\right| \leq (H-h)\sqrt{\frac{4H\log(2SAT/p)}{\overline{N}_h^k(s,a)}}\Bigg\}.
\end{aligned} \tag{C.30}$$

Similarly, for each fixed state-action-step pair $(s,a,h) \in \mathcal{S} \times \mathcal{A} \times [H]$, with probability at least $1 - p/(SAH)$, we have event $\mathcal{E}_2$ holds with

$$\mathcal{E}_2 := \left\{\forall k \in [K], \left|\sum_{i=1}^t \alpha_t^i \left(r_h^{k_i} - r_h(s,\widetilde{a}_h^{k_i})\right)\right| \leq \sqrt{\frac{4H\log(2SAT/p)}{\overline{N}_h^k(s,a)}}\right\}. \tag{C.31}$$

Then if the agent chooses $b_t = c\sqrt{H^3\iota/t}$ for some constant $c$ and $\iota = \log(2SAT/p)$, by the property (3) of $\alpha_t^i$, we have $b_t \leq \sum_{i=1}^{t} \alpha_t^i b_t \leq 2b_t$. Under events $\mathcal{E}_1$ and $\mathcal{E}_2$, for $t \geq 1$,the third term of the RHS of equation (C.28) can be bounded by

$$\begin{aligned} &(cH - 2(H-h) - 2)\sqrt{H\iota/t} \\ \leq &\sum_{i=1}^{t} \alpha_t^i \left( r_h^{k_i} - r_h(s, \widetilde{a}_h^{k_i}) + V_{h+1}^{\dagger}(s_{h+1}^{k_i}) - P_h V_{h+1}^{\dagger}(s, \widetilde{a}_h^{k_i}) + b_i \right) \\ \leq &(cH + 2(H-h) + 2)\sqrt{H\iota/t}. \end{aligned} \tag{C.32}$$

For notational simplicity, let $\beta_h(t) = (cH + 2(H-h) + 2)\sqrt{H\iota/t}$ when $t > 0$ and $\beta_h(0) = 0$ for any step $h$.

We split the fourth term of the RHS of equation (C.28) into two cases.

If $\widetilde{a}_h^k = \pi_h^{\dagger}(s)$ holds,we have

$$Q_h^{\dagger}(s, \widetilde{a}_h^k) - Q_h^{\dagger}(s, \pi_h^{-}(s)) = Q_h^{\dagger}(s, \pi_h^{\dagger}(s)) - Q_h^{\dagger}(s, \pi_h^{-}(s)). \tag{C.33}$$

Let $B_h(t) = (e(H-h)+1)\sqrt{\frac{2S\iota}{t}}$ when $t > 0$ and $B_h(0) = H$ for any step $h$.

If $\widetilde{a}_h^k \neq \pi_h^{\dagger}(s)$ holds, the attacker attacked the agent, and from Lemma 2, we have with probability $1-p$,

$$Q_h^{\dagger}(s, \pi_h^{-}(s)) \geq L_h^k(s, \pi_h^{-}(s)) \geq L_h^k(s, \widetilde{a}_h^k) \geq Q_h^{\dagger}(s, \widetilde{a}_h^k) - 2B_h\left(N_h^k(s, \widetilde{a}_h^k)\right), \tag{C.34}$$

and $0 \leq Q_h^{\dagger}(s, \widetilde{a}_h^k) - Q_h^{\dagger}(s, \pi_h^{-}(s)) \leq 2B_h\left(N_h^k(s, \widetilde{a}_h^k)\right)$.

If $a = \pi_h^{\dagger}(s)$, the attacker does not attack so $\widetilde{a}_h^{k_i} = a = \pi_h^{\dagger}(s)$. Then by combining (C.28) and (C.32), we have for $c \geq 2$

$$\begin{aligned} &\sum_{i=1}^{t} \alpha_t^i \left( \overline{V}_{h+1}^{k_i}(s_{h+1}^{k_i}) - V_{h+1}^{\dagger}(s_{h+1}^{k_i}) \right) \leq \overline{Q}_h^k(s, \pi_h^{\dagger}(s)) - Q_h^{\dagger}(s, \pi_h^{\dagger}(s)) \\ \leq &\alpha_t^0 H + \sum_{i=1}^{t} \alpha_t^i \left( \overline{V}_{h+1}^{k_i}(s_{h+1}^{k_i}) - V_{h+1}^{\dagger}(s_{h+1}^{k_i}) \right) + \beta_h(t). \end{aligned} \tag{C.35}$$

Since $\overline{V}^{k_i}_{H+1} = V^{\dagger}_{H+1} = 0$, from induction on $h = H, H-1, \dots, 1$, we have $\overline{V}^k_h(s) \geq \min\{\overline{Q}^k_h(s, \pi^\dagger_h(s)), H\} \geq V^\dagger_h(s)$ for all state $s$, step $h$ and episode $k$ with probability $1-2p$.

If $a \neq \pi^\dagger_h(s)$, the attacker attacks by changing the action to the target action or a possible worst action. From (C.28) and (C.32), we have

$$\begin{aligned}
&\overline{Q}^k_h(s,a) - Q^\dagger_h(s, \pi^-_h(s)) \\
\leq &\alpha^0_t H + \sum_{i=1}^t \alpha^i_t \left( \overline{V}^{k_i}_{h+1}(s^{k_i}_{h+1}) - V^\dagger_{h+1}(s^{k_i}_{h+1}) \right) + \beta_h(t) \\
&+ \sum_{i=1}^t \alpha^i_t \mathbb{1}\left( \widetilde{a}^{k_i}_h = \pi^\dagger_h(s) \right) \left( Q^\dagger_h(s, \pi^\dagger_h(s)) - Q^\dagger_h(s, \pi^-_h(s)) \right) \\
&+ \sum_{i=1}^t \alpha^i_t \mathbb{1}\left( \widetilde{a}^{k_i}_h \neq \pi^\dagger_h(s) \right) 2B_h\left( N^{k_i}_h(s, \widetilde{a}^{k_i}_h) \right),
\end{aligned} \tag{C.36}$$

and

$$\begin{aligned}
&\overline{Q}^k_h(s,a) - Q^\dagger_h(s, \pi^\dagger_h(s)) = \overline{Q}^k_h(s,a) - Q^\dagger_h(s, \pi^-_h(s)) - \Delta_h(s) \\
\leq &\alpha^0_t H + \sum_{i=1}^t \alpha^i_t \left( \overline{V}^{k_i}_{h+1}(s^{k_i}_{h+1}) - V^\dagger_{h+1}(s^{k_i}_{h+1}) \right) + \beta_h(t) \\
&+ \sum_{i=1}^t \alpha^i_t \mathbb{1}\left( \widetilde{a}^{k_i}_h \neq \pi^\dagger_h(s) \right) \left( 2B_h\left( N^{k_i}_h(s, \widetilde{a}^{k_i}_h) \right) - \Delta_h(s) \right),
\end{aligned} \tag{C.37}$$

where $\Delta_h(s) := Q^\dagger_h(s, \pi^\dagger_h(s)) - Q^\dagger_h(s, \pi^-_h(s))$.

### C.3.2 Proof of Theorem 10

In this section, we assume the two events $\mathcal{E}_1$, $\mathcal{E}_2$ hold. For any state $s \in \mathcal{S}$ and any step $h \in [H]$, Lemma 4 shows that in the agent's observations, $\overline{Q}^k_h(s, \pi^\dagger_h(s)) \geq Q^\dagger_h(s, \pi^\dagger_h(s))$ for all episodes $k \in [K]$ with probability $1-3p$. Since UCB-H takes action by the function $a^k_h = \arg\max_{a \in \mathcal{A}} \overline{Q}^k_h(s^k_h, a)$, we have that with probability $1-3p$, $\overline{Q}^k_h(s^k_h, a^k_h) \geq \overline{Q}^k_h(s^k_h, \pi^\dagger_h(s^k_h)) \geq Q^\dagger_h(s^k_h, \pi^\dagger_h(s^k_h))$ for all episodes $k \in [K]$ and all steps $h \in [H]$. Thus, we can bound the loss and cost functions by

$$\begin{aligned}
&\sum_{k=1}^{K}\sum_{h=1}^{H} \mathbb{1}\left(a_h^k \neq \pi^\dagger(s_h^k)\right) \Delta_h(s_h^k) \\
=&\sum_{k=1}^{K}\sum_{h=1}^{H} \mathbb{1}\left(a_h^k \neq \pi^\dagger(s_h^k)\right) \left(Q_h^\dagger(s_h^k, \pi_h^\dagger(s_h^k)) - Q_h^\dagger(s_h^k, \pi_h^-(s_h^k))\right) \\
\leq&\sum_{k=1}^{K}\sum_{h=1}^{H} \mathbb{1}\left(a_h^k \neq \pi^\dagger(s_h^k)\right) \left(\overline{Q}_h^k(s_h^k, a_h^k) - Q_h^\dagger(s_h^k, \pi_h^-(s_h^k))\right).
\end{aligned} \tag{C.38}$$

First consider a fixed step $h$. The contribution of step $h$ to the loss function can be written as $\text{Loss}_h(K) = \sum_{k=1}^{K} \mathbb{1}\left(a_h^k \neq \pi^\dagger(s_h^k)\right)$. For notational convenience, denote

$$\phi_{h,h}^k := \mathbb{1}\left(a_h^k \neq \pi^\dagger(s_h^k)\right) \text{ and } \delta_h^k := \overline{Q}_h^k(s_h^k, a_h^k) - Q_h^\dagger\left(s_h^k, \pi_h^\dagger(s_h^k)\right). \tag{C.39}$$

From the update equation of $V$-values in UCB-H algorithm, we have

$$\overline{V}_h^k(s_h^k) - \overline{V}_h^\dagger(s_h^k) = \min\{H, \max_{a\in\mathcal{A}} \overline{Q}_h^k(s_h^k, a)\} - \overline{V}_h^\dagger(s_h^k) \leq \delta_h^k. \tag{C.40}$$

From Lemma 4, with probability $1 - 3p$, we have

$$\begin{aligned}
\sum_{k=1}^{K} \phi_{h,h}^k \delta_h^k \leq& \sum_{k=1}^{K} \phi_{h,h}^k \alpha_{\overline{N}_h^k(s_h^k,a_h^k)}^0 H + \sum_{k=1}^{K} \phi_{h,h}^k \beta_h\left(\overline{N}_h^k(s_h^k, a_h^k)\right) \\
&+ \sum_{k=1}^{K} \phi_{h,h}^k \sum_{i=1}^{\overline{N}_h^k(s_h^k,a_h^k)} \alpha_{\overline{N}_h^k(s_h^k,a_h^k)}^i \delta_{h+1}^{k_i(s_h^k,a_h^k,h)} \\
&+ \sum_{k=1}^{K} \phi_{h,h}^k \sum_{i=1}^{\overline{N}_h^k(s_h^k,a_h^k)} \alpha_{\overline{N}_h^k(s_h^k,a_h^k)}^i \mathbb{1}\left(\widetilde{a}_h^{k_i(s_h^k,a_h^k,h)} \neq \pi_h^\dagger(s_h^{k_i(s_h^k,a_h^k,h)})\right) \cdot \\
&\quad \left(2B_h\left(N_h^{k_i(s_h^k,a_h^k,h)}(s_h^{k_i(s_h^k,a_h^k,h)}, \widetilde{a}_h^{k_i(s_h^k,a_h^k,h)})\right) - \Delta_h(s_h^{k_i(s_h^k,a_h^k,h)})\right),
\end{aligned} \tag{C.41}$$

where $k_i(s, a, h)$ represents the episode where $(s, a)$ was taken by the agent at step $h$ for the $i$th time.

The key step is to upper bound the third term in the RHS of (C.41). Note that for any episode $k$, the third term takes all the prior episodes $k_i < k$ where $(s_h^k, a_h^k)$ was taken into account. In other words, for any episode $k'$, the term $\delta_{h+1}^{k'}$ appears in the second term at all posterior episodes $k > k'$ where $(s_h^{k'}, a_h^{k'})$ was taken. The first time it appears we have $\overline{N}_h^k(s_h^k, a_h^k) = \overline{N}_h^k(s_h^{k'}, a_h^{k'}) = \overline{N}_h^{k'}(s_h^{k'}, a_h^{k'}) + 1$ and the second time it appears we have $\overline{N}_h^k(s_h^k, a_h^k) = \overline{N}_h^k(s_h^{k'}, a_h^{k'}) = \overline{N}_h^{k'}(s_h^{k'}, a_h^{k'}) + 2$, and so on. Thus, we exchange the order of summation and have

$$\begin{aligned}&\sum_{k=1}^{K} \phi_{h,h}^k \sum_{i=1}^{\overline{N}_h^k(s_h^k,a_h^k)} \alpha^i_{\overline{N}_h^k(s_h^k,a_h^k)} \delta_{h+1}^{k_i(s_h^k,a_h^k,h)} \\ =&\sum_{k'=1}^{K} \delta_{h+1}^{k'} \sum_{t=\overline{N}_h^{k'}(s_h^{k'},a_h^{k'})+1}^{\overline{N}_h^K(s_h^{k'},a_h^{k'})} \phi_{h,h}^{k_t(s_h^{k'},a_h^{k'},h)} \alpha_t^{\overline{N}_h^{k'}(s_h^{k'},a_h^{k'})+1}.\end{aligned} \tag{C.42}$$

For any $k \in [K]$, let $\phi_{h,h+1}^k = \sum_{t=\overline{N}_h^k(s_h^k,a_h^k)+1}^{\overline{N}_h^K(s_h^k,a_h^k)} \phi_{h,h}^{k_t(s_h^k,a_h^k)} \alpha_t^{\overline{N}_h^k(s_h^k,a_h^k)+1}$. The third term in the RHS of (C.41) can be simplified as $\sum_{k=1}^K \phi_{h,h+1}^k \delta_{h+1}^k$. The fourth term in the RHS of (C.41) can be simplified as

$$\sum_{k=1}^{K} \phi_{h,h+1}^k \mathbb{1}\left(\widetilde{a}_h^k \neq \pi_h^\dagger(s_h^k)\right) \left(2B_h\left(N_h^k(s_h^k, \widetilde{a}_h^k)\right) - \Delta_h(s_h^k)\right). \tag{C.43}$$

Since $\alpha_t^0 = 0$ when $t \geq 1$, $\sum_{k=1}^K \phi_{h,h}^k \alpha^0_{\overline{N}_h^k(s_h^k,a_h^k)} H \leq SAH$. Thus, we can rewrite (C.41) as

$$\begin{aligned}\sum_{k=1}^{K} \phi_{h,h}^k \delta_h^k \leq& SAH + \sum_{k=1}^{K} \phi_{h,h+1}^k \delta_{h+1}^k + \sum_{k=1}^{K} \phi_{h,h}^k \beta_h\left(\overline{N}_h^k(s_h^k, a_h^k)\right) \\ &+ \sum_{k=1}^{K} \phi_{h,h+1}^k \mathbb{1}\left(\widetilde{a}_h^k \neq \pi_h^\dagger(s_h^k)\right) \left(2B_h\left(N_h^k(s_h^k, \widetilde{a}_h^k)\right) - \Delta_h(s_h^k)\right).\end{aligned} \tag{C.44}$$

Recursing the result for $h' = h, h+1, \ldots, H$, and using the fact $\delta_{H+1}^k = 0$ for all episode $k$, we have

$$\begin{aligned}\sum_{k=1}^{K}\phi_{h,h}^{k}\delta_{h}^{k}\leq & SAH(H-h+1)+\sum_{h'=h}^{H}\sum_{k=1}^{K}\phi_{h,h'}^{k}\beta_{h'}\left(\overline{N}_{h'}^{k}(s_{h'}^{k},a_{h'}^{k})\right)\\ &+\sum_{h'=h}^{H}\sum_{k=1}^{K}\phi_{h,h'+1}^{k}\mathbb{1}\left(\widetilde{a}_{h'}^{k}\neq\pi_{h'}^{\dagger}(s_{h}^{k})\right)2B_{h'}\left(N_{h'}^{k}(s_{h'}^{k},\widetilde{a}_{h'}^{k})\right)\\ &-\sum_{h'=h}^{H}\sum_{k=1}^{K}\phi_{h,h'+1}^{k}\mathbb{1}\left(\widetilde{a}_{h'}^{k}\neq\pi_{h'}^{\dagger}(s_{h}^{k})\right)\Delta_{h}(s_{h'}^{k}).\end{aligned}\tag{C.45}$$

Here, we present some important properties of $\phi_{h,h'}^{k}$ for all step $h'\geq h$ when step $h$ are fixed:

(1) $\sum_{k=1}^{K}\phi_{h,h}^{k}=\sum_{k=1}^{K}\mathbb{1}\left(a_{h}^{k}\neq\pi^{\dagger}(s)\right)=\text{Loss}_{h}(K)$;

(2)$\sum_{k=1}^{K}\phi_{h,h'}^{k}=\sum_{k=1}^{K}\phi_{h,h}^{k}$, for all step $h'\geq h$;

(3)$\max_{k\in[K]}\phi_{h,h'+1}^{k}\leq(1+\frac{1}{H})\max_{k\in[K]}\phi_{h,h'}^{k}$ for all step $h'\geq h$;

(4)$\max_{k\in[K]}\phi_{h,h}^{k}=1$, and $\max_{k\in[K]}\phi_{h,h'}^{k}\leq e$ for all step $h'\geq h$.

Property (1) is from the definition of $\overline{N}_{h}^{k}(s)$. Properties (2) and (3) can be proved by the properties of $\alpha_{t}^{i}$. In particular, for all step $h'\geq h$,

$$\sum_{k=1}^{K}\phi_{h,h'+1}^{k}=\sum_{k=1}^{K}\phi_{h,h'}^{k}\sum_{i=1}^{\overline{N}_{h'}^{k}(s_{h'}^{k},a_{h'}^{k})}\alpha^{i}_{\overline{N}_{h'}^{k}(s_{h'}^{k},a_{h'}^{k})}=\sum_{k=1}^{K}\phi_{h,h'}^{k},\tag{C.46}$$

and for all step $h'\geq h$ and all episode $k\in[K]$,

$$\begin{aligned}\phi_{h,h'+1}^{k}= & \sum_{t=\overline{N}_{h'}^{k}(s_{h'}^{k},a_{h'}^{k})+1}^{\overline{N}_{h'}^{K}(s_{h'}^{k},a_{h'}^{k})}\phi_{h,h'}^{k_{t}(s_{h'}^{k},a_{h'}^{k},h')}\alpha_{t}^{\overline{N}_{h'}^{k}(s_{h'}^{k},a_{h'}^{k})+1}\\ \leq & \sum_{t=\overline{N}_{h'}^{k}(s_{h'}^{k},a_{h'}^{k})+1}^{\overline{N}_{h'}^{K}(s_{h'}^{k},a_{h'}^{k})}\alpha_{t}^{\overline{N}_{h'}^{k}(s_{h'}^{k},a_{h'}^{k})+1}\max_{k\in[K]}\phi_{h,h'}^{k}\\ \leq & (1+\frac{1}{H})\max_{k\in[K]}\phi_{h,h'}^{k}.\end{aligned}\tag{C.47}$$

Property (4) is from Property (3) and the fact $(1+\frac{1}{H})^{H}\leq e$.

Now we are ready to prove Theorem 10. At first, we bound the second term of the RHS of (C.45). We regroup the summands in a different way.

$$\begin{aligned}\sum_{h'=h}^{H}\sum_{k=1}^{K}\phi_{h,h'}^{k}\cdot\beta_{h'}\left(\overline{N}_{h'}^{k}(s_{h'}^{k},a_{h'}^{k})\right)=&\sum_{h'=h}^{H}\sum_{(s,a)\in\mathcal{S}\times\mathcal{A}}\sum_{t=1}^{\overline{N}_{h'}^{K}(s,a)}\phi_{h,h'}^{k_t(s,a,h')}\beta_{h'}(t-1)\\=&\sum_{h'=h}^{H}\sum_{(s,a)\in\mathcal{S}\times\mathcal{A}}\sum_{t=2}^{\overline{N}_{h'}^{K}(s,a)}\phi_{h,h'}^{k_t(s,a,h')}\beta_{h'}(t-1),\end{aligned}\tag{C.48}$$

because $\beta_{h'}(0)=0$. Define $\phi_{h,h'}^{(s,a)}=\sum_{t=1}^{\overline{N}_{h'}^{K}(s,a)}\phi_{h,h'}^{k_t(s,a,h)}$. Since $\sqrt{\frac{1}{t}}$ is a monotonically decreasing positive function for $n\geq 1$ and $\phi_{h,h'}^{k_t(s,a,h')}\leq e$, by the rearrangement inequality, for $h'\geq h$, we have

$$\begin{aligned}\sum_{t=1}^{\overline{N}_{h'}^{K}(s,a)}\phi_{h,h'}^{k_t(s,a,h)}\sqrt{\frac{1}{t}}\leq&\sum_{t=1}^{\lfloor\phi_{h,h'}^{(s,a)}/e\rfloor}e\sqrt{\frac{1}{t}}+(\phi_{h,h'}^{(s,a)}-\lfloor\phi_{h,h'}^{(s,a)}/e\rfloor)\sqrt{\frac{1}{\lceil\phi_{h,h'}^{(s,a)}/e\rceil}}\\\leq& e\sqrt{\frac{1}{1}}+\int_{1}^{\phi_{h,h'}^{(s,a)}/e}e\sqrt{\frac{1}{t}}dt\leq 2\sqrt{e\phi_{h,h'}^{(s,a)}}.\end{aligned}\tag{C.49}$$

By plugging (C.49) back into (C.48) we have

$$\begin{aligned}\sum_{h'=h}^{H}\sum_{k=1}^{K}\phi_{h,h'}^{k}\cdot\beta_{h'}\left(\overline{N}_{h'}^{k}(s_{h'}^{k},a_{h'}^{k})\right)\leq&\sum_{h'=h}^{H}2(cH+2(H-h')+2)\sqrt{eSAH\iota\sum_{k=1}^{K}\phi_{h,h}^{k}}\\\leq&H(cH+2H+2)\sqrt{eSAH\iota\sum_{k=1}^{K}\phi_{h,h}^{k}}\\=&H(cH+2H+2)\sqrt{eSAH\iota\mathrm{Loss}_h(K)},\end{aligned}\tag{C.50}$$

where the first inequality holds due to $\sum_{(s,a)\in\mathcal{S}\times\mathcal{A}}\phi_{h,h'}^{(s,a)}=\sum_{k=1}^{K}\phi_{h,h'}^{k}$ and $\sqrt{t}$ is a concave function for $t\geq 0$.

Similarly, we can bound a part of the third term of the RHS of (C.45) by

$$\begin{aligned}
&\sum_{h'=h}^{H}\sum_{k=1}^{K}\phi_{h,h'+1}^{k}\mathbb{1}\left(\widetilde{a}_{h'}^{k}\neq\pi_{h'}^{\dagger}(s_{h}^{k})\right)2B_{h'}\left(N_{h'}^{k}(s_{h'}^{k},\widetilde{a}_{h'}^{k})\right)\\
\overset{①}{\leq}&\sum_{h'=h}^{H}\sum_{(s,\widetilde{a})\in\mathcal{S}\times\mathcal{A}}\sum_{t=1}^{N_{h'}^{K}(s,\widetilde{a})}\phi_{h,h'+1}^{k_t(s,\widetilde{a},h')}2B_{h'}(t-1)\\
\overset{②}{\leq}&\sum_{h'=h}^{H}\sum_{(s,\widetilde{a})\in\mathcal{S}\times\mathcal{A}}\sum_{t=2}^{N_{h'}^{K}(s,\widetilde{a})}\phi_{h,h'+1}^{k_t(s,\widetilde{a},h')}2B_{h'}(t-1)+2e(H-h+1)SAH\\
\overset{③}{\leq}&H^2e\sqrt{SA\iota\sum_{k=1}^{K}\phi_{h,h}^{k}}+2e(H-h+1)SAH\\
=&H^2e\sqrt{SA\iota\text{Loss}_h(K)}+2e(H-h+1)SAH
\end{aligned}\tag{C.51}$$

where $k_t(s,\widetilde{a},h)$ represents the episode where $(s,\widetilde{a})$ was taken by the attacker at step $h$ for the $i$th time. Here, ① comes from deleting the indicator function and regrouping the summands; ② follows $\phi_{h,h'}^{k}\leq e$ and $B_h(0)=H$; ③ follows the same steps in (C.49) and (C.50).

As shown in (C.38), we have

$$0\leq\sum_{k=1}^{K}\phi_{h,h}^{k}\left(\overline{Q}_{h}^{k}(s_{h}^{k},a_{h}^{k})-Q_{h}^{\dagger}\left(s_{h}^{k},\pi_{h}^{-}(s_{h}^{k})\right)\right)-\sum_{k=1}^{K}\phi_{h,h}^{k}\Delta_h(s_{h}^{k})\leq\sum_{k=1}^{K}\phi_{h,h}^{k}\delta_{h}^{k}.\tag{C.52}$$

Thus, we need to find the lower bound of the fourth term of the RHS of (C.45). Since $\Delta_h(s_h^k)>\Delta_{min}>0$, we have

$$\begin{aligned}
&\sum_{h'=h}^{H}\sum_{k=1}^{K}\phi_{h,h'+1}^{k}\mathbb{1}\left(\widetilde{a}_{h'}^{k}\neq\pi_{h'}^{\dagger}(s_{h}^{k})\right)\Delta_h(s_{h'}^{k})\\
\geq&\sum_{k=1}^{K}\phi_{h,h+1}^{k}\mathbb{1}\left(\widetilde{a}_{h}^{k}\neq\pi_{h}^{\dagger}(s_{h}^{k})\right)\Delta_h(s_{h}^{k})\\
\geq&\Delta_{min}\sum_{k=1}^{K}\phi_{h,h+1}^{k}\mathbb{1}\left(\widetilde{a}_{h}^{k}\neq\pi_{h}^{\dagger}(s_{h}^{k})\right)\\
=&\Delta_{min}\left(\text{Loss}_h(K)-\sum_{k=1}^{K}\phi_{h,h+1}^{k}\mathbb{1}\left(\widetilde{a}_{h}^{k}=\pi_{h}^{\dagger}(s_{h}^{k})\right)\right).
\end{aligned}\tag{C.53}$$

Recall the definition of $\phi_{h,h+1}^{k}$ and the property (5) of $\alpha_t^i$, we have

$$\begin{aligned}
&\sum_{k=1}^{K}\phi_{h,h+1}^{k}\mathbb{1}\left(\widetilde{a}_h^k=\pi_h^{\dagger}(s_h^k)\right)\\
=&\sum_{k=1}^{K}\sum_{t=\overline{N}_h^k(s_h^k,a_h^k)+1}^{\overline{N}_h^K(s_h^k,a_h^k)}\phi_{h,h}^{k_t(s_h^k,a_h^k,h)}\alpha_t^{\overline{N}_h^k(s_h^k,a_h^k)+1}\mathbb{1}\left(\widetilde{a}_h^k=\pi_h^{\dagger}(s_h^k)\right)\\
=&\sum_{s\in\mathcal{S}}\sum_{k=1}^{K}\mathbb{1}\left(s_h^k=s\right)\mathbb{1}\left(\widetilde{a}_h^k=\pi_h^{\dagger}(s)\right)\mathbb{1}\left(a_h^k\neq\pi_h^{\dagger}(s)\right)\sum_{t=\overline{N}_h^k(s,a_h^k)+1}^{\overline{N}_h^K(s,a_h^k)}\alpha_t^{\overline{N}_h^k(s,a_h^k)+1}\\
\leq&(1+\frac{1}{H})\sum_{k=1}^{K}\mathbb{1}\left(\widetilde{a}_h^k=\pi_h^{\dagger}(s_h^k)\right)\mathbb{1}\left(a_h^k\neq\pi_h^{\dagger}(s_h^k)\right).
\end{aligned}\tag{C.54}$$

Recall the inequality (C.18). We have with probability $1-p$, for all $h\in[H]$

$$\begin{aligned}
&\sum_{k=1}^{K}\mathbb{1}\left(a_h^k\neq\pi_h^{\dagger}(s_h^k)\right)\mathbb{1}\left(\widetilde{a}_h^k\neq\pi_h^{\dagger}(s)\right)\\
\geq&\frac{1}{H}\sum_{k=1}^{K}\mathbb{1}\left(a_h^k\neq\pi_h^{\dagger}(s_h^k)\right)-\sqrt{2\log(2H/p)\sum_{k=1}^{K}\mathbb{1}\left(a_h^k\neq\pi_h^{\dagger}(s_h^k)\right)},
\end{aligned}\tag{C.55}$$

which is equivalent to

$$\begin{aligned}
&\sum_{k=1}^{K}\mathbb{1}\left(a_h^k\neq\pi_h^{\dagger}(s_h^k)\right)\mathbb{1}\left(\widetilde{a}_h^k=\pi_h^{\dagger}(s)\right)\\
\leq&(1-\frac{1}{H})\sum_{k=1}^{K}\mathbb{1}\left(a_h^k\neq\pi_h^{\dagger}(s_h^k)\right)+\sqrt{2\log(2H/p)\sum_{k=1}^{K}\mathbb{1}\left(a_h^k\neq\pi_h^{\dagger}(s_h^k)\right)},
\end{aligned}\tag{C.56}$$

Plugging these back into (C.54) and further (C.53), we have

$$\begin{aligned}
&\sum_{h'=h}^{H}\sum_{k=1}^{K}\phi_{h,h'+1}^{k}\mathbb{1}\left(\widetilde{a}_{h'}^k\neq\pi_{h'}^{\dagger}(s_h^k)\right)\Delta_h(s_{h'}^k)\\
&\geq\Delta_{min}\left(\frac{1}{H^2}\text{Loss}_h(K)-(1+\frac{1}{H})\sqrt{2\log(2H/p)\sum_{k=1}^{K}\text{Loss}_h(K)}\right).
\end{aligned}\tag{C.57}$$

Combining (C.45), (C.50), (C.51) and (C.57), we have

$$\begin{aligned}
&\Delta_{min}\left(\frac{1}{H^2}\text{Loss}_h(K) - (1+\frac{1}{H})\sqrt{2log(2H/p)\sum_{k=1}^{K}\text{Loss}_h(K)}\right) \\
\leq &H^2 e\sqrt{SA\iota\text{Loss}_h(K)} + 2e(H-h+1)SAH \\
&+ SAH(H-h+1) + H(cH+2H+2)\sqrt{eSAH\iota\text{Loss}_h(K)},
\end{aligned} \tag{C.58}$$

which is equivalent to

$$\begin{aligned}
\text{Loss}_h(K) \leq &2(H^2+H)^2\log(2H/p) + \frac{1}{\Delta_{min}}SAH^2(H-h+1) \\
&+ \frac{1}{\Delta_{min}^2}e^2H^8SA\iota + \frac{1}{\Delta_{min}^2}eH^7(cH+2H+2)^2SA\iota.
\end{aligned} \tag{C.59}$$

This establishes

$$\text{Cost}(K,H) \leq \text{Loss}(K) \leq O\left(H^5\log(2H/p) + \frac{1}{\Delta_{min}}SAH^4 + \frac{1}{\Delta_{min}^2}H^{10}SA\iota\right). \tag{C.60}$$

# Appendix D

# Appendix of Chapter 5

## D.1 Notations

In this section, we introduce some notations that will be frequently used in appendixes.

The attack strategies in this paper are all Markov and only depend on the current state and actions. The post-attack reward function has the same form as the original reward function which is Markov and bounded in $[0, 1]$. Thus, the combination of the attacker and the environment $MG(\mathcal{S}, \{\mathcal{A}_i\}_{i=1}^m, H, P, \{R_i\}_{i=1}^m)$ can also be considered as a new environment $\widetilde{MG}(\mathcal{S}, \{\mathcal{A}_i\}_{i=1}^m, H, \widetilde{P}, \{\widetilde{R}_i\}_{i=1}^m)$, and the agents interact with the new environment. $\widetilde{R}_{i,h} : \mathcal{S} \times \mathcal{A} \to [0, 1]$ represents the post-attack reward function for the $i^{th}$ agent in the step $h$. The post-attack transition probabilities satisfy $\widetilde{P}_h(s'|s, \boldsymbol{a}) = \sum_{\boldsymbol{a}'} \mathbb{A}_h(\boldsymbol{a}'|s, \boldsymbol{a}) P_h(s'|s, \boldsymbol{a}')$.

We use $\widetilde{R}_i$, $\widetilde{N}_i$, $\widetilde{Q}_i$ and $\widetilde{V}_i$ to denote the mean rewards, counter, $Q$-values and value functions of the new post-attack environment that each agent $i$ observes. We use $N^k$, $V^k$ and $\pi^k$ to denote the counter, value functions, and policy maintained by the agents' algorithm at the beginning of the episode $k$.

For notation simplicity, we define two operators $\mathbb{P}$ and $\mathbb{D}$ as follows:

$$\begin{aligned} \mathbb{P}_h[V](s, \boldsymbol{a}) &= \mathbb{E}_{s' \sim P_h(\cdot|s,\boldsymbol{a})}\left[V(s')\right], \\ \mathbb{D}_\pi[Q](s) &= \mathbb{E}_{\boldsymbol{a}\sim\pi(\cdot|s)}\left[Q(s,\boldsymbol{a})\right]. \end{aligned} \tag{D.1}$$

Furthermore, we let $\mathbb{A}$ denote the action manipulation. $\mathbb{A} = \{\mathbb{A}_h\}_{h\in[H]}$ is a collection of action-manipulation matrices, so that $\mathbb{A}_h(\cdot|s,\boldsymbol{a})$ gives the probability distribution of the post-attack action if actions $\boldsymbol{a}$ are taken at state $s$ and step $h$. Using this notation, in the $d$-portion attack strategy, we have $\mathbb{A}_h(\pi_h^\dagger(s)|s,\boldsymbol{a}) = d_h(s,\boldsymbol{a})/m$, and $\mathbb{A}_h(\pi_h^-(s)|s,\boldsymbol{a}) = 1 - d_h(s,\boldsymbol{a})/m$.

## D.2 Proof of the insufficiency of action poisoning only attacks and reward poisoning only attacks

### D.2.1 Proof of Theorem 11

We consider a simple case of Markov game where $m = 2$, $H = 1$ and $|\mathcal{S}| = 1$. The reward function can be expressed in the matrix form in Table D.1.

Table D.1: Reward matrix

| | Cooperate | Defect |
|---|---|---|
| Cooperate | (1, 1) | (0.5, 0.5) |
| Defect | (0.5, 0.5) | (0.1, 0.1) |

The target policy is that the two agents both choose to defect. In this MG, the two agents' rewards are the same under any action. As the action attacks only change the agent's action, the post-attack rewards have the same property. The post-attack reward function can be expressed in the matrix form in Table D.2.

To achieve the objective in (5.1), we first have $r_2 \leq r_4$ and $r_3 \leq r_4$, as the target policy should be an NE. Since the other distinct policy should not be an $\epsilon$-approximate CCE, we consider the other three pure-strategy policies and have

Table D.2: Post-attack reward matrix

| | Cooperate | Defect |
|---|---|---|
| Cooperate | $(r_1, r_1)$ | $(r_2, r_2)$ |
| Defect | $(r_3, r_3)$ | $(r_4, r_4)$ |

$$\begin{cases} r_1 > r_2 + \epsilon, \text{or } r_4 > r_2 + \epsilon \\ r_1 > r_3 + \epsilon, \text{or } r_4 > r_3 + \epsilon \cdot \\ r_3 > r_1 + \epsilon, \text{or } r_2 > r_1 + \epsilon \end{cases} \tag{D.2}$$

Note that $r_3 > r_1 + \epsilon$ and $r_1 > r_3 + \epsilon$ are contradictory and $r_2 > r_1 + \epsilon$ and $r_1 > r_2 + \epsilon$ are contradictory. We must have $r_4 > r_3 + \epsilon$ or $r_4 > r_2 + \epsilon$. As the action attacks will keep the same boundary of the rewards, $r_3 \geq 0.1$ and $r_2 \geq 0.1$. Then, $r_4 > 0.1 + \epsilon$.

Suppose there exists an action poisoning attack strategy that can successfully attack MARL agents. We have $\sum_{k=1}^{K} \sum_{h=1}^{H} \sum_{i=1}^{m} \mathbb{1}\left(a_{i,h}^k = \pi^\dagger(s_{i,h}^k)\right) = T - o(T) = \Omega(T)$, i.e. the attack loss scales on $o(T)$. To achieve the post-attack reward satisfy $r_4 > 0.1 + \epsilon$, the attacker needs to change the target action (Defect, Defect) to other actions with probability at least $\epsilon$, when the agents choose the target action. Then, we have $\sum_{k=1}^{K} \sum_{h=1}^{H} \sum_{i=1}^{m} \mathbb{E}(\mathbb{1}(\widetilde{a}_{i,h}^k \neq a_{i,h}^k)) = \Omega(\epsilon T)$. The expected attack cost is linearly dependent on $T$. Hence, there does not exist an action poisoning attack strategy that is both efficient and successful for this case.

### D.2.2 Proof of Theorem 12

We consider a simple case of Markov game where $m = 2$, $H = 2$ and $|\mathcal{S}| = 3$. The reward functions are expressed in the following Table D.3.

The initial state is $s_1$ at $h = 1$ and the transition probabilities are:

$$\begin{aligned} &P(s_2|s_1, a) = 0.9, P(s_3|s_1, a) = 0.1, \text{ if } a = \text{(Defect, Defect)}, \\ &P(s_2|s_1, a) = 0.1, P(s_3|s_1, a) = 0.9, \text{ if } a \neq \text{(Defect, Defect)}. \end{aligned} \tag{D.3}$$

The target policy is that the two agents both choose to defect at any state. The post-attack

Table D.3: Reward matrix

| state $s_1$ | Cooperate | Defect |
|---|---|---|
| Cooperate | (1, 1) | (0.5, 0.5) |
| Defect | (0.5, 0.5) | (0.2, 0.2) |
| state $s_2$ | Cooperate | Defect |
| Cooperate | (1, 1) | (0.5, 0.5) |
| Defect | (0.5, 0.5) | (0.1, 0.1) |
| state $s_3$ | Cooperate | Defect |
| Cooperate | (1, 1) | (0.5, 0.5) |
| Defect | (0.5, 0.5) | (0.9, 0.9) |

reward function of the three states can be expressed in the matrix form in Table D.4.

Table D.4: Post-attack reward matrix

| state $s_1$ | Cooperate | Defect |
|---|---|---|
| Cooperate | $(r_1, r_2)$ | $(r_3, r_4)$ |
| Defect | $(r_5, r_6)$ | $(r_7, r_8)$ |
| state $s_2$ | Cooperate | Defect |
| Cooperate | ( - , - ) | ( - , - ) |
| Defect | ( - , - ) | $(r_9, r_{10})$ |
| state $s_3$ | Cooperate | Defect |
| Cooperate | ( - , - ) | ( - , - ) |
| Defect | ( - , - ) | $(r_{11}, r_{12})$ |

We limit that the post-attack mean reward $\widetilde{R}$ has the same boundary condition with that of the pre-attack mean reward $R$, i.e. $\widetilde{R} \in [0, 1]$. Then, $0 \leq r_1, \ldots, r_{12} \leq 1$.

Suppose there exists a reward poisoning attack strategy that can successfully attack MARL agents, we have $\sum_{k=1}^{K} \sum_{h=1}^{H} \sum_{i=1}^{m} \mathbb{1}\left(a_{i,h}^k = \pi^\dagger(s_{i,h}^k)\right) = T - o(T) = \Omega(T)$, i.e. the attack loss scales on $o(T)$.

If $|r_9 - 0.1| > 0.1$, $|r_{10} - 0.1| > 0.1$, $|r_{11} - 0.9| > 0.1$, or $|r_{12} - 0.9| > 0.1$, we have the attack cost $\sum_{k=1}^{K} \sum_{h=1}^{H} \sum_{i=1}^{m} \mathbb{E}(|\widetilde{r}_{i,h}^k - r_{i,h}^k|) = \Omega(0.1 * K) = \Omega(T)$. Thus, $|r_9 - 0.1| \leq 0.1$, $|r_{10} - 0.1| \leq 0.1$, $|r_{11} - 0.9| \leq 0.1$ and $|r_{12} - 0.9| \leq 0.1$.

For the target policy, we have $\widetilde{V}_{i,1}^{\pi^\dagger}(s_1) = r_7 + 0.9 * r_9 + 0.1 * r_{11}$. For the policy $\pi'$ with $\pi'_1(s_1) =$ (Cooperate, Defect), $\pi'_2(s_2) =$ (Defect, Defect), $\pi'_2(s_3) =$ (Defect, Defect), we have $\widetilde{V}_{i,1}^{\pi'}(s_1) = r_3 + 0.1 * r_9 + 0.9 * r_{11}$.

To achieve the objective in (5.1), the attacker should let the target policy to be an NE. Thus, we have $\widetilde{V}_{i,1}^{\pi^\dagger}(s_1) \geq \widetilde{V}_{i,1}^{\pi'}(s_1)$ and then $r_7 + 0.9 * r_9 + 0.1 r_{11} \geq r_3 + 0.1 * r_9 + 0.9 * r_{11}$. As $|r_9 - 0.1| \leq 0.1$ and $|r_{11} - 0.9| \leq 0.1$, we have $r_7 \geq r_3 + 0.48$. From the boundary condition, we have $r_3 \geq 0$ and then $r_7 \geq 0.48$. The attack cost scales at least on $\Omega(0.28 * T)$ for a successful reward attack strategy.

In summary, there does not exist an reward poisoning attack strategy that is both efficient and successful for this case.

## D.3 Analysis of the $d$-portion Attack

### D.3.1 Proof of Theorem 13

We assume that the minimum gap exists and is positive, i.e. $\Delta_{min} > 0$. This positive gap provides an opportunity for efficient action poisoning attacks.

We assume that the agent does not know the attacker's manipulations and the presence of the attacker. The attacker's manipulations on actions are stationary. We can consider the combination of the attacker and the environment MG$(\mathcal{S}, \{\mathcal{A}_i\}_{i=1}^m, H, P, \{R_i\}_{i=1}^m)$ as a new environment $\widetilde{\text{MG}}(\mathcal{S}, \{\mathcal{A}_i\}_{i=1}^m, H, \widetilde{P}, \{\widetilde{R}_i\}_{i=1}^m)$, and the agents interact with the new environment. We define $\widetilde{Q}_i$ and $\widetilde{V}_i$ as the $Q$-values and value functions of the new environment $\widetilde{\text{MG}}$ that each agent $i$ observes.

We first prove that $\pi^\dagger$ is an NE from every agent's point of view.

Condition 1 implies that $\pi^\dagger$ is not the worst policy from every agent's point of view, and there exists a policy $\pi^-$ that is worse than the target policy from every agent's point of view. Denote $\Delta_{i,h}^{\dagger-}(s) = Q_{i,h}^{\pi^\dagger}(s, \pi_h^\dagger(s)) - Q_{i,h}^{\pi^\dagger}(s, \pi_h^-(s))$. We define the minimum gap $\Delta_{min} = \min_{h\in[H], s\in\mathcal{S}, i\in[m]} = \Delta_{i,h}^{\dagger-}(s)$.

We set $\mathbb{P}_h V_{i,h+1}^{\pi}(s, \boldsymbol{a}) = \mathbb{E}_{s'\sim P_h(\cdot|s,\boldsymbol{a})}\left[V_{i,h+1}^{\pi}(s')\right]$. From $d$-portion attack strategy, we have

$$\widetilde{Q}_{i,h}^{\pi}(s, \boldsymbol{a}) = \widetilde{R}_{i,h}(s, \boldsymbol{a}) + \frac{d_h(s, \boldsymbol{a})}{m}\mathbb{P}_h \widetilde{V}_{i,h+1}^{\pi}(s, \pi_h^\dagger(s)) + \left(1 - \frac{d_h(s, \boldsymbol{a})}{m}\right)\mathbb{P}_h \widetilde{V}_{i,h+1}^{\pi}(s, \pi_h^-(s)), \tag{D.4}$$

and

$$\widetilde{R}_{i,h}(s,\boldsymbol{a}) = \frac{d_h(s,\boldsymbol{a})}{m} R_{i,h}(s,\pi_h^{\dagger}(s)) + \left(1 - \frac{d_h(s,\boldsymbol{a})}{m}\right) R_{i,h}(s,\pi_h^{-}(s)). \tag{D.5}$$

Since the attacker does not attack when the agents follow the target policy, we have $\widetilde{V}_{i,h+1}^{\pi^\dagger}(s) = V_{i,h+1}^{\pi^\dagger}(s)$. Then,

$$\widetilde{Q}_{i,h}^{\pi^\dagger}(s,\boldsymbol{a}) = \frac{d_h(s,\boldsymbol{a})}{m} Q_{i,h}^{\pi^\dagger}(s,\pi_h^{\dagger}(s)) + \left(1 - \frac{d_h(s,\boldsymbol{a})}{m}\right) Q_{i,h}^{\pi^\dagger}(s,\pi_h^{-}(s)). \tag{D.6}$$

If $a_i \neq \pi_{i,h}^{\dagger}(s)$, we have

$$\widetilde{Q}_{i,h}^{\pi^\dagger}(s,\pi_{i,h}^{\dagger}(s)\times \boldsymbol{a}_{-i}) - \widetilde{Q}_{i,h}^{\pi^\dagger}(s,\boldsymbol{a}) = \frac{1}{2m}\left(Q_{i,h}^{\pi^\dagger}(s,\pi_h^{\dagger}(s)) - Q_{i,h}^{\pi^\dagger}(s,\pi_h^{-}(s))\right) \geq \frac{\Delta_{min}}{2m}. \tag{D.7}$$

We have that policy $\pi_i^\dagger$ is best-in-hindsight policy towards the target policy $\pi_{-i}^\dagger$ at step $h$ in the observation of each agent $i$, i.e. $\widetilde{V}_{i,h+1}^{\pi_i^\dagger \times \pi_{-i}^\dagger}(s) = \widetilde{V}_{i,h+1}^{\dagger,\pi_{-i}^\dagger}(s)$ for any agent $i$, any state $s$ and any policy $\pi_{-i}$.

Since the above argument works for any step $h \in [H]$, we have that the best response of each agent $i$ towards the target product policy $\pi_{-i}^\dagger$ is $\pi_i^\dagger$ and the target policy is an {NE, CE, CCE} under $d$-portion attack.

Now we prove that the target policy $\pi_i^\dagger$ is the unique {NE, CE, CCE}, when every state $s \in \mathcal{S}$ is reachable at every step $h \in [H]$ under the target policy.

If there exists an CCE $\pi'$ under $d$-portion attack, we have $\max_{i\in[m]}(\widetilde{V}_{i,1}^{\dagger,\pi'_{-i}}(s) - \widetilde{V}_{i,1}^{\pi'}(s)) = 0$ for any initial state $s$.

At the step $H$, $\widetilde{Q}_{i,H}^{\pi}(s,\boldsymbol{a}) = \widetilde{R}_{i,H}(s,\boldsymbol{a})$. Since $R_{i,H}(s,\pi_H^\dagger(s)) \geq R_{i,H}(s,\pi_H^-(s)) + \Delta_{min}$ with $\Delta_{min} > 0$, the policy $\pi_{i,H}^\dagger$ is the unique best response towards any policy $\pi_{-i,H}$, i.e. $\widetilde{V}_{i,H}^{\pi_{i,H}^\dagger,\pi_{-i,H}}(s) = \widetilde{V}_{i,H}^{\dagger,\pi_{-i,H}}(s)$ and $\widetilde{V}_{i,H}^{\pi_{i,H}^\dagger,\pi_{-i,H}}(s) > \widetilde{V}_{i,H}^{\pi_{i,H},\pi_{-i,H}}(s)$ for any $\pi_{i,H}(\cdot|s) \neq \pi_{i,H}^\dagger(\cdot|s)$. Thus, we have $\pi'_H(s_H) = \pi_H^\dagger(s_H)$ for any state $s_H$ that is reachable at the time step $H$ under policy

$\pi'$. We assume that every state $s \in \mathcal{S}$ is reachable at every step $h \in [H]$ under the target policy. Under $d$-portion attacks, the post-attack action $\widetilde{\boldsymbol{a}}_h = \pi^\dagger_h(s)$ with probability more than $0.5$. Thus, every state $s \in \mathcal{S}$ is reachable at every step $h \in [H]$ under any policy $\pi$ and $d$-portion attacks.

Recall that for any $a_i \neq \pi^\dagger_{i,h}(s)$,

$$\widetilde{Q}^{\pi^\dagger}_{i,h}(s, \pi^\dagger_{i,h}(s) \times \boldsymbol{a}_{-i}) - \widetilde{Q}^{\pi^\dagger}_{i,h}(s, \boldsymbol{a}) \geq \Delta_{min}/2m. \tag{D.8}$$

Suppose $\{\pi'_{h'}\}_{h'=h+1}^H = \{\pi^\dagger_{h'}\}_{h'=h+1}^H$ for any states. If $\pi'_{i,h}(\cdot|s) \neq \pi^\dagger_{i,h}(\cdot|s)$ at a reachable state $s$, we have

$$\begin{aligned}
\widetilde{V}^{\pi'}_{i,h}(s) =& \mathbb{E}_{\boldsymbol{a}\sim\pi'_h(\cdot|s)}[\widetilde{Q}^{\pi'}_{i,h}(s,\boldsymbol{a})] \\
=& \mathbb{E}_{\boldsymbol{a}\sim\pi'_h(\cdot|s)}[\widetilde{Q}^{\pi^\dagger}_{i,h}(s,\boldsymbol{a})] \\
\leq& \mathbb{E}_{\boldsymbol{a}_{-i}\sim\pi'_{-i,h}(\cdot|s)}[\widetilde{Q}^{\pi^\dagger}_{i,h}(s,\pi^\dagger_{i,h}(s)\times \boldsymbol{a}_{-i})] - c\Delta_{min}/2m \\
=& \widetilde{V}^{\pi^\dagger\times\pi'_{-i}}_{i,h}(s) - c\Delta_{min}/m
\end{aligned} \tag{D.9}$$

with some constant $c > 0$. Then, $\pi'$ is not an CCE in such a case.

From induction on $h = H, H-1, \cdots, 1$, $\pi' = \pi^\dagger$ for any states. If every state $s \in \mathcal{S}$ is reachable at every step $h \in [H]$ under the target policy, $\pi^\dagger$ is the unique {NE, CE, CCE}.

### D.3.2 Proof of Theorem 14

Consider an arbitrary Markov policy $\pi$. From $d$-portion attack strategy, we have

$$\begin{aligned}
\widetilde{Q}^{\pi}_{i,h}(s,\boldsymbol{a}) =& \frac{d_h(s,\boldsymbol{a})}{m} R_{i,h}(s,\pi^\dagger_h(s)) + \frac{m - d_h(s,\boldsymbol{a})}{m} R_{i,h}(s,\pi^-_h(s)) \\
&+ \frac{d_h(s,\boldsymbol{a})}{m}\mathbb{P}_h \widetilde{V}^{\pi}_{i,h+1}(s,\pi^\dagger_h(s)) + \frac{m - d_h(s,\boldsymbol{a})}{m}\mathbb{P}_h \widetilde{V}^{\pi}_{i,h+1}(s,\pi^-_h(s)) \\
=& \frac{d_h(s,\boldsymbol{a}) - m}{m}\Big(R_{i,h}(s,\pi^\dagger_h(s)) - R_{i,h}(s,\pi^-_h(s))\Big) \\
&+ \frac{d_h(s,\boldsymbol{a}) - m}{m}\Big(\mathbb{P}_h \widetilde{V}^{\pi}_{i,h+1}(s,\pi^\dagger_h(s)) - \mathbb{P}_h \widetilde{V}^{\pi}_{i,h+1}(s,\pi^-_h(s))\Big) \\
&+ R_{i,h}(s,\pi^\dagger_h(s)) + \mathbb{P}_h \widetilde{V}^{\pi}_{i,h+1}(s,\pi^\dagger_h(s))
\end{aligned} \tag{D.10}$$

and

$$
\begin{aligned}
\widetilde{V}^{\pi}_{i,h}(s) =& \mathbb{D}_{\pi_h}[\widetilde{Q}^{\pi}_{i,h}](s) \\
=& \frac{\mathbb{D}_{\pi_h}[d](s) - m}{m} \left( R_{i,h}(s, \pi^{\dagger}_h(s)) - R_{i,h}(s, \pi^{-}_h(s)) \right) \\
&+ \frac{\mathbb{D}_{\pi_h}[d](s) - m}{m} \left( \mathbb{P}_h \widetilde{V}^{\pi}_{i,h+1}(s, \pi^{\dagger}_h(s)) - \mathbb{P}_h \widetilde{V}^{\pi}_{i,h+1}(s, \pi^{-}_h(s)) \right) \\
&+ R_{i,h}(s, \pi^{\dagger}_h(s)) + \mathbb{P}_h \widetilde{V}^{\pi}_{i,h+1}(s, \pi^{\dagger}_h(s)).
\end{aligned} \tag{D.11}
$$

Now we bound the difference between $\widetilde{V}^{\pi}_{i,h}(s)$ and $\widetilde{V}^{\pi^\dagger}_{i,h}(s)$ for any policy $\pi$.

$$
\widetilde{V}^{\pi^\dagger}_{i,h}(s) - \widetilde{V}^{\pi}_{i,h}(s) = \underbrace{\widetilde{V}^{\pi^\dagger}_{i,h}(s) - \mathbb{D}_{\pi_h}[\widetilde{Q}^{\pi^\dagger}_{i,h}](s)}_{(a)} + \underbrace{\mathbb{D}_{\pi_h}[\widetilde{Q}^{\pi^\dagger}_{i,h}](s) - \widetilde{V}^{\pi}_{i,h}(s)}_{(b)}. \tag{D.12}
$$

For term (a), from equations (D.10) and (D.11), we have

$$
\begin{aligned}
&\widetilde{V}^{\pi^\dagger}_{i,h}(s) - \mathbb{D}_{\pi_h}[\widetilde{Q}^{\pi^\dagger}_{i,h}](s) \\
=& \frac{m - \mathbb{D}_{\pi_h}[d](s)}{m} \left( R_{i,h}(s, \pi^{\dagger}_h(s)) - R_{i,h}(s, \pi^{-}_h(s)) \right) \\
&+ \frac{m - \mathbb{D}_{\pi_h}[d](s)}{m} \left( \mathbb{P}_h \widetilde{V}^{\pi^\dagger}_{i,h+1}(s, \pi^{\dagger}_h(s)) - \mathbb{P}_h \widetilde{V}^{\pi^\dagger}_{i,h+1}(s, \pi^{-}_h(s)) \right).
\end{aligned} \tag{D.13}
$$

Since the attacker does not attack when the agents follow the target policy, we have $\widetilde{V}^{\pi^\dagger}_{i,h+1}(s) = V^{\pi^\dagger}_{i,h+1}(s)$.

$$
\widetilde{V}^{\pi^\dagger}_{i,h}(s) - \mathbb{D}_{\pi_h}[\widetilde{Q}^{\pi^\dagger}_{i,h}](s) = \frac{m - \mathbb{D}_{\pi_h}[d](s)}{m} \left( Q^{\pi^\dagger}_{i,h}(s, \pi^{\dagger}_h(s)) - Q^{\pi^\dagger}_{i,h}(s, \pi^{-}_h(s)) \right). \tag{D.14}
$$

Denote $\Delta^{\dagger -}_{i,h}(s) = Q^{\pi^\dagger}_{i,h}(s, \pi^{\dagger}_h(s)) - Q^{\pi^\dagger}_{i,h}(s, \pi^{-}_h(s))$. We have

$$
\widetilde{V}^{\pi^\dagger}_{i,h}(s) - \mathbb{D}_{\pi_h}[\widetilde{Q}^{\pi^\dagger}_{i,h}](s) = \frac{\Delta^{\dagger -}_{i,h}(s)}{2m} \mathbb{E}_{\boldsymbol{a} \sim \pi_h(\cdot|s)} \left[ \sum_{i=1}^{m} \mathbb{1}(a_i \neq \pi^{\dagger}_{i,h}(s)) \right]. \tag{D.15}
$$

For term (b), from equations (D.10) and (D.11), we have

$$\begin{aligned}
&\mathbb{D}_{\pi_h}[\widetilde{Q}^{\pi^\dagger}_{i,h}](s) - \widetilde{V}^{\pi}_{i,h}(s)\\
=&\frac{\mathbb{D}_{\pi_h}[d](s)}{m}\mathbb{P}_h\widetilde{V}^{\pi^\dagger}_{i,h+1}(s,\pi^\dagger_h(s)) + \frac{m-\mathbb{D}_{\pi_h}[d](s)}{m}\mathbb{P}_h\widetilde{V}^{\pi^\dagger}_{i,h+1}(s,\pi^-_h(s))\\
&-\frac{\mathbb{D}_{\pi_h}[d](s)}{m}\mathbb{P}_h\widetilde{V}^{\pi}_{i,h+1}(s,\pi^\dagger_h(s)) - \frac{m-\mathbb{D}_{\pi_h}[d](s)}{m}\mathbb{P}_h\widetilde{V}^{\pi}_{i,h+1}(s,\pi^-_h(s))\\
=&\frac{\mathbb{D}_{\pi_h}[d](s)}{m}\mathbb{P}_h[\widetilde{V}^{\pi^\dagger}_{i,h+1} - \widetilde{V}^{\pi}_{i,h+1}](s,\pi^\dagger_h(s)) + \frac{m-\mathbb{D}_{\pi_h}[d](s)}{m}\mathbb{P}_h[\widetilde{V}^{\pi^\dagger}_{i,h+1} - \widetilde{V}^{\pi}_{i,h+1}](s,\pi^-_h(s))\\
=&\mathbb{E}_{s'\sim P_h(\cdot|s,\widetilde{\boldsymbol{a}}),\widetilde{\boldsymbol{a}}\sim\mathbb{A}_h(\cdot|s,\boldsymbol{a}),\boldsymbol{a}\sim\pi_h(\cdot|s)}[\widetilde{V}^{\pi^\dagger}_{i,h+1}(s') - \widetilde{V}^{\pi}_{i,h+1}(s')].
\end{aligned} \tag{D.16}$$

By combining terms (a) and (b), we have

$$\begin{aligned}
&\widetilde{V}^{\pi^\dagger}_{i,h}(s_h) - \widetilde{V}^{\pi}_{i,h}(s_h)\\
=&\frac{\Delta^{\dagger-}_{i,h}(s_h)}{2m}\mathbb{E}_{\boldsymbol{a}\sim\pi_h(\cdot|s_h)}\left[\mathbb{1}(a_i \neq \pi^\dagger_{i,h}(s_h))\right]\\
&+\mathbb{E}_{s_{h+1}\sim P_h(\cdot|s_h,\widetilde{\boldsymbol{a}}),\widetilde{\boldsymbol{a}}\sim\mathbb{A}_h(\cdot|s_h,\boldsymbol{a}),\boldsymbol{a}\sim\pi_h(\cdot|s_h)}[\widetilde{V}^{\pi^\dagger}_{i,h+1}(s_{h+1}) - \widetilde{V}^{\pi}_{i,h+1}(s_{h+1})]\\
=&\cdots = \mathbb{E}_{\pi,\mathbb{A},P}\left[\sum_{h'=h}^{H}\sum_{i=1}^{m}\mathbb{1}(a_{i,h'} \neq \pi^\dagger_{i,h'}(s_{h'}))\frac{\Delta^{\dagger-}_{i,h'}(s_{h'})}{2m}\right].
\end{aligned} \tag{D.17}$$

From the definition of the best-in-hindsight regret and (D.17), we have

$$\begin{aligned}
\mathrm{Reg}_i(K,H) =& \max_{\pi'_i}\sum_{k=1}^{K}[\widetilde{V}^{\pi'_i\times\pi^k_{-i}}_{i,1}(s^k_1) - \widetilde{V}^{\pi^k}_{i,1}(s^k_1)]\\
\geq&\sum_{k=1}^{K}[\widetilde{V}^{\pi^\dagger_i\times\pi^k_{-i}}_{i,1}(s^k_1) - \widetilde{V}^{\pi^k}_{i,1}(s^k_1)].
\end{aligned} \tag{D.18}$$

Now, we bound $\sum_{i=1}^{m}[\widetilde{V}^{\pi^\dagger_i\times\pi_{-i}}_{i,1}(s_1) - \widetilde{V}^{\pi}_{i,1}(s_1)]$ for any policy $\pi$. We introduce some special strategy modifications $\{\phi^\dagger_{i,h}\}_{h=1}^{H}$. For any $h' \geq h$, we have $\phi^\dagger_{i,h} \diamond \pi_{i,h'}(s) = \pi^\dagger_{i,h'}(s)$ and for any

$h' < h$, we have $\phi^{\dagger}_{i,h} \diamond \pi_{i,h'}(s) = \pi_{i,h'}(s)$. Thus,

$$\begin{aligned}&\sum_{i=1}^{m}[\widetilde{V}_{i,1}^{\pi_i^{\dagger}\times\pi_{-i}}(s_1) - \widetilde{V}_{i,1}^{\pi}(s_1)]\\ =&\sum_{h=1}^{H}\sum_{i=1}^{m}[\widetilde{V}_{i,1}^{\phi^{\dagger}_{i,h}\diamond\pi_i\times\pi_{-i}}(s_1) - \widetilde{V}_{i,1}^{\phi^{\dagger}_{i,h+1}\diamond\pi_i\times\pi_{-i}}(s_1)].\end{aligned} \tag{D.19}$$

When $h = H$, we have

$$\begin{aligned}&\sum_{i=1}^{m}\left(\widetilde{V}_{i,1}^{\phi^{\dagger}_{i,H}\diamond\pi_i\times\pi_{-i}}(s_1) - \widetilde{V}_{i,1}^{\phi^{\dagger}_{i,H+1}\diamond\pi_i\times\pi_{-i}}(s_1)\right)\\ =&\mathbb{E}_{\pi,\mathbb{A},P}\left[\sum_{i=1}^{m}\left(\widetilde{V}_{i,H}^{\phi^{\dagger}_{i,H}\diamond\pi_i\times\pi_{-i}}(s_H) - \widetilde{V}_{i,H}^{\phi^{\dagger}_{i,H+1}\diamond\pi_i\times\pi_{-i}}(s_H)\right)\right]\\ =&\mathbb{E}_{\pi,\mathbb{A},P}\left[\sum_{i=1}^{m}\left(\widetilde{V}_{i,H}^{\pi_i^{\dagger}\times\pi_{-i}}(s_H) - \widetilde{V}_{i,H}^{\pi}(s_H)\right)\right]\\ =&\mathbb{E}_{\pi,\mathbb{A},P}\left[\sum_{i=1}^{m}\mathbb{1}(a_{i,H}\neq\pi^{\dagger}_{i,H}(s_H))\frac{\Delta^{\dagger -}_{i,H}(s_H)}{2m}\right].\end{aligned} \tag{D.20}$$

For $h < H$, we have

$$\begin{aligned}&\sum_{i=1}^{m}\left(\widetilde{V}_{i,1}^{\phi^{\dagger}_{i,h}\diamond\pi_i\times\pi_{-i}}(s_1) - \widetilde{V}_{i,1}^{\phi^{\dagger}_{i,h+1}\diamond\pi_i\times\pi_{-i}}(s_1)\right)\\ =&\mathbb{E}_{\pi,\mathbb{A},P}\left[\sum_{i=1}^{m}\left(\widetilde{V}_{i,h}^{\phi^{\dagger}_{i,h}\diamond\pi_i\times\pi_{-i}}(s_h) - \widetilde{V}_{i,h}^{\phi^{\dagger}_{i,h+1}\diamond\pi_i\times\pi_{-i}}(s_h)\right)\right]\\ =&\mathbb{E}_{\pi,\mathbb{A},P}\left[\sum_{i=1}^{m}\left(\mathbb{D}_{\phi^{\dagger}_{i,h}\diamond\pi_{i,h}\times\pi_{-i,h}} - \mathbb{D}_{\phi^{\dagger}_{i,h+1}\diamond\pi_{i,h}\times\pi_{-i,h}}\right)\left[\widetilde{Q}_{i,h}^{\phi^{\dagger}_{i,h+1}\diamond\pi_i\times\pi_{-i}}\right](s_h)\right]\\ =&\mathbb{E}_{\pi,\mathbb{A},P}\left[\sum_{i=1}^{m}\frac{1-\pi_{i,h}\left(s_h,\pi^{\dagger}_{i,h}(s_h)\right)}{2m}\left(\mathbb{D}_{\pi^{\dagger}} - \mathbb{D}_{\pi^{-}}\right)\left[R_{i,h} + \mathbb{P}_h\widetilde{V}_{i,h+1}^{\phi^{\dagger}_{i,h+1}\diamond\pi_i\times\pi_{-i}}\right](s_h)\right]\end{aligned} \tag{D.21}$$

where the second equation holds as $\phi^{\dagger}_{i,h}\diamond\pi_i\times\pi_{-i} = \phi^{\dagger}_{i,h+1}\diamond\pi_i\times\pi_{-i}$ at any time step $h' > h$ and the last equation holds from equation (D.10).

Note that $Q_{i,h}^{\pi^\dagger} = R_{i,h} + \mathbb{P}_h V_{i,h+1}^{\pi^\dagger} = R_{i,h} + \mathbb{P}_h \widetilde{V}_{i,h+1}^{\pi^\dagger}$. From equation (D.21), we have

$$\begin{aligned}
&\sum_{i=1}^{m}\left(\widetilde{V}_{i,1}^{\phi_{i,h}^\dagger \diamond \pi_i \times \pi_{-i}}(s_1) - \widetilde{V}_{i,1}^{\phi_{i,h+1}^\dagger \diamond \pi_i \times \pi_{-i}}(s_1)\right) \\
=&\underbrace{\mathbb{E}_{\pi,\mathbb{A},P}\left[\sum_{i=1}^{m}\frac{1-\pi_{i,h}\left(s_h,\pi_{i,h}^\dagger(s_h)\right)}{2m}\left(\mathbb{D}_{\pi^\dagger}-\mathbb{D}_{\pi^-}\right)\left[Q_{i,h}^{\pi^\dagger}\right](s_h)\right]}_{\textcircled{1}} \\
&+\underbrace{\mathbb{E}_{\pi,\mathbb{A},P}\left[\sum_{i=1}^{m}\frac{1-\pi_{i,h}\left(s_h,\pi_{i,h}^\dagger(s_h)\right)}{2m}\left(\mathbb{D}_{\pi^\dagger}-\mathbb{D}_{\pi^-}\right)\left[\mathbb{P}_h\widetilde{V}_{i,h+1}^{\phi_{i,h+1}^\dagger \diamond \pi_i \times \pi_{-i}} - \mathbb{P}_h\widetilde{V}_{i,h+1}^{\pi^\dagger}\right](s_h)\right]}_{\textcircled{2}}.
\end{aligned} \tag{D.22}$$

Denote $\Delta_{i,h}^{\dagger -}(s) = Q_{i,h}^{\pi^\dagger}(s,\pi_h^\dagger(s)) - Q_{i,h}^{\pi^\dagger}(s,\pi_h^-(s))$. Thus,

$$\textcircled{1} = \mathbb{E}_{\pi,\mathbb{A},P}\left[\sum_{i=1}^{m}\frac{1-\pi_{i,h}\left(s_h,\pi_{i,h}^\dagger(s_h)\right)}{2m}\Delta_{i,h}^{\dagger -}(s_h)\right]. \tag{D.23}$$

Now, we bound item ②. If $\left(\mathbb{D}_{\pi^\dagger}-\mathbb{D}_{\pi^-}\right)\left[\mathbb{P}_h\widetilde{V}_{i,h+1}^{\phi_{i,h+1}^\dagger \diamond \pi_i \times \pi_{-i}} - \mathbb{P}_h\widetilde{V}_{i,h+1}^{\pi^\dagger}\right](s_h) \geq 0$,

$$\begin{aligned}
&\left(\mathbb{D}_{\pi^\dagger}-\mathbb{D}_{\pi^-}\right)\left[\mathbb{P}_h\widetilde{V}_{i,h+1}^{\phi_{i,h+1}^\dagger \diamond \pi_i \times \pi_{-i}} - \mathbb{P}_h\widetilde{V}_{i,h+1}^{\pi^\dagger}\right](s_h) \\
\geq&\frac{2\mathbb{D}_{\pi_h}[d](s_h)}{m}\mathbb{D}_{\pi^\dagger}\mathbb{P}_h[\widetilde{V}_{i,h+1}^{\phi_{i,h+1}^\dagger \diamond \pi_i \times \pi_{-i}} - \widetilde{V}_{i,h+1}^{\pi^\dagger}](s_h) \\
&+\frac{2(m-\mathbb{D}_{\pi_h}[d](s_h))}{m}\mathbb{D}_{\pi^-}\mathbb{P}_h[\widetilde{V}_{i,h+1}^{\phi_{i,h+1}^\dagger \diamond \pi_i \times \pi_{-i}} - \widetilde{V}_{i,h+1}^{\pi^\dagger}](s_h) \\
=&2\mathbb{E}_{\pi,\mathbb{A},P}\left[\widetilde{V}_{i,h+1}^{\phi_{i,h+1}^\dagger \diamond \pi_i \times \pi_{-i}}(s_{h+1}) - \widetilde{V}_{i,h+1}^{\pi^\dagger}(s_{h+1})\right],
\end{aligned} \tag{D.24}$$

because the RHS of the inequality is smaller or equal to $0$.

If $(\mathbb{D}_{\pi^\dagger} - \mathbb{D}_{\pi^-})\left[\mathbb{P}_h\widetilde{V}_{i,h+1}^{\phi^\dagger_{i,h+1}\diamond\pi_i\times\pi_{-i}} - \mathbb{P}_h\widetilde{V}_{i,h+1}^{\pi^\dagger}\right](s_h) \leq 0$,

$$\begin{aligned}
&(\mathbb{D}_{\pi^\dagger} - \mathbb{D}_{\pi^-})\left[\mathbb{P}_h\widetilde{V}_{i,h+1}^{\phi^\dagger_{i,h+1}\diamond\pi_i\times\pi_{-i}} - \mathbb{P}_h\widetilde{V}_{i,h+1}^{\pi^\dagger}\right](s_h)\\
\geq&\frac{2\mathbb{D}_{\pi_h}[d](s_h)}{m}(\mathbb{D}_{\pi^\dagger} - \mathbb{D}_{\pi^-})\left[\mathbb{P}_h\widetilde{V}_{i,h+1}^{\phi^\dagger_{i,h+1}\diamond\pi_i\times\pi_{-i}} - \mathbb{P}_h\widetilde{V}_{i,h+1}^{\pi^\dagger}\right](s_h)\\
\geq&\frac{2\mathbb{D}_{\pi_h}[d](s_h)}{m}\mathbb{D}_{\pi^\dagger}\mathbb{P}_h[\widetilde{V}_{i,h+1}^{\phi^\dagger_{i,h+1}\diamond\pi_i\times\pi_{-i}} - \widetilde{V}_{i,h+1}^{\pi^\dagger}](s_h)\\
&+\frac{2(m-\mathbb{D}_{\pi_h}[d](s_h))}{m}\mathbb{D}_{\pi^-}\mathbb{P}_h[\widetilde{V}_{i,h+1}^{\phi^\dagger_{i,h+1}\diamond\pi_i\times\pi_{-i}} - \widetilde{V}_{i,h+1}^{\pi^\dagger}](s_h)\\
=&2\mathbb{E}_{\pi,\mathbb{A},P}\left[\widetilde{V}_{i,h+1}^{\phi^\dagger_{i,h+1}\diamond\pi_i\times\pi_{-i}}(s_{h+1}) - \widetilde{V}_{i,h+1}^{\pi^\dagger}(s_{h+1})\right].
\end{aligned} \tag{D.25}$$

From (D.17), we have

$$\begin{aligned}
&\widetilde{V}_{i,h+1}^{\pi^\dagger}(s_{h+1}) - \widetilde{V}_{i,h+1}^{\phi^\dagger_{i,h+1}\diamond\pi_i\times\pi_{-i}}(s_{h+1})\\
=&\mathbb{E}_{\phi^\dagger_{i,h+1}\diamond\pi_i\times\pi_{-i},\mathbb{A},P}\left[\sum_{h'=h+1}^{H}\sum_{i=1}^{m}\mathbb{1}(a_{i,h'}\neq\pi^\dagger_{i,h'}(s_{h'}))\frac{\Delta^{\dagger-}_{i,h'}(s_{h'})}{2m}\right]\\
\leq&\sum_{h'=h+1}^{H}(m-1)\max_{s\in\mathcal{S},i\in[m]}\frac{\Delta^{\dagger-}_{i,h'}(s)}{2m}\\
\leq&\frac{(m-1)}{2m}\Delta^{\dagger-}_{i,h}(s_h),
\end{aligned} \tag{D.26}$$

where the last inequality holds when $\min_{s\in\mathcal{S},i\in[m]}\Delta^{\dagger-}_{i,h}(s) \geq \sum_{h'=h+1}^{H}\max_{s\in\mathcal{S},i\in[m]}\Delta^{\dagger-}_{i,h'}(s)$. Combine the above inequalities, we have

$$② \geq -\mathbb{E}_{\pi,\mathbb{A},P}\left[\sum_{i=1}^{m}\frac{1-\pi_{i,h}\left(s_h,\pi^\dagger_{i,h}(s_h)\right)}{2m}\frac{(m-1)}{m}\Delta^{\dagger-}_{i,h}(s_h)\right], \tag{D.27}$$

and

$$\begin{aligned}
&\sum_{i=1}^{m}\left(\widetilde{V}_{i,1}^{\phi^{\dagger}_{i,h}\diamond\pi_i\times\pi_{-i}}(s_1)-\widetilde{V}_{i,1}^{\phi^{\dagger}_{i,h+1}\diamond\pi_i\times\pi_{-i}}(s_1)\right)\\
=&\text{①}+\text{②}\\
\geq&\mathbb{E}_{\pi,\mathbb{A},P}\left[\sum_{i=1}^{m}\frac{1-\pi_{i,h}\left(s_h,\pi^{\dagger}_{i,h}(s_h)\right)}{2m^2}\Delta^{\dagger-}_{i,h}(s_h)\right]\\
=&\mathbb{E}_{\pi,\mathbb{A},P}\left[\sum_{i=1}^{m}\mathbb{1}(a_{i,h}\neq\pi^{\dagger}_{i,h}(s_h))\frac{\Delta^{\dagger-}_{i,h}(s_h)}{2m^2}\right]\\
\geq&\mathbb{E}_{\pi,\mathbb{A},P}\left[\sum_{i=1}^{m}\mathbb{1}(a_{i,h}\neq\pi^{\dagger}_{i,h}(s_h))\right]\frac{\Delta_{min}}{2m^2}.
\end{aligned}\tag{D.28}$$

In summary,

$$\text{Reg}_i(K,H)\geq\sum_{h=1}^{H}\mathbb{E}_{\pi,\mathbb{A},P}\left[\sum_{i=1}^{m}\mathbb{1}(a_{i,h}\neq\pi^{\dagger}_{i,h}(s_h))\right]\frac{\Delta_{min}}{2m^2}=\mathbb{E}[\text{Loss1}(K,H)]\frac{\Delta_{min}}{2m^2}.\tag{D.29}$$

If the best-in-hindsight regret Reg$(K,H)$ of each agent's algorithm is bounded by a sub-linear bound $\mathcal{R}(T)$, then the attack loss is bounded by $\mathbb{E}[\text{Loss1}(K,H)]\leq 2m^2\mathcal{R}(T)/\Delta_{min}$.

The $d$-portion attack strategy attacks all agents when any agent $i$ chooses an non-target action. We have

$$\begin{aligned}
\text{Cost}(K,H)=&\sum_{k=1}^{K}\sum_{h=1}^{H}\sum_{i=1}^{m}\left(\mathbb{1}(\widetilde{a}^k_{i,h}\neq a^k_{i,h})+|\widetilde{r}^k_{i,h}-r^k_{i,h}|\right)\\
\leq&\sum_{k=1}^{K}\sum_{h=1}^{H}\sum_{i=1}^{m}\mathbb{1}[\widetilde{a}^k_{i,h}\neq a^k_{i,h}]m.
\end{aligned}\tag{D.30}$$

Then, the attack cost is bounded by $m\mathbb{E}[\text{Loss1}(K,H)]\leq 2m^3\mathcal{R}(T)/\Delta_{min}$.

## D.4 Analysis of the $\eta$-gap attack

### D.4.1 Proof of Theorem 15

We assume that the agent does not know the attacker's manipulations and the presence of the attacker. The attacker's manipulations on rewards are stationary. We can consider the combination of the attacker and the environment $\text{MG}(\mathcal{S}, \{\mathcal{A}_i\}_{i=1}^m, H, P, \{R_i\}_{i=1}^m)$ as a new environment $\widetilde{\text{MG}}(\mathcal{S}, \{\mathcal{A}_i\}_{i=1}^m, H, P, \{\widetilde{R}_i\}_{i=1}^m)$, and the agents interact with the new environment. We define $\widetilde{Q}_i$ and $\widetilde{V}_i$ as the $Q$-values and value functions of the new environment $\widetilde{\text{MG}}$ that each agent $i$ observes.

We introduce some special strategy modifications $\{\phi^\dagger_{i,h}\}_{h=1}^H$. For any $h' \geq h$, we have $\phi^\dagger_{i,h} \diamond \pi_{i,h'}(s) = \pi^\dagger_{i,h'}(s)$ and for any $h' < h$, we have $\phi^\dagger_{i,h} \diamond \pi_{i,h'}(s) = \pi_{i,h'}(s)$. Thus,

$$\widetilde{V}_{i,1}^{\pi_i^\dagger \times \pi_{-i}}(s_1) - \widetilde{V}_{i,1}^{\pi}(s_1) = \sum_{h=1}^{H} [\widetilde{V}_{i,1}^{\phi^\dagger_{i,h} \diamond \pi_i \times \pi_{-i}}(s_1) - \widetilde{V}_{i,1}^{\phi^\dagger_{i,h+1} \diamond \pi_i \times \pi_{-i}}(s_1)]. \tag{D.31}$$

We have that for any policy $\pi$,

$$\begin{aligned}
&[\widetilde{V}_{i,1}^{\phi^\dagger_{i,h} \diamond \pi_i \times \pi_{-i}}(s_1) - \widetilde{V}_{i,1}^{\phi^\dagger_{i,h+1} \diamond \pi_i \times \pi_{-i}}(s_1)] \\
=&\mathbb{E}_{\pi, \mathbb{A}, P}\left[\left(\widetilde{V}_{i,h}^{\phi^\dagger_{i,h} \diamond \pi_i \times \pi_{-i}}(s_h) - \widetilde{V}_{i,h}^{\phi^\dagger_{i,h+1} \diamond \pi_i \times \pi_{-i}}(s_h)\right)\right] \\
=&\mathbb{E}_{\pi, \mathbb{A}, P}\left[\left(\mathbb{D}_{\phi^\dagger_{i,h} \diamond \pi_{i,h} \times \pi_{-i,h}} - \mathbb{D}_{\phi^\dagger_{i,h+1} \diamond \pi_{i,h} \times \pi_{-i,h}}\right)\left[\widetilde{Q}_{i,h}^{\phi^\dagger_{i,h+1} \diamond \pi_i \times \pi_{-i}}\right](s_h)\right] \\
=&\mathbb{E}_{\pi, \mathbb{A}, P}\left[\left(\mathbb{D}_{\pi^\dagger_{i,h} \times \pi_{-i,h}} - \mathbb{D}_{\pi_h}\right)\left[\widetilde{R}_{i,h} + \mathbb{P}_h \widetilde{V}_{i,h+1}^{\phi^\dagger_{i,h+1} \diamond \pi_i \times \pi_{-i}}\right](s_h)\right].
\end{aligned} \tag{D.32}$$

Since $\widetilde{R}_{i,h}(s, \boldsymbol{a}) = R_{i,h}(s, \pi^\dagger(s)) - (\eta + (H-h)\Delta_R)\mathbb{1}(a_i \neq \pi^\dagger_{i,h}(s))$ from $\eta$-gap attack strategy and $(H-h)\min_{s' \times a' \times h'} R_{i,h'}(s', a') < \mathbb{P}_h \widetilde{V}_{i,h+1}^{\pi}(s', a') \leq (H-h)\max_{s' \times a' \times h'} R_{i,h'}(s', a')$ for any

$s$ and $a$, we have

$$\begin{aligned}
&[\widetilde{V}_{i,1}^{\phi_{i,h}^{\dagger}\diamond\pi_i\times\pi_{-i}}(s_1)-\widetilde{V}_{i,1}^{\phi_{i,h+1}^{\dagger}\diamond\pi_i\times\pi_{-i}}(s_1)]\\
=&\mathbb{E}_{\pi,\mathbb{A},P}\left[\sum_{\boldsymbol{a}}\pi_h(\boldsymbol{a}|s_h)(\eta+(H-h)\Delta_R)\mathbb{1}(a_i\neq\pi_{i,h}^{\dagger}(s_h))\right]\\
&+\mathbb{E}_{\pi,\mathbb{A},P}\left[\sum_{\boldsymbol{a}}\pi_h(\boldsymbol{a}|s_h)\mathbb{1}(a_i\neq\pi_{i,h}^{\dagger}(s_h))\left(\mathbb{P}_h\widetilde{V}_{i,h+1}^{\phi_{i,h+1}^{\dagger}\diamond\pi_i\times\pi_{-i}}(s_h,\pi_{i,h}^{\dagger}(s_h)\times\boldsymbol{a}_{-i})-\mathbb{P}_h\widetilde{V}_{i,h+1}^{\phi_{i,h+1}^{\dagger}\diamond\pi_i\times\pi_{-i}}(s_h,\boldsymbol{a})\right)\right]\\
\geq&\mathbb{E}_{\pi,\mathbb{A},P}\left[\sum_{\boldsymbol{a}}\pi_h(\boldsymbol{a}|s_h)\eta\mathbb{1}(a_i\neq\pi_{i,h}^{\dagger}(s))\right]\\
=&\mathbb{E}_{\pi,\mathbb{A},P}\left[\mathbb{1}(a_{i,h}\neq\pi_{i,h}^{\dagger}(s))\right]\eta.
\end{aligned} \tag{D.33}$$

Thus,

$$\widetilde{V}_{i,1}^{\pi_i^{\dagger}\times\pi_{-i}}(s_1)-\widetilde{V}_{i,1}^{\pi}(s_1)\geq\mathbb{E}_{\pi,\mathbb{A},P}\left[\sum_{h=1}^{H}\mathbb{1}(a_{i,h}\neq\pi_{i,h}^{\dagger}(s_h))\right]\eta\geq 0. \tag{D.34}$$

The target policy is an NE, and then also is an CE or CCE.

Now we prove that the target policy $\pi_i^{\dagger}$ is the unique {NE, CE, CCE}, when every state $s\in\mathcal{S}$ is reachable at every step $h\in[H]$ under the target policy.

If there exists an CCE $\pi'$ under $\eta$-gap attack, we have $\max_{i\in[m]}(\widetilde{V}_{i,1}^{\dagger,\pi'_{-i}}(s)-\widetilde{V}_{i,1}^{\pi'}(s))=0$ for any initial state $s$. If $\pi'_{i,h}(\cdot|s)\neq\pi_{i,h}^{\dagger}(\cdot|s)$ at a reachable state $s$, we have that $\mathbb{E}_{\pi',\mathbb{A},P}\left[\sum_{h=1}^{H}\mathbb{1}(a_{i,h}\neq\pi_{i,h}^{\dagger}(s_h))\right]$. Thus,

$$\widetilde{V}_{i,1}^{\pi_i^{\dagger}\times\pi'_{-i}}(s_1)-\widetilde{V}_{i,1}^{\pi'}(s_1)\geq\mathbb{E}_{\pi',\mathbb{A},P}\left[\sum_{h=1}^{H}\mathbb{1}(a_{i,h}\neq\pi_{i,h}^{\dagger}(s_h))\right]\eta>0, \tag{D.35}$$

and $\pi'$ is not an CCE. In summary, the target policy $\pi_i^{\dagger}$ is the unique {NE, CE, CCE}.

### D.4.2 Proof of Theorem 16

From the definition of the best-in-hindsight regret and (D.41), we have

$$\begin{aligned}\mathrm{Reg}_i(K,H) =& \max_{\pi'_i}\sum_{k=1}^{K}[\widetilde{V}_{i,1}^{\pi'_i\times\pi^k_{-i}}(s_1^k) - \widetilde{V}_{i,1}^{\pi^k}(s_1^k)] \\ \geq& \sum_{k=1}^{K}[\widetilde{V}_{i,1}^{\pi^\dagger_i\times\pi^k_{-i}}(s_1^k) - \widetilde{V}_{i,1}^{\pi^k}(s_1^k)].\end{aligned} \tag{D.36}$$

From (D.34), we have

$$\mathrm{Reg}_i(K,H) \geq \sum_{k=1}^{K}\mathbb{E}_{\pi^k,\mathbb{A},P}\left[\sum_{h=1}^{H}\mathbb{1}[a_{i,h}^k \neq \pi_{i,h}^\dagger(s_h^k)]\right]\eta \tag{D.37}$$

and

$$\sum_{i=1}^{m}\mathrm{Reg}_i(K,H) = \eta\mathbb{E}[\mathrm{Loss1}(K,H)]. \tag{D.38}$$

If the best-in-hindsight regret $\mathrm{Reg}(K,H)$ of each agent's algorithm is bounded by a sub-linear bound $\mathcal{R}(T)$, then the attack loss is bounded by $\mathbb{E}[\mathrm{Loss1}(K,H)] \leq m\mathcal{R}(T)/\eta$.

The $\eta$-gap attack strategy attacks all agents when any agent $i$ chooses an non-target action. Note that the rewards are bounded in $[0,1]$. We have

$$\begin{aligned}\mathrm{Cost}(K,H) =& \sum_{k=1}^{K}\sum_{h=1}^{H}\sum_{i=1}^{m}\left(\mathbb{1}(\widetilde{a}_{i,h}^k \neq a_{i,h}^k) + |\widetilde{r}_{i,h}^k - r_{i,h}^k|\right) \\ \leq& \sum_{k=1}^{K}\sum_{h=1}^{H}\sum_{i=1}^{m}\mathbb{1}[a_{i,h}^k \neq \pi_{i,h}^\dagger(s_h^k)]m.\end{aligned} \tag{D.39}$$

Hence, the attack cost is bounded by $m\mathbb{E}[\mathrm{Loss1}(K,H)] \leq m^2\mathcal{R}(T)/\eta$.

## D.5 Analysis of the gray-box attacks

### D.5.1 Proof of Theorem 17

We assume that the agent does not know the attacker's manipulations and the presence of the attacker. The attacker's manipulations on actions are stationary. We can consider the combination of the attacker and the environment $\text{MG}(\mathcal{S}, \{\mathcal{A}_i\}_{i=1}^m, H, P, \{R_i\}_{i=1}^m)$ as a new environment $\widetilde{\text{MG}}(\mathcal{S}, \{\mathcal{A}_i\}_{i=1}^m, H, \widetilde{P}, \{\widetilde{R}_i\}_{i=1}^m)$, and the agents interact with the new environment. We define $\widetilde{Q}_i$ and $\widetilde{V}_i$ as the $Q$-values and value functions of the new environment $\widetilde{\text{MG}}$ that each agent $i$ observes.

We first prove that the best response of each agent $i$ towards any policy $\pi_{-i}$ is $\pi_i^\dagger$.

From the mixed attack strategy, we have

$$\widetilde{Q}_{i,h}^{\pi}(s, \boldsymbol{a}) = \mathbb{1}[a_i = \pi_{i,h}^\dagger(s)] R_{i,h}(s, \pi_h^\dagger(s)) + \mathbb{P}_h \widetilde{V}_{i,h+1}^{\pi}(s, \pi_h^\dagger(s)). \tag{D.40}$$

Consider an arbitrary policy $\pi$ and an arbitrary initial state $s_1$. We have

$$\begin{aligned}
&\widetilde{V}_{i,1}^{\pi_i^\dagger \times \pi_{-i}}(s_1) - \widetilde{V}_{i,1}^{\pi}(s_1)\\
=&\widetilde{V}_{i,1}^{\pi_i^\dagger \times \pi_{-i}}(s_1) - \mathbb{D}_\pi[\widetilde{Q}_{i,1}^{\pi_i^\dagger \times \pi_{-i}}](s_1) + \mathbb{D}_\pi[\widetilde{Q}_{i,1}^{\pi_i^\dagger \times \pi_{-i}}](s_1) - \widetilde{V}_{i,1}^{\pi}(s_1)\\
=&\mathbb{E}_{a_{i,1} \sim \pi_{i,1}(\cdot|s_1)} \left[\mathbb{1}[a_{i,1} \neq \pi_{i,1}^\dagger(s_1)] R_{i,1}(s_1, \pi_1^\dagger(s_1))\right] + \mathbb{P}_1 \widetilde{V}_{i,2}^{\pi_i^\dagger \times \pi_{-i}}(s_1, \pi_1^\dagger(s_1)) - \mathbb{P}_1 \widetilde{V}_{i,2}^{\pi}(s_1, \pi_1^\dagger(s_1))\\
=&\mathbb{E}_{a_{i,1} \sim \pi_{i,1}(\cdot|s_1)} \left[\mathbb{1}[a_{i,1} \neq \pi_{i,1}^\dagger(s_1)] R_{i,1}(s_1, \pi_1^\dagger(s_1))\right] + \mathbb{P}_1 [\widetilde{V}_{i,2}^{\pi_i^\dagger \times \pi_{-i}} - \widetilde{V}_{i,2}^{\pi}](s_1, \pi_1^\dagger(s_1))\\
=&\cdots = \mathbb{E}_{\pi, \mathbb{A}, P}\left[\sum_{h=1}^{H} \left(1 - \pi_{i,h}\left(\pi_{i,h}^\dagger(s_h)|s_h\right)\right) R_{i,h}(s_h, \pi_h^\dagger(s_h))\right] \geq 0.
\end{aligned} \tag{D.41}$$

Since $R_{i,h}(s_h, \pi_h^\dagger(s_h)) > 0$, $\widetilde{V}_{i,1}^{\pi_i^\dagger \times \pi_{-i}}(s_1) - \widetilde{V}_{i,1}^{\pi}(s_1) = 0$ holds if and only if $\pi_i^\dagger = \pi_i$ holds for the states that are reachable under policy $\pi^\dagger$. We conclude that the best response of each agent $i$ towards any policy $\pi_{-i}$ is $\pi_i^\dagger$ under the mixed attack strategy. The target policy $\pi^\dagger$ is an NE, CE, CCE under the mixed attack strategy.

Now we prove that the target policy $\pi_i^\dagger$ is the unique {NE, CE, CCE} under the mixed attack strategy, when every state $s \in \mathcal{S}$ is reachable at every step $h \in [H]$ under the target policy.

If there exists an CCE $\pi'$ under the mixed attack strategy, we have $\max_{i\in[m]}(\widetilde{V}_{i,1}^{\dagger,\pi'_{-i}}(s) - \widetilde{V}_{i,1}^{\pi'}(s)) = 0$ for any initial state $s$.

From (D.41), we have that if $\pi'_{i,h}(\cdot|s) \neq \pi^{\dagger}_{i,h}(\cdot|s)$ at a reachable state $s$, $\widetilde{V}_{i,1}^{\pi_i^{\dagger}\times\pi'_{-i}}(s_1) - \widetilde{V}_{i,1}^{\pi'}(s_1) > 0$. Then $\widetilde{V}_{i,1}^{\dagger,\pi'_{-i}}(s_1) > \widetilde{V}_{i,1}^{\pi_i^{\dagger}\times\pi'_{-i}}(s_1) > \widetilde{V}_{i,1}^{\pi'}(s_1)$. $\pi'$ is not an CCE in this case.

We can conclude that $\pi' = \pi^{\dagger}$ for the states that are reachable under policy $\pi'$. If every state $s \in \mathcal{S}$ is reachable at every step $h \in [H]$ under the target policy, $\pi^{\dagger}$ is the unique {NE, CE, CCE}.

### D.5.2 Proof of Theorem 18

We set $R_{min} = \min_{h\in[H]}\min_{s\in\mathcal{S}}\min_{i\in[m]} R_{i,h}(s, \pi^{\dagger}_h(s))$. From the definition of the best-in-hindsight regret and (D.41), we have

$$\begin{aligned}
\text{Reg}_i(K,H) =& \max_{\pi'_i}\sum_{k=1}^{K}[\widetilde{V}_{i,1}^{\pi'_i\times\pi^k_{-i}}(s_1^k) - \widetilde{V}_{i,1}^{\pi^k}(s_1^k)] \\
\geq& \sum_{k=1}^{K}[\widetilde{V}_{i,1}^{\pi_i^{\dagger}\times\pi^k_{-i}}(s_1^k) - \widetilde{V}_{i,1}^{\pi^k}(s_1^k)] \\
=& \sum_{k=1}^{K}\mathbb{E}_{\pi^k,\mathbb{A},P}\left[\sum_{h=1}^{H}\left(1 - \pi^k_{i,h}\left(\pi^{\dagger}_{i,h}(s_h^k)|s_h^k\right)\right)R_{i,h}(s_h^k, \pi^{\dagger}_h(s_h^k))\right] \qquad \text{(D.42)} \\
=& \sum_{k=1}^{K}\mathbb{E}_{\pi^k,\mathbb{A},P}\left[\sum_{h=1}^{H}\mathbb{1}[a^k_{i,h} \neq \pi^{\dagger}_{i,h}(s_h^k)]R_{i,h}(s_h^k, \pi^{\dagger}_h(s_h^k))\right] \\
\geq& R_{min}\sum_{k=1}^{K}\mathbb{E}_{\pi^k,\mathbb{A},P}\left[\sum_{h=1}^{H}\mathbb{1}[a^k_{i,h} \neq \pi^{\dagger}_{i,h}(s_h^k)]\right]
\end{aligned}$$

and

$$\sum_{i=1}^{m}\text{Reg}_i(K,H) \geq R_{min}\mathbb{E}[\text{Loss1}(K,H)]. \qquad \text{(D.43)}$$

If the best-in-hindsight regret $\text{Reg}(K,H)$ of each agent's algorithm is bounded by a sub-linear bound $\mathcal{R}(T)$ under the mixed attack strategy, then the attack loss is bounded by $\mathbb{E}[\text{Loss1}(K,H)] \leq m\mathcal{R}(T)/R_{min}$.

The mixed attack strategy only attacks agent $i$ when agent $i$ chooses a non-target action. We have

$$\begin{aligned}\text{Cost}(K,H) =& \sum_{k=1}^{K}\sum_{h=1}^{H}\sum_{i=1}^{m}\left(\mathbb{1}(\widetilde{a}_{i,h}^{k} \neq a_{i,h}^{k}) + |\widetilde{r}_{i,h}^{k} - r_{i,h}^{k}|\right) \\ \leq& \sum_{k=1}^{K}\sum_{h=1}^{H}\sum_{i=1}^{m}\mathbb{1}[\widetilde{a}_{i,h}^{k} \neq a_{i,h}^{k}]\,(1+1)\,.\end{aligned} \tag{D.44}$$

Then, the attack cost is bounded by $2\mathbb{E}[\text{Loss1}(K,H)] \leq 2m\mathcal{R}(T)/R_{min}$.

## D.6 Analysis of the black-box attacks

### D.6.1 Proof of Lemma 14

We denote by $\overline{Q}_{\dagger,h}^{k}$, $\underline{Q}_{\dagger,h}^{k}$, $\overline{V}_{\dagger,h}^{k}$ $\underline{V}_{\dagger,h}^{k}$, $N_{h}^{k}$, $\hat{\mathbb{P}}_{h}^{k}$, $\pi_{h}^{k}$ and $\hat{R}_{\dagger,h}^{k}$ the observations of the approximate mixed attacker at the beginning of episode $k$ and time step $h$. As before, we begin with proving that the estimations are indeed upper and lower bounds of the corresponding $Q$-values and state value functions. We use $\pi^*$ to denote the optimal policy that maximizes the attacker's rewards, i.e. $V_{\dagger,1}^{\pi^*}(s) = \max_{\pi} V_{\dagger,1}^{\pi}(s)$.

**Lemma 15.** With probability $1-p$, for any $(s,\boldsymbol{a},h)$ and $k \leq \tau$,

$$\overline{Q}_{\dagger,h}^{k}(s,\boldsymbol{a}) \geq Q_{\dagger,h}^{\pi^*}(s,\boldsymbol{a}),\ \underline{Q}_{\dagger,h}^{k}(s,\boldsymbol{a}) \leq Q_{\dagger,h}^{\pi^k}(s,\boldsymbol{a}), \tag{D.45}$$

$$\overline{V}_{\dagger,h}^{k}(s) \geq V_{\dagger,h}^{\pi^*}(s),\ \underline{V}_{\dagger,h}^{k}(s) \leq V_{\dagger,h}^{\pi^k}(s). \tag{D.46}$$

**Proof.** For each fixed $k$, we prove this by induction from $h = H+1$ to $h = 1$. For the step $H+1$, we have $\overline{V}_{\dagger,H+1}^{k} = \underline{V}_{\dagger,H+1}^{k} = Q_{\dagger,H+1}^{\pi^*} = \mathbf{0}$. Now, we assume inequality (D.46) holds for the step

$h+1$. By the definition of $Q$-values and Algorithm 5.1, we have

$$\begin{aligned}
&\overline{Q}^k_{\dagger,h}(s,\boldsymbol{a}) - Q^{\pi^*}_{\dagger,h}(s,\boldsymbol{a})\\
=&\hat{R}^k_{\dagger,h}(s,\boldsymbol{a}) - R^k_{\dagger,h}(s,\boldsymbol{a}) + \hat{\mathbb{P}}^k_h\overline{V}^k_{\dagger,h+1}(s,\boldsymbol{a}) - \mathbb{P}_h V^{\pi^*}_{i,h}(s,\boldsymbol{a}) + B(N^k_h(s,\boldsymbol{a}))\\
=&\hat{\mathbb{P}}^k_h(\overline{V}^k_{\dagger,h+1} - V^{\pi^*}_{\dagger,h})(s,\boldsymbol{a}) + (\hat{R}^k_{\dagger,h} - R^k_{\dagger,h})(s,\boldsymbol{a}) + (\hat{\mathbb{P}}^k_h - \mathbb{P}_h)V^{\pi^*}_{\dagger,h}(s,\boldsymbol{a}) + B(N^k_h(s,\boldsymbol{a})).
\end{aligned} \tag{D.47}$$

Recall that $B(N) = (H\sqrt{S}+1)\sqrt{\log(2AH\tau/p)/(2N)}$. By Azuma-Hoeffding inequality, we have that with probability $1-2p/SAH$,

$$\forall k \le \tau, \left|\hat{R}^k_{\dagger,h}(s,\boldsymbol{a}) - R^k_{\dagger,h}(s,\boldsymbol{a})\right| \le \sqrt{\frac{\log(2SAH\tau/p)}{2N^k_h(s,\boldsymbol{a})}}, \tag{D.48}$$

and

$$\forall k \le \tau, \left|(\hat{\mathbb{P}}^k_h - \mathbb{P}_h)V^{\pi^*}_{\dagger,h}(s,\boldsymbol{a})\right| \le H\sqrt{\frac{S\log(2SAH\tau/p)}{2N^k_h(s,\boldsymbol{a})}}. \tag{D.49}$$

Putting everything together, we have $\overline{Q}^k_{\dagger,h}(s,\boldsymbol{a}) - Q^{\pi^*}_{\dagger,h}(s,\boldsymbol{a}) \ge \hat{\mathbb{P}}^k_h(\overline{V}^k_{\dagger,h+1} - V^{\pi^*}_{\dagger,h})(s,\boldsymbol{a}) \ge 0$. Similarly, $\underline{Q}^k_{\dagger,h}(s,\boldsymbol{a}) \le Q^{\pi^k}_{\dagger,h}(s,\boldsymbol{a})$.

Now we assume inequality (D.45) holds for the step $h$. As discussed above, if inequality (D.46) holds for the step $h+1$, inequality (D.45) holds for the step $h$. By Algorithm 5.1, we have

$$\overline{V}^k_{\dagger,h}(s) = \overline{Q}^k_{\dagger,h}(s,\pi^k_h(s)) \ge \overline{Q}^k_{\dagger,h}(s,\pi^*_h(s)) \ge Q^{\pi^*}_{\dagger,h}(s,\pi^*_h(s)) = V^{\pi^*}_{\dagger,h}(s). \tag{D.50}$$

Similarly, $\underline{V}^k_{\dagger,h}(s) \le V^{\pi^k}_{\dagger,h}(s)$. $\square$

Now, we are ready to prove Lemma 14. By Azuma-Hoeffding inequality, we have that with probability $1-2p$,

$$\left|\left(\mathbb{E}_{s_1\sim P_0(\cdot)} - \mathbb{E}_{s_1\sim\hat{\mathbb{P}}_0(\cdot)}\right)\left[V^{\pi^*}_{\dagger,1}(s_1) - V^{\pi^k}_{\dagger,1}(s_1)\right]\right| \le H\sqrt{\frac{S\log(2\tau/p)}{2k}}, \tag{D.51}$$

and $\forall k \le \tau$,

$$\left|\sum_{k'=1}^{k}\left(\mathbb{E}_{s_1\sim P_0(\cdot)} - \mathbb{1}(s_1 = s_1^{k'})\right)\left[V_{\dagger,1}^{\pi^*}(s_1) - V_{\dagger,1}^{\pi^{k'}}(s_1)\right]\right| \le H\sqrt{\frac{S\log(2\tau/p)}{2k}}. \tag{D.52}$$

Thus, for any $k \le \tau$,

$$\begin{aligned} &\mathbb{E}_{s_1\sim P_0(\cdot)}\left[V_{\dagger,1}^{\pi^*}(s_1) - V_{\dagger,1}^{\pi^k}(s_1)\right] \\ \le&\mathbb{E}_{s_1\sim\hat{\mathbb{P}}_0^k(\cdot)}\left[V_{\dagger,1}^{\pi^*}(s_1) - V_{\dagger,1}^{\pi^k}(s_1)\right] + H\sqrt{S\log(2\tau/p)/(2k)} \\ \le&\mathbb{E}_{s_1\sim\hat{\mathbb{P}}_0^k(\cdot)}\left[\overline{V}_{\dagger,1}^k(s_1) - \underline{V}_{\dagger,1}^k(s_1)\right] + H\sqrt{S\log(2\tau/p)/(2k)}. \end{aligned} \tag{D.53}$$

According to (D.51) and (D.52), we have

$$\begin{aligned} &\sum_{k=1}^{\tau}\left(\mathbb{E}_{s_1\sim\hat{\mathbb{P}}_0^k(\cdot)}\left[\overline{V}_{\dagger,1}^k(s_1) - \underline{V}_{\dagger,1}^k(s_1)\right] + H\sqrt{S\log(2\tau/p)/(2k)}\right) \\ \le&\sum_{k=1}^{\tau}\left(\overline{V}_{\dagger,1}^k(s_1^k) - \underline{V}_{\dagger,1}^k(s_1^k)\right) + \sum_{k=1}^{\tau}3H\sqrt{S\log(2\tau/p)/(2k)}. \end{aligned} \tag{D.54}$$

We define $\Delta V_h^k(s) = \overline{V}_{\dagger,h}^k(s) - \underline{V}_{\dagger,h}^k(s)$, $\Delta Q_h^k(s,\boldsymbol{a}) = \overline{Q}_{\dagger,h}^k(s,\boldsymbol{a}) - \underline{Q}_{\dagger,h}^k(s,\boldsymbol{a})$. By the update equations in Algorithm 5.1, we have $\Delta Q_h^k(s,\boldsymbol{a}) \le \hat{\mathbb{P}}_h^k\Delta V_{h+1}^k(s,\boldsymbol{a}) + 2B(N_h^k(s,\boldsymbol{a}))$ and $\Delta V_h^k(s) = \Delta Q_h^k(s,\pi_h^k(s))$. We define $\psi_h^k = \Delta V_h^k(s_h^k) = \Delta Q_h^k(s_h^k, a_h^k)$. From (D.49) and (D.56), we have

$$\begin{aligned} \psi_h^k \le&\hat{\mathbb{P}}_h^k\Delta V_{h+1}^k(s_h^k,\boldsymbol{a}_{\boldsymbol{h}}^{\boldsymbol{k}}) + 2B(N_h^k(s_h^k,\boldsymbol{a}_h^k)) \\ \le&\mathbb{P}_h^k\Delta V_{h+1}^k(s_h^k,\boldsymbol{a}_{\boldsymbol{h}}^{\boldsymbol{k}}) + 3B(N_h^k(s_h^k,\boldsymbol{a}_h^k)) \\ \le&\mathbb{P}_h^k\Delta V_{h+1}^k(s_h^k,\boldsymbol{a}_{\boldsymbol{h}}^{\boldsymbol{k}}) - \psi_{h+1}^k + \psi_{h+1}^k + 3B(N_h^k(s_h^k,\boldsymbol{a}_h^k)). \end{aligned} \tag{D.55}$$

By Azuma-Hoeffding inequality, we have that with probability $1 - p/H$, $\forall k \le \tau$,

$$\left|\sum_{k'=1}^{k}|\mathbb{P}_h^{k'}\Delta V_{h+1}^k(s_h^{k'},\boldsymbol{a}_{\boldsymbol{h}}^{\boldsymbol{k'}}) - \psi_{h+1}^{k'}|\right| \le H\sqrt{\frac{S\log(2H\tau/p)}{2k}}. \tag{D.56}$$

Since $\psi_{H+1}^k = 0$ for all $k$, we have

$$\begin{aligned}\sum_{k=1}^{\tau} \psi_1^k \leq & \sum_{k=1}^{\tau}\sum_{h=1}^{H} |\mathbb{P}_h^k \Delta V_{h+1}^k(s_h^k, \boldsymbol{a}_{\boldsymbol{h}}^{\boldsymbol{k}}) - \psi_{h+1}^k| + \sum_{k=1}^{\tau}\sum_{h=1}^{H} 3B(N_h^k(s_h^k, \boldsymbol{a}_h^k)) \\ \leq & \sum_{h=1}^{H} H\sqrt{\frac{S\log(2H\tau/p)}{2\tau}} + \sum_{h=1}^{H}\sum_{(s,\boldsymbol{a})}\sum_{n=1}^{N_h^\tau(s,\boldsymbol{a})} (H\sqrt{S}+1)\sqrt{\frac{\log(2SAH\tau/p)}{2n}} \\ \leq & H^2\sqrt{\frac{S\log(2H\tau/p)}{2\tau}} + H(H\sqrt{S}+1)\sqrt{2SA\tau\log(2SAH\tau/p)}\end{aligned} \tag{D.57}$$

and therefore

$$\begin{aligned}& \sum_{k=1}^{\tau}\left(\mathbb{E}_{s_1\sim\hat{\mathbb{P}}_0^k(\cdot)}\left[\overline{V}_{\dagger,1}^k(s_1) - \underline{V}_{\dagger,1}^k(s_1)\right] + H\sqrt{S\log(2\tau/p)/(2k)}\right) \\ \leq & H^2\sqrt{\frac{S\log(2H\tau/p)}{2\tau}} + 3H\sqrt{2S\tau\log(2\tau/p)} + H(H\sqrt{S}+1)\sqrt{2SA\tau\log(2SAH\tau/p)}.\end{aligned} \tag{D.58}$$

Since

$$\pi^\dagger = \min_{\pi_k}\left(\mathbb{E}_{s_1\sim\hat{\mathbb{P}}_0^k(\cdot)}\left[\sum_{i=1}^{m}\left(V_{\dagger,1}^{\pi^*}(s_1) - V_{\dagger,1}^{\pi^k}(s_1)\right)\right] + H\sqrt{S\log(2\tau/p)/(2k)}\right), \tag{D.59}$$

$$\begin{aligned}& \mathbb{E}_{s_1\sim P_0(\cdot)}\left[V_{\dagger,1}^{\pi^*}(s_1) - V_{\dagger,1}^{\pi^\dagger}(s_1)\right] \\ \leq & \mathbb{E}_{s_1\sim\hat{\mathbb{P}}_0^k(\cdot)}\left[V_{\dagger,1}^{\pi^*}(s_1) - V_{\dagger,1}^{\pi^\dagger}(s_1)\right] + H\sqrt{S\log(2\tau/p)/(2k)} \\ \leq & H^2\sqrt{\frac{S\log(2H\tau/p)}{2\tau^3}} + 3H\sqrt{2S\log(2\tau/p)/\tau} + H(H\sqrt{S}+1)\sqrt{2SA\log(2SAH\tau/p)/\tau} \\ \leq & 2H^2S\sqrt{2A\log(2SAH\tau/p)/\tau},\end{aligned} \tag{D.60}$$

where the last inequality holds when $S, H, A \geq 2$. Similarly,

$$\sum_{k=1}^{K}\left[V_{\dagger,1}^{\pi^*}(s_1^k) - V_{\dagger,1}^{\pi^\dagger}(s_1^k)\right] \leq 2H^2S\sqrt{2A\log(2SAH\tau/p)/\tau}. \tag{D.61}$$

### D.6.2 Proof of Theorem 19

We use the same learning rate $\alpha_t$ in [42]. We also use an auxiliary sequence $\{\alpha_t^i\}_{i=1}^t$ defined in [42] based on the learning rate, which will be frequently used in the proof:

$$\alpha_t = \frac{H+1}{H+t},\ \alpha_t^0 = \prod_{j=1}^t (1-\alpha_j),\ \alpha_t^i = \alpha_i \prod_{j=i+1}^t (1-\alpha_j). \tag{D.62}$$

We follow the requirement for the adversarial bandit algorithm used in V-learning, which is to have a high probability weighted external regret guarantee as follows.

**Assumption 1.** For any $t \in \mathbb{N}$ and any $\delta \in (0,1)$, with probability at least $1-\delta$, we have

$$\max_\theta \sum_{j=1}^t \alpha_t^j [\langle \theta_j, l_j \rangle - \langle \theta, l_j \rangle] \leq \xi(B, t, \log(1/\delta)). \tag{D.63}$$

In addition, there exists an upper bound $\Xi(B, t, \log(1/\delta)) \geq \sum_{t'=1}^t \xi(B, t, \log(1/\delta))$ where (i) $\xi(B, t, \log(1/\delta))$ is non-decreasing in $B$ for any $t$, $\delta$; (ii) $\Xi(B, t, \log(1/\delta))$ is concave in $t$ for any $B$, $\delta$.

In particular, it was proved in [42] that the Follow-the-Regularized-Leader (FTRL) algorithm (Algorithm 5.3) satisfies Assumption 1 with bounds $\xi(B, t, \log(1/\delta)) \leq \mathcal{O}(\sqrt{HB\log(B/\delta)/t})$ and $\Xi(B, t, \log(1/\delta)) \leq \mathcal{O}(\sqrt{HBt\log(B/\delta)})$. By choosing hyper-parameter $w_t = \alpha_t \left(\prod_{i=2}^t (1-\alpha_i)\right)^{-1}$ and $\gamma_t = \sqrt{\frac{H\log B}{Bt}}$, $\xi(B, t, \log(1/\delta)) = 10\sqrt{HB\log(B/\delta)/t}$ and $\Xi(B, t, \log(1/\delta)) = 20\sqrt{HBt\log(B/\delta)}$.

We use $V^k$, $N^k$, $\pi^k$ to denote the value, counter and policy maintained by V-learning algorithm at the beginning of the episode $k$. Suppose $s$ was previously visited at episodes $k^1, \cdots, k^t < k$ at the step $h$. Set $t'$ such that $k^{t'} \leq \tau$ and $k^{t'+1} > \tau$.

In the exploration phase of the proposed approximate mixed attack strategy, the rewards are equal to $1$ for any state $s$, any action $\boldsymbol{a}$, any agent $i$ and any step $h$. The loss updated to the adversarial bandit update step in Algorithm 5.2 is equal to $\frac{h-1}{H}$.

In the attack phase, the expected loss updated to the adversarial bandit update step in

Algorithm 5.2 is equal to

$$\sum_{j=1}^{t'} \alpha_t^j \frac{h-1}{H} + \sum_{j=t'+1}^{t} \alpha_t^j \mathbb{D}_{\pi^\dagger} \left( \frac{H - \mathbb{P}_h V_{i,h+1}^{k^j}}{H} \right)(s) + \sum_{j=t'+1}^{t} \alpha_t^j \mathbb{D}_{\pi_h^{k^j}} \left( \frac{-\widetilde{r}_{i,h}}{H} \right)(s). \tag{D.64}$$

Thus, in both of the exploration phase and the attack phase, $\pi^\dagger$ is the best policy for the adversarial bandit algorithm.

By Assumption 1 and the adversarial bandit update step in Algorithm 5.2, with probability at least $1-\delta$, for any $(s,h) \in \mathcal{S} \times [H]$ and any $k > \tau$, we have

$$\begin{aligned} \xi(A,t,\iota) \geq & \sum_{j=1}^{t'} \alpha_t^j \frac{h-1}{H} + \sum_{j=t'+1}^{t} \alpha_t^j \mathbb{D}_{\pi^\dagger} \left( \frac{H - \mathbb{P}_h V_{i,h+1}^{k^j}}{H} \right)(s) + \sum_{j=t'+1}^{t} \alpha_t^j \mathbb{D}_{\pi_h^{k^j}} \left( \frac{-\widetilde{r}_{i,h}}{H} \right)(s) \\ & - \sum_{j=1}^{t'} \alpha_t^j \frac{h-1}{H} + \sum_{j=t'+1}^{t} \alpha_t^j \mathbb{D}_{\pi^\dagger} \left( \frac{H - \mathbb{P}_h V_{i,h+1}^{k^j}}{H} \right)(s) + \sum_{j=t'+1}^{t} \alpha_t^j \mathbb{D}_{\pi^\dagger} \left( \frac{-\widetilde{r}_{i,h}}{H} \right)(s) \\ = & \sum_{j=t'+1}^{t} \alpha_t^j \left( 1 - \pi_{i,h}^{k^j}(\pi_{i,h}^\dagger(s)|s) \right) \frac{r_{i,h}(s, \pi_h^\dagger(s))}{H}, \end{aligned} \tag{D.65}$$

where $\iota = \log(mHSAK/\delta)$.

Note that $R_{min} = \min_{h\in[H]} \min_{s\in\mathcal{S}} \min_{i\in[m]} R_{i,h}(s, \pi_h^\dagger(s))$. We have

$$\frac{H}{R_{min}} \xi(A,t,\iota) \geq \sum_{j=t'+1}^{t} \alpha_t^j \left( 1 - \pi_{i,h}^{k^j}(\pi_{i,h}^\dagger(s)|s) \right). \tag{D.66}$$

Let $n_h^k = N_h^k(s_h^k)$ and suppose $s_h^k$ was previously visited at episodes $k^1, \cdots, k^{n_h^k} < k$ at the step $h$. Let $k^j(s)$ denote the episode that $s$ was visited in $j$-th time.

$$\frac{H}{R_{min}} \xi(A, n_h^k, \iota) \geq \sum_{j=N_h^\tau(s_h^k)+1}^{n_h^k} \alpha_{n_h^k}^j \left( 1 - \pi_{i,h}^{k^j}(\pi_{i,h}^\dagger(s_h^k)|s_h^k) \right). \tag{D.67}$$

According to the property of the learning rate $\alpha_t$, we have

$$\frac{H}{R_{min}\alpha_t^t}\xi(A, n_h^k, \iota) + \sum_{j=N_h^\tau(s_h^k)+1}^{n_h^k-1} \frac{H}{R_{min}}\xi(A, j, \iota) \leq \sum_{j=N_h^\tau(s_h^k)+1}^{n_h^k} \left(1 - \pi_{i,h}^{k^j}(\pi_{i,h}^\dagger(s_h^k)|s_h^k)\right). \tag{D.68}$$

Then,

$$\frac{40H}{R_{min}}\sqrt{HAn_h^k\iota} \leq \sum_{j=N_h^\tau(s_h^k)+1}^{n_h^k} \left(1 - \pi_{i,h}^{k^j}(\pi_{i,h}^\dagger(s_h^k)|s_h^k)\right). \tag{D.69}$$

Computing the summation of the above inequality over $h$ and $s$, we have

$$\begin{aligned}
&\mathbb{E}\left[\sum_{h=1}^{H}\sum_{k=\tau+1}^{K} \mathbb{1}[a_{i,h}^k \neq \pi_{i,h}^\dagger(s_h^k)]\right] \\
=&\sum_{h=1}^{H}\sum_{s\in\mathcal{S}}\sum_{j=N_h^\tau(s)+1}^{N_h^K(s)} \left(1 - \pi_{i,h}^{k^j(s)}(\pi_{i,h}^\dagger(s)|s)\right) \\
\leq&\sum_{s\in\mathcal{S}}\frac{40}{R_{min}}\sqrt{H^5AN_h^K(s)\iota} \\
\leq&\frac{40}{R_{min}}\sqrt{H^5ASK\iota}.
\end{aligned} \tag{D.70}$$

In the exploration phase, the loss at each episode is up to $H$. In the attack phase, the expected number of episodes that the agents do not follow $\pi^\dagger$ is up to $\frac{40}{R_{min}}m\sqrt{H^7ASK\iota}$.

According to Lemma 14, the attack loss is bounded by

$$\mathbb{E}\left[\text{loss}(K, H)\right] \leq H\tau + \frac{40}{R_{min}}m\sqrt{H^9ASK\iota} + 2H^2SK\sqrt{2A\iota/\tau}. \tag{D.71}$$

In the exploration phase, the approximate mixed attack strategy attacks at any step and any episode. In the attack phase, the approximate mixed attack strategy only attacks agent $i$ when agent $i$ chooses a non-target action. We have

$$\begin{aligned}\text{Cost}(K,H) =&\sum_{k=1}^{K}\sum_{h=1}^{H}\sum_{i=1}^{m}\left(\mathbb{1}(\widetilde{a}_{i,h}^k \neq a_{i,h}^k) + |\widetilde{r}_{i,h}^k - r_{i,h}^k|\right)\\ \leq&\sum_{k=1}^{\tau}\sum_{h=1}^{H}\sum_{i=1}^{m}(1+1) + \sum_{k=\tau=1}^{K}\sum_{h=1}^{H}\sum_{i=1}^{m}\mathbb{1}[\widetilde{a}_{i,h}^k \neq a_{i,h}^k]\,(1+1)\,.\end{aligned} \tag{D.72}$$

Then, the attack cost is bounded by

$$\mathbb{E}\left[\text{Cost}(K,H)\right] \leq 2mH\tau + \frac{80}{R_{min}}\sqrt{H^5ASK\iota}. \tag{D.73}$$

For the executing output policy $\hat{\pi}$ of V-learning, we have

$$\begin{aligned}&1-\hat{\pi}_{i,h}(\pi_{i,h}^{\dagger}(s)|s)\\ =&\frac{1}{K}\sum_{k=1}^{K}\sum_{j=1}^{N_h^k(s)}\alpha_{N_h^k(s)}^j\left(1-\pi_{i,h}^{k^j(s)}(\pi_{i,h}^{\dagger}(s)|s)\right)\\ =&\frac{1}{K}\sum_{k=\tau+1}^{K}\sum_{j=N_h^{\tau}(s)+1}^{N_h^k(s)}\alpha_{N_h^k(s)}^j\left(1-\pi_{i,h}^{k^j(s)}(\pi_{i,h}^{\dagger}(s)|s)\right) + \frac{1}{K}\sum_{k=1}^{\tau}\sum_{j=1}^{N_h^k(s)}\alpha_{N_h^k(s)}^j\left(1-\pi_{i,h}^{k^j(s)}(\pi_{i,h}^{\dagger}(s)|s)\right)\\ &+\frac{1}{K}\sum_{k=\tau+1}^{K}\sum_{j=1}^{N_h^{\tau}(s)}\alpha_{N_h^k(s)}^j\left(1-\pi_{i,h}^{k^j(s)}(\pi_{i,h}^{\dagger}(s)|s)\right)\\ \leq&\frac{20}{R_{min}}\sqrt{\frac{H^3A\iota}{K}} + \frac{2\tau}{K}.\end{aligned} \tag{D.74}$$

The probability that the agents with $\hat{\pi}$ do not follow the target policy is bounded by $\frac{20mS}{R_{min}}\sqrt{\frac{H^5A\iota}{K}} + \frac{2\tau mSH}{K}$.

According to Lemma 14, the attack loss of the executing output policy $\hat{\pi}$ is upper bounded by

$$V_{\dagger,1}^{\pi^*}(s_1) - V_{\dagger,1}^{\hat{\pi}}(s_1) \leq H\left(\frac{20mS}{R_{min}}\sqrt{\frac{H^5A\iota}{K}} + \frac{2\tau mSH}{K}\right) + 2H^2S\sqrt{2A\iota/\tau}. \tag{D.75}$$

# Appendix E

# Appendix of Chapter 6

## E.1 Proof of Proposition 4

The uncertainty set of the policy execution has the form in:

$$\Pi^{\rho}(\pi) := \{\widetilde{\pi} | \forall s, \widetilde{\pi}_h(\cdot|s) = (1-\rho)\pi(\cdot|s) + \rho\pi'_h(\cdot|s), \pi'_h(\cdot|s) \in \Delta_{\mathcal{A}}\}. \tag{E.1}$$

We define

$$\begin{aligned} C_h^{\pi,\pi',\rho}(s) &:= \mathbb{E}\left[\sum_{h'=h}^{H} R_{h'}(s_{h'}, a_{h'}) | s_h = s, a_{h'} \sim \widetilde{\pi}_{h'}(\cdot|s_{h'})\right] \\ D_h^{\pi,\pi',\rho}(s,a) &:= \mathbb{E}\left[\sum_{h'=h}^{H} R_{h'}(s_{h'}, a_{h'}) | s_h = s, a_h = a, a_{h'} \sim \widetilde{\pi}_{h'}(\cdot|s_{h'})\right]. \end{aligned}$$

**Robust Bellman Equation** First we prove the action robust Bellman equation holds for any policy $\pi$, state $s$ action $a$ and step $h$. From the definition of the robust value function in (6.1), we have $V_{H+1}^{\pi}(s) = 0, \forall s \in \mathcal{S}$.

We prove the robust Bellman equation by building a policy $\pi^-$. Here, policy $\pi^-$ is the optimal adversarial policy towards the policy $\pi$.

At step $H$, we set $\pi_H^-(s) = \arg\min_{a\in\mathcal{A}} R_H(s,a)$. We have

$$\begin{aligned} V_H^\pi(s) &= \min_{\pi'} C_H^{\pi,\pi',\rho}(s) \\ &= (1-\rho)[\mathbb{D}_{\pi_H} R_H](s) + \rho \min_{\pi'} [\mathbb{D}_{\pi'_H} R_H](s) \\ &= (1-\rho)[\mathbb{D}_{\pi_H} Q_H^\pi](s) + \rho \min_{a\in\mathcal{A}} Q_H^\pi(s,a) = C_H^{\pi,\pi^-,\rho}(s), \end{aligned} \tag{E.2}$$

as $V_{H+1} = 0$.

The robust Bellman equation holds at step $H$ and $\min_{\pi'} \sum_s w(s) C_H^{\pi,\pi',\rho}(s) = \sum_s w(s) \min_{\pi'} C_H^{\pi,\pi',\rho}(s) = \sum_s w(s) C_H^{\pi,\pi^-,\rho}(s)$ for any state $s$ and any weighted function $w : \mathcal{S} \to \Delta_{\mathcal{S}}$.

Suppose the robust Bellman equation holds at step $h+1$ and $\min_{\pi'} \sum_s w(s) C_{h+1}^{\pi,\pi',\rho}(s) = \sum_s w(s) \min_{\pi'} C_{h+1}^{\pi,\pi',\rho}(s) = \sum_s w(s) C_{h+1}^{\pi,\pi^-,\rho}(s)$ for any state $s$ and any weighted function $w : \mathcal{S} \to \Delta_{\mathcal{S}}$.

Now we prove the robust Bellman equation holds at step $h$. From the definition of the robust $Q$-function in (6.2) and the form of uncertainty set, we have

$$\begin{aligned} Q_h^\pi(s,a) &= \min_{\widetilde{\pi}\in\Pi(\pi)} \mathbb{E}\left[\sum_{h'=h}^{H} R_{h'}(s_{h'}, a_{h'}) | s_h = s, a_h = a, a_{h'} \sim \widetilde{\pi}_{h'}(\cdot|s_{h'})\right] \\ &= \min_{\pi'} D_h^{\pi,\pi',\rho}(s,a) \\ &= R_h(s,a) + \min_{\pi'} \mathbb{E}_{s'\sim P_h(\cdot|s,a)} C_{h+1}^{\pi,\pi',\rho}(s) \\ &= R_h(s,a) + \mathbb{E}_{s'\sim P_h(\cdot|s,a)} \min_{\pi'} C_{h+1}^{\pi,\pi',\rho}(s) \\ &= R_h(s,a) + [P_h V_{h+1}^\pi](s,a). \end{aligned} \tag{E.3}$$

We also have that $Q_h^\pi(s,a) = D_h^{\pi,\pi^-,\rho}(s,a)$.

Recall that a (stochastic) Markov policy is a set of $H$ maps $\pi := \{\pi_h : \mathcal{S} \to \Delta_{\mathcal{A}}\}_{h\in[H]}$. From the definition of the robust value function in (6.1) and the form of uncertainty set, we have

$$\begin{aligned}
V_h^\pi(s) =& \min_{\widetilde{\pi}\in\Pi(\pi)} \mathbb{E}\left[\sum_{h'=h}^{H} R_{h'}(s_{h'}, a_{h'})|s_h = s, a_{h'} \sim \widetilde{\pi}_{h'}(\cdot|s_{h'})\right] \\
=& \min_{\pi'} C_h^{\pi,\pi',\rho}(s) \\
=& \min_{\pi'_h} \min_{\{\pi'_{h'}\}_{h'=h+1}^H} C_h^{\pi,\pi',\rho}(s) \\
\geq& (1-\rho) \min_{\{\pi'_{h'}\}_{h'=h+1}^H} \mathbb{E}_{a\sim\pi_h(\cdot|s)} D_h^{\pi,\pi',\rho}(s,a) + \rho \min_{\pi'_h} \min_{\{\pi'_{h'}\}_{h'=h+1}^H} \mathbb{E}_{a\sim\pi'_h(\cdot|s)} D_h^{\pi,\pi',\rho}(s,a) \\
\geq& (1-\rho)\mathbb{E}_{a\sim\pi_h(\cdot|s)} \min_{\{\pi'_{h'}\}_{h'=h+1}^H} D_h^{\pi,\pi',\rho}(s,a) + \rho \min_{\pi'_h} \mathbb{E}_{a\sim\pi'_h(\cdot|s)} \min_{\{\pi'_{h'}\}_{h'=h+1}^H} D_h^{\pi,\pi',\rho}(s,a) \\
=& (1-\rho)[\mathbb{D}_{\pi_h} Q_h^\pi](s) + \rho \min_{a\in\mathcal{A}} Q_h^\pi(s,a).
\end{aligned} \tag{E.4}$$

We set $\pi_h^-(s) = \arg\min_{a\in\mathcal{A}} Q_h^\pi(s,a) = \arg\min_{a\in\mathcal{A}} D_h^{\pi,\pi^-,\rho}(s,a)$.

At step $h$, we have

$$\begin{aligned}
V_h^\pi(s) \leq& C_h^{\pi,\pi^-,\rho}(s) \\
=& (1-\rho)[\mathbb{D}_{\pi_h} D_h^{\pi,\pi^-,\rho}](s) + \rho \min_{a\in\mathcal{A}} D_h^{\pi,\pi^-,\rho}(s,a) \\
=& (1-\rho)[\mathbb{D}_{\pi_h} Q_h^\pi](s) + \rho \min_{a\in\mathcal{A}} Q_h^\pi(s,a),
\end{aligned} \tag{E.5}$$

where the last equation comes from the robust Bellman equation at step $h+1$ and

$$D_h^{\pi,\pi^-,\rho}(s,a) = R_h(s,a) + [P_h C_{h+1}^{\pi,\pi^-,\rho}](s,a) = R_h(s,a) + [P_h V_{h+1}^\pi](s,a).$$

Thus, the robust Bellman equation holds at step $h$.

Then, we prove the commutability of the expectation and the minimization operations at step $h$. For any weighted function $w$, we have $\min_{\pi'} \sum_s w(s) C_h^{\pi,\pi',\rho}(s) \geq \sum_s w(s) \min_{\pi'} C_h^{\pi,\pi',\rho}(s)$. Then, $\min_{\pi'} \sum_s w(s) C_h^{\pi,\pi',\rho}(s) \leq \sum_s w(s) C_h^{\pi,\pi^-,\rho}(s) = \sum_s w(s) \min_{\pi'} C_h^{\pi,\pi',\rho}(s)$.

By induction on $h = H, \cdots, 1$, we prove the robust Bellman equation.

**Perfect Duality and Robust Bellman Optimality Equation** We now prove that the perfect duality holds and can be solved by the optimal robust Bellman equation.

The control problem in the LHS of (6.4) is equivalent to

$$\max_{\pi} \min_{\widetilde{\pi} \in \Pi^{\rho}(\pi)} \mathbb{E}\left[\sum_{h'=h}^{H} R_{h'}(s_{h'}, a_{h'}) | s_h = s, a_{h'} \sim \widetilde{\pi}_{h'}(\cdot | s_{h'})\right] = \max_{\pi} \min_{\pi'} C_h^{\pi,\pi',\rho}(s). \tag{E.6}$$

The control problem in the RHS of (6.4) is equivalent to

$$\min_{\widetilde{\pi} \in \Pi^{\rho}(\pi)} \max_{\pi} \mathbb{E}\left[\sum_{h'=h}^{H} R_{h'}(s_{h'}, a_{h'}) | s_h = s, a_{h'} \sim \widetilde{\pi}_{h'}(\cdot | s_{h'})\right] = \min_{\pi'} \max_{\pi} C_h^{\pi,\pi',\rho}(s). \tag{E.7}$$

For step $H$, we have $C_H^{\pi,\pi',\rho}(s) = [\mathbb{D}_{((1-\rho)\pi+\rho\pi')_H} R_H](s) = (1-\rho)[\mathbb{D}_{\pi_H} R_H](s) + \rho[\mathbb{D}_{\pi'_H} R_H](s)$. Thus, we have

$$\begin{aligned} \max_{\pi} \min_{\pi'} C_H^{\pi,\pi',\rho}(s) =& (1-\rho) \max_{\pi} [\mathbb{D}_{\pi_H} R_H](s) + \rho \min_{\pi'} [\mathbb{D}_{\pi'_H} R_H](s) \\ =& (1-\rho) \max_{a \in \mathcal{A}} R_H(s,a) + \rho \min_{b \in \mathcal{A}} R_H(s,b), \end{aligned} \tag{E.8}$$

and

$$\begin{aligned} \min_{\pi'} \max_{\pi} C_H^{\pi,\pi',\rho}(s) =& (1-\rho) \max_{\pi} [\mathbb{D}_{\pi_H} R_H](s) + \rho \min_{\pi'} [\mathbb{D}_{\pi'_H} R_H](s) \\ =& (1-\rho) \max_{a \in \mathcal{A}} R_H(s,a) + \rho \min_{b \in \mathcal{A}} R_H(s,b). \end{aligned} \tag{E.9}$$

At step $H$, the perfect duality holds for all $s$ and there always exists an optimal robust policy $\pi_H^*(s) = \arg\max_{a \in \mathcal{A}} Q_H^*(s,a) = \arg\max_{a \in \mathcal{A}} R_H(s,a)$ and its corresponding optimal adversarial policy $\pi_H^-(s) = \arg\min_{a \in \mathcal{A}} R_H(s,a)$ which are deterministic. The action robust Bellman optimality equation holds at step $H$ for any stats $s$ and action $a$.

In addition, $\max_{\pi} \min_{\pi'} \sum_s w(s) C_H^{\pi,\pi',\rho}(s) = \sum_s w(s) \max_{\pi} \min_{\pi'} C_H^{\pi,\pi',\rho}(s)$ for any weighted function $w : \mathcal{S} \to \Delta_{\mathcal{S}}$. This can be shown as

$$\begin{aligned}
&\max_{\pi}\min_{\pi'}\sum_{s\in\mathcal{S}} w(s)C_H^{\pi,\pi',\rho}(s) \\
=&(1-\rho)\max_{\pi}\sum_{s\in\mathcal{S}} w(s)[\mathbb{D}_{\pi_H}R_H](s) + \rho\min_{\pi'}\sum_{s\in\mathcal{S}} w(s)[\mathbb{D}_{\pi'_H}R_H](s) \\
=&(1-\rho)\sum_{s\in\mathcal{S}} w(s)\max_{a\in\mathcal{A}} R_H(s,a) + \rho\sum_{s\in\mathcal{S}} w(s)\min_{b\in\mathcal{A}} R_H(s,b).
\end{aligned} \tag{E.10}$$

Suppose that at steps from $h+1$ to $H$, the perfect duality holds for any $s$, the action robust Bellman optimality equation holds for any state $s$ and action $a$, there always exists an optimal robust policy $\pi^*_{h'} = \arg\max_{a\in\mathcal{A}} Q^*_{h'}(s,a)$ and its corresponding optimal adversarial policy $\pi^-_{h'}(s) = \arg\min_{a\in\mathcal{A}} Q^*_{h'}(s,a)$, $\forall h' \geq h+1$, which is deterministic, and $\max_\pi \min_{\pi'} \sum_s w(s) C_{h'}^{\pi,\pi',\rho}(s) = \sum_s w(s)\max_\pi \min_{\pi'} C_{h'}^{\pi,\pi',\rho}(s)$ for any state $s$, any weighted function $w : \mathcal{S} \to \Delta_{\mathcal{S}}$ and any $h' \geq h+1$. We have $V^*_{h'}(s) = V^{\pi^*}_{h'}(s) = C_{h'}^{\pi^*,\pi^-,\rho}(s)$ and $Q^*_{h'}(s,a) = Q^{\pi^*}_{h'}(s,a) = D_{h'}^{\pi^*,\pi^-,\rho}(s,a)$ for any state $s$ and any $h' \geq h+1$.

We first prove that the robust Bellman optimality equation holds at step $h$.

We have

$$\begin{aligned}
Q^*_h(s,a) &= \max_{\pi}\min_{\pi'} D_h^{\pi,\pi',\rho}(s,a) \\
&= \max_{\pi}\min_{\pi'}(R_h(s,a) + [P_h C_{h+1}^{\pi,\pi',\rho}](s,a)) \\
&= R_h(s,a) + [P_h(\max_{\pi}\min_{\pi'} C_{h+1}^{\pi,\pi',\rho})](s,a) \\
&= R_h(s,a) + [P_h V^*_{h+1}](s,a).
\end{aligned} \tag{E.11}$$

and also $Q^*_h(s,a) = Q^{\pi^*}_h(s,a) = D_h^{\pi^*,\pi^-,\rho}(s,a)$.

From the robust Bellman equation, we have

$$\begin{aligned}\max_{\pi} V_h^{\pi}(s) =& \max_{\pi}\left((1-\rho)[\mathbb{D}_{\pi_h} Q_h^{\pi}](s) + \rho \min_{a\in\mathcal{A}} Q_h^{\pi}(s,a)\right)\\ \leq& (1-\rho)\max_{\pi_h}\max_{\{\pi_h\}_{h'=h+1}^{H}}[\mathbb{D}_{\pi_h} Q_h^{\pi}](s) + \rho \max_{\{\pi_h\}_{h'=h+1}^{H}}\min_{a\in\mathcal{A}} Q_h^{\pi}(s,a)\\ \leq& (1-\rho)\max_{\pi_h}\max_{\{\pi_h\}_{h'=h+1}^{H}}[\mathbb{D}_{\pi_h} Q_h^{\pi}](s) + \rho \min_{a\in\mathcal{A}}\max_{\{\pi_h\}_{h'=h+1}^{H}} Q_h^{\pi}(s,a)\\ \leq& (1-\rho)\max_{\pi_h}[\mathbb{D}_{\pi_h} Q_h^{*}](s) + \rho \min_{a\in\mathcal{A}} Q_h^{*}(s,a)\\ =& (1-\rho)\max_{a\in\mathcal{A}} Q_h^{*}(s,a) + \rho \min_{a\in\mathcal{A}} Q_h^{*}(s,a).\end{aligned} \tag{E.12}$$

We set $\pi_h^*(s) = \max_{a\in\mathcal{A}} Q_h^*(s,a)$. According to the robust bellman equation, we have

$$\begin{aligned}\max_{\pi} V_h^{\pi}(s) \geq V_h^{\pi^*}(s) &= (1-\rho)[\mathbb{D}_{\pi_h^*} Q_h^{\pi^*}](s) + \rho \min_{a\in\mathcal{A}} Q_h^{\pi^*}(s,a)\\ &= (1-\rho)\max_{a\in\mathcal{A}} Q_h^{\pi^*}(s,a) + \rho \min_{a\in\mathcal{A}} Q_h^{\pi^*}(s,a)\\ &= (1-\rho)\max_{a\in\mathcal{A}} Q_h^{*}(s,a) + \rho \min_{a\in\mathcal{A}} Q_h^{*}(s,a).\end{aligned} \tag{E.13}$$

Thus, the robust Bellman optimality equation holds at step $h$. There always exists an optimal robust policy $\pi_h^* = \arg\max_{a\in\mathcal{A}} Q_h^*(s,a)$ and its corresponding optimal adversarial policy $\pi_h^-(s) = \arg\min_{a\in\mathcal{A}} Q_h^*(s,a)$ that is deterministic so that $C_h^{\pi^*,\pi^-,\rho}(s) = V_h^*(s)$.

Then, we prove the commutability of the expectation, the minimization and the maximization operations at step $h$.

In the proof of robust Bellman equation, we have shown that

$$\min_{\pi'}\sum_{s} w(s) C_h^{\pi,\pi',\rho}(s) = \sum_{s} w(s)\min_{\pi'} C_h^{\pi,\pi',\rho}(s)$$

for any policy $\pi$ and any weighted function $w$. Hence

$$\max_{\pi}\min_{\pi'}\sum_{s} w(s) C_h^{\pi,\pi',\rho}(s)\sum_{s} = \max_{\pi}\sum_{s} w(s)\min_{\pi'} C_h^{\pi,\pi',\rho}(s).$$

First, we have

$$\max_{\pi}\sum_{s} w(s)\min_{\pi'} C_h^{\pi,\pi',\rho}(s) \leq \sum_{s} w(s)\max_{\pi}\min_{\pi'} C_h^{\pi,\pi',\rho}(s).$$

Then, we can show

$$\begin{aligned}\max_{\pi}\sum_{s} w(s)\min_{\pi'} C_h^{\pi,\pi',\rho}(s) &\geq \sum_{s} w(s)\min_{\pi'} C_h^{\pi^*,\pi',\rho}(s)\\ &= \sum_{s} w(s) C_h^{\pi^*,\pi^-,\rho}(s)\\ &= \sum_{s} w(s)\max_{\pi}\min_{\pi'} C_h^{\pi,\pi',\rho}(s). \qquad \text{(E.14)}\end{aligned}$$

In summary,

$$\max_{\pi}\min_{\pi'}\sum_{s} w(s) C_h^{\pi,\pi',\rho}(s)\sum_{s} = w(s)\max_{\pi}\min_{\pi'} C_h^{\pi,\pi',\rho}(s).$$

We can show the perfect duality at step $h$ by

$$\max_{\pi}\min_{\pi'} C_h^{\pi,\pi',\rho}(s) = C_h^{\pi^*,\pi^-,\rho}(s) = \max_{\pi} C_h^{\pi,\pi^-,\rho}(s) \geq \min_{\pi'}\max_{\pi} C_h^{\pi,\pi',\rho}(s). \qquad \text{(E.15)}$$

By induction on $h = H, \cdots, 1$, we prove Proposition 4.

## E.2 Proof for Action Robust Reinforcement Learning with Certificates

In this section, we prove Theorem 20. Recall that we use $\overline{Q}_h^k$,$\overline{V}_h^k$,$\underline{Q}_h^k$,$\underline{V}_h^k$, $N_h^k$, $\hat{P}_h^k$,$\hat{r}_h^k$ and $\theta_h^k$ to denote the values of $\overline{Q}_h$,$\overline{V}_h$,$\underline{Q}_h$,$\underline{V}_h$, $\max\{N_h, 1\}$, $\hat{P}_h$, $r_h$ and $\theta_h$ at the beginning of the $k$-th episode in Algorithm 6.1.

### E.2.1 Proof sketch

In this section, we provide sketch of the proof, which will highlight our the main ideas of our proof. First, we will show that $\overline{V}_h(s) \geq V_h^*(s) \geq V_h^{\overline{\pi}}(s) \geq \underline{V}_h(s)$ hold for all $s$ and $a$. The regret can be bounded by $\overline{V}_1 - \underline{V}_1$ and then be divided by four items, each of which can be bounded separately. The full proof can be found in the appendix contained in the supplementary material.

**Proof sketch of monotonicity**

We define $\mathcal{E}^R$ to be the event where

$$\left|\hat{r}_h^k(s,a) - R_h(s,a)\right| \leq \sqrt{\frac{2\hat{r}_h^k(s,a)\iota}{N_h^k(s,a)}} + \frac{7\iota}{3(N_h^k(s,a))} \tag{E.16}$$

holds for all $(s,a,h,k) \in S \times A \times [H] \times [K]$. We also define $\mathcal{E}^{PV}$ to be the event where

$$\left|(\hat{P}_h^k - P_h)V_{h+1}^*(s,a)\right| \leq \sqrt{\frac{2\mathbb{V}_{\hat{P}_h^k}V_{h+1}^*(s,a)\iota}{N_h^k(s,a)}} + \frac{7H\iota}{3(N_h^k(s,a))} \tag{E.17}$$

and

$$\left|(\hat{P}_h^k - P_h)V_{h+1}^{\overline{\pi}^k}(s,a)\right| \leq \sqrt{\frac{2\mathbb{V}_{\hat{P}_h^k}V_{h+1}^{\overline{\pi}^k}(s,a)\iota}{N_h^k(s,a)}} + \frac{7H\iota}{3N_h^k(s,a)} \tag{E.18}$$

hold for all $(s,a,h,k) \in S \times A \times [H] \times [K]$.

Event $\mathcal{E}^R$ means that the estimations of all reward functions stay in certain neighborhood of the true values. Event $\mathcal{E}^{PV}$ represents that the estimation of the value functions at the next step stay in some intervals. The following lemma shows $\mathcal{E}^R$ and $\mathcal{E}^{PV}$ hold with high probability. The analysis will be done assuming the successful event $\mathcal{E}^R \cap \mathcal{E}^{PV}$ holds in the rest of this section.

**Lemma 16.** $\mathbb{P}(\mathcal{E}^R \cap \mathcal{E}^{PV}) \geq 1 - 3\delta$.

**Lemma 17.** Conditioned on event $\mathcal{E}^R \cap \mathcal{E}^{PV}$, $\overline{V}_h^k(s) \geq V_h^*(s) \geq V_h^{\overline{\pi}^k}(s) \geq \underline{V}_h^k(s)$ and $\overline{Q}_h^k(s,a) \geq Q_h^*(s,a) \geq Q_h^{\overline{\pi}^k}(s,a) \geq \underline{Q}_h^k(s,a)$ hold for all $(s,a,h,k) \in S \times A \times [H] \times [K]$.

**Regret analysis**

We decompose the regret and analyze the different terms. Set $\Theta_h^k(s,a) = \sqrt{\frac{8\mathbb{V}_{P_h} C_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}(s,a)\iota}{N_h^k(s,a)}} + \sqrt{\frac{32}{N_h^k(s,a)}} + \frac{46\sqrt{SH^4\iota}}{N_h^k(s,a)}$, where $\pi^{k*}$ is the optimal policy towards the adversary policy $\underline{\pi}^k$ with $\pi_h^{k*}(s) = \arg\max_\pi C_h^{\pi,\underline{\pi}^k,\rho}(s)$.

We set

$$M_1 = \sum_{k=1}^{K}\sum_{h=1}^{H}[\mathbb{D}_{\widetilde{\pi}_h^k}\hat{P}_h^k(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_h^k) - \hat{P}_h^k(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_h^k, a_h^k)], \tag{E.19}$$

$$M_2 = \sum_{k=1}^{K}\sum_{h=1}^{H}\frac{1}{H}[\mathbb{D}_{\widetilde{\pi}_h^k}P_h(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_h^k) - P_h(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_h^k, a_h^k)] \tag{E.20}$$

$$M_3 = \sum_{k=1}^{K}\sum_{h=1}^{H}[P_h^k(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_h^k, a_h^k) - (\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_{h+1}^k)] \tag{E.21}$$

$$M_4 = \sum_{k=1}^{K}\sum_{h=1}^{H}\left[\frac{(SH + SH^2)\iota}{N_h^k(s_h^k, a_h^k)} + \mathbb{D}_{\widetilde{\pi}_h^k}\Theta_h^k(s_h^k)\right] \tag{E.22}$$

Here $M_1$ and $M_2$ are the cumulative sample error from the random choices of the adversarial policy or agent's policy. $M_3$ is the cumulative sample error from the randomness of Monte Carlo sampling of the next state. $M_4$ is the cumulative error from the bonus item $\theta$. Lemma 18 shows that the regret can be bounded by these four terms.

**Lemma 18.** With probability at least $1 - (S+5)\delta$,

$$Regret(K) \leq \sum_{k=1}^{K}(\overline{V}_1^k(s_1^k) - \underline{V}_1^k(s_1^k)) \leq 21(M_1 + M_2 + M_3 + M_4). \tag{E.23}$$

We now bound each of these four items separately.

**Lemma 19.** With probability at least $1-\delta$, $|M_1| \leq H\sqrt{2HK\iota}$.

**Lemma 20.** With probability at least $1-\delta$, $|M_2| \leq \sqrt{2HK\iota}$.

**Lemma 21.** With probability at least $1-\delta$, $|M_3| \leq H\sqrt{2HK\iota}$.

**Lemma 22.** With probability at least $1-2\delta$, $|M_4| \leq 2S^2AH^3\iota^2 + 8\sqrt{SAH^2K\iota} + 46S^{\frac{3}{2}}AH^3\iota^2 + \sqrt{24SAH^3K\iota} + 6\sqrt{SAH^5}\iota$.

**Putting all together.** By Lemmas 18, 19, 20, 21, and 22, we conclude that, with probability $1-(S+10)\delta$,

$$Regret(K) \leq O(\sqrt{SAH^3K\iota} + S^2AH^3\iota^2). \tag{E.24}$$

By rescaling $\delta$, $\log(\frac{2SAHK}{\delta/(S+10)}) \leq c\iota$ for some constant $c$ and we finish the proof of regret. As $\sum_{k=1}^{K}(\overline{V}_1^k(s_1^k) - \underline{V}_1^k(s_1^k)) \leq O(\sqrt{SAH^3K\iota} + S^2AH^3\iota^2)$, we have that $V_1^*(s_1) - V_1^{\pi^{out}}(s_1) \leq \min_k \overline{V}_1^k(s_1^k) - \underline{V}_1^k(s_1^k) \leq O(\frac{\sqrt{SAH^3}\iota}{K} + \frac{S^2AH^3\iota^2}{K})$ and we finish the proof of sample complexity.

### E.2.2 Proof of monotonicity

**Proof of Lemma 16**

When $N_h^k(s,a) \leq 1$, (E.17), (E.18) and (E.16) hold trivially by the bound of the rewards and value functions.

For every $h \in [H]$ the empiric Bernstein inequality combined with a union bound argument, to take into account that $N_h^k(s,a) > 1$ is a random number, leads to the following inequality w.p. $1 - SAH\delta$ (see Theorem 4 in [73])

$$\left|(\hat{P}_h^k - P_h)V_{h+1}^*(s,a)\right| \leq \sqrt{\frac{2\mathbb{V}_{\hat{P}_h^k}V_{h+1}^*(s,a)\iota}{N_h^k(s,a)}} + \frac{7H\iota}{3(N_h^k(s,a))}, \tag{E.25}$$

and

$$\left|(\hat{P}_h^k - P_h)V_{h+1}^{\overline{\pi}^k}(s,a)\right| \leq \sqrt{\frac{2\mathbb{V}_{\hat{P}_h^k}V_{h+1}^{\overline{\pi}^k}(s,a)\iota}{N_h^k(s,a)}} + \frac{7H\iota}{3(N_h^k(s,a))}. \tag{E.26}$$

Similarly, with Azuma's inequality, w.p. $1 - SAH\delta$

$$\left|\hat{r}_h^k(s,a) - R_h(s,a)\right| \leq \sqrt{\frac{2Var(r_h^k(s,a))\iota}{N_h^k(s,a)}} + \frac{7\iota}{3(N_h^k(s,a))} \leq \sqrt{\frac{2\hat{r}_h^k(s,a)\iota}{N_h^k(s,a)}} + \frac{7\iota}{3(N_h^k(s,a))}, \quad \text{(E.27)}$$

where $Var(r_h^k(s,a))$ is the empirical variance of $R_h(s,a)$ computed by the $N_h^k(s,a)$ samples and $Var(r_h^k(s,a)) \leq \hat{r}_h^k(s,a)$ .

**Proof of Lemma 17**

We first prove that $\overline{Q}_h^k(s,a) \geq Q_h^*(s,a)$ for all $(s,a,h,k) \in S \times A \times [H] \times [K]$, by backward induction conditioned on the event $\mathcal{E}^R \cap \mathcal{E}^{PV}$. Firstly, the conclusion holds for $h = H+1$ because $\overline{V}_{H+1}(s) = \underline{V}_{H+1}(s) = 0$ and $\overline{Q}_{H+1}(s,a) = \underline{Q}_{H+1}(s,a) = 0$ for all $s$ and $a$. For $h \in [H]$, assuming the conclusion holds for $h+1$, by Algorithm 6.1, we have

$$\begin{aligned}
&\hat{r}_h^k(s,a) + \hat{P}_h^k\overline{V}_{h+1}(s,a) + \theta_h^k(s,a) - Q_h^*(s,a)\\
=&\hat{r}_h^k(s,a) + \hat{P}_h^k\overline{V}_{h+1}(s,a) + \theta_h^k(s,a) - R_h(s,a) - P_hV_{h+1}^*(s,a)\\
=&\hat{r}_h^k(s,a) - R_h(s,a) + \hat{P}_h^k\left(\overline{V}_{h+1} - V_{h+1}^*\right)(s,a) + (\hat{P}_h^k - P_h)V_{h+1}^*(s,a) + \theta_h^k(s,a)\\
\geq&(\hat{P}_h^k - P_h)V_{h+1}^*(s,a) + \sqrt{\frac{2\mathbb{V}_{\hat{P}_h^k}[(\overline{V}_{h+1}^k + \underline{V}_{h+1}^k)/2](s,a)\iota}{N_h^k(s,a)}} + \frac{\hat{P}_h^k\left(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k\right)(s,a)}{H} + \frac{8H^2\iota}{N_h^k(s,a)}\\
\geq&\sqrt{\frac{2\mathbb{V}_{\hat{P}_h^k}[(\overline{V}_{h+1}^k + \underline{V}_{h+1}^k)/2](s,a)\iota}{N_h^k(s,a)}} + \frac{\hat{P}_h^k\left(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k\right)(s,a)}{H} + \frac{8H^2\iota}{N_h^k(s,a)} - \sqrt{\frac{2\mathbb{V}_{\hat{P}_h^k}V_{h+1}^*(s,a)\iota}{N_h^k(s,a)}},
\end{aligned}$$

(E.28)

where the first inequality comes from event $\mathcal{E}^R$, $\overline{V}_{h+1}(s) \geq V^*_{h+1}(s)$ and the definition of $\theta^k_h(s,a)$ and the last inequality from event $\mathcal{E}^{PV}$. By the relation of $V$-values in the step $(h+1)$,

$$\begin{aligned}
&\left|\mathbb{V}_{\hat{P}^k_h}\left(\frac{\overline{V}^k_{h+1}+\underline{V}^k_{h+1}}{2}\right)(s,a) - \mathbb{V}_{\hat{P}^k_h}V^*_{h+1}(s,a)\right| \\
\leq &\left|[\hat{P}^k_h(\overline{V}^k_{h+1}+\underline{V}^k_{h+1})/2]^2 - (\hat{P}^k_h V^*_{h+1})^2\right|(s,a) + \left|\hat{P}^k_h[(\overline{V}^k_{h+1}+\underline{V}^k_{h+1})/2]^2 - \hat{P}^k_h(V^*_{h+1})^2\right|(s,a) \\
\leq &4H\hat{P}^k_h\left|(\overline{V}^k_{h+1}+\underline{V}^k_{h+1})/2 - V^*_{h+1}\right|(s,a) \\
\leq &2H\hat{P}^k_h\left(\overline{V}^k_{h+1}-\underline{V}^k_{h+1}\right)(s,a)
\end{aligned} \tag{E.29}$$

and

$$\begin{aligned}
&\sqrt{\frac{2\mathbb{V}_{\hat{P}^k_h}V^*_{h+1}(s,a)\iota}{N^k_h(s,a)}} \\
\leq &\sqrt{\frac{2\mathbb{V}_{\hat{P}^k_h}[(\overline{V}^k_{h+1}+\underline{V}^k_{h+1})/2](s,a)\iota + 4H\hat{P}^k_h\left(\overline{V}^k_{h+1}-\underline{V}^k_{h+1}\right)(s,a)\iota}{N^k_h(s,a)}} \\
\leq &\sqrt{\frac{2\mathbb{V}_{\hat{P}^k_h}[(\overline{V}^k_{h+1}+\underline{V}^k_{h+1})/2](s,a)\iota}{N^k_h(s,a)}} + \sqrt{\frac{4H\hat{P}^k_h\left(\overline{V}^k_{h+1}-\underline{V}^k_{h+1}\right)(s,a)\iota}{N^k_h(s,a)}} \\
\leq &\sqrt{\frac{2\mathbb{V}_{\hat{P}^k_h}[(\overline{V}^k_{h+1}+\underline{V}^k_{h+1})/2](s,a)\iota}{N^k_h(s,a)}} + \frac{\hat{P}^k_h\left(\overline{V}^k_{h+1}-\underline{V}^k_{h+1}\right)(s,a)}{H} + \frac{8H^2\iota}{N^k_h(s,a)}.
\end{aligned} \tag{E.30}$$

Plugging (E.30) back into (E.28), we have $\hat{r}^k_h(s,a) + \hat{P}^k_h\overline{V}_{h+1}(s,a) + \theta^k_h(s,a) \geq Q^*_h(s,a)$. Thus, $\overline{Q}^k_h(s,a) = \min\{H-h+1, \hat{r}^k_h(s,a) + \hat{P}^k_h\overline{V}^k_{h+1}(s,a) + \theta^k_h(s,a)\} \geq Q^*_h(s,a)$.

From the definition of $\overline{V}^k_h(s)$ and $\overline{\pi}^k_h$, we have

$$\begin{aligned}
\overline{V}^k_h(s) =&(1-\rho)\overline{Q}^k_h(s,\overline{\pi}^k_h(s)) + \rho\overline{Q}^k_h(s,\underline{\pi}^k_h(s)) \\
\geq&(1-\rho)\overline{Q}^k_h(s,\pi^*_h(s)) + \rho Q^*_h(s,\underline{\pi}^k_h(s)) \\
\geq&(1-\rho)Q^*_h(s,\pi^*_h(s)) + \rho\min_{a\in\mathcal{A}}Q^*_h(s,a) = V^*_h(s).
\end{aligned} \tag{E.31}$$

Similarly, we can prove that $\underline{Q}_h^k(s,a) \le Q_h^{\overline{\pi}^k}(s,a)$ and $\underline{V}_h^k(s) \le V_h^{\overline{\pi}^k}(s)$.

$$
\begin{aligned}
&\hat{r}_h^k(s,a) + \hat{P}_h^k \underline{V}_{h+1}(s,a) - \theta_h^k(s,a) - Q_h^{\overline{\pi}^k}(s,a) \\
=&\hat{r}_h^k(s,a) + \hat{P}_h^k \underline{V}_{h+1}(s,a) - \theta_h^k(s,a) - R_h(s,a) - P_h V_{h+1}^{\overline{\pi}^k}(s,a) \\
=&\hat{r}_h^k(s,a) - R_h(s,a) + \hat{P}_h^k \left( \underline{V}_{h+1} - V_{h+1}^{\overline{\pi}^k} \right)(s,a) + (\hat{P}_h^k - P_h) V_{h+1}^{\overline{\pi}^k}(s,a) - \theta_h^k(s,a) \\
\le&(\hat{P}_h^k - P_h) V_{h+1}^{\overline{\pi}^k}(s,a) - \sqrt{\frac{2\mathbb{V}_{\hat{P}_h^k}[(\overline{V}_{h+1}^k + \underline{V}_{h+1}^k)/2](s,a)\iota}{N_h^k(s,a)}} \\
&- \frac{\hat{P}_h^k \left( \overline{V}_{h+1}^k - \underline{V}_{h+1}^k \right)(s,a)}{H} - \frac{8H^2\iota}{N_h^k(s,a)} \\
\le&\sqrt{\frac{2\mathbb{V}_{\hat{P}_h^k} V_{h+1}^{\overline{\pi}^k}(s,a)\iota}{N_h^k(s,a)}} - \sqrt{\frac{2\mathbb{V}_{\hat{P}_h^k}[(\overline{V}_{h+1}^k + \underline{V}_{h+1}^k)/2](s,a)\iota}{N_h^k(s,a)}} \\
&- \frac{\hat{P}_h^k \left( \overline{V}_{h+1}^k - \underline{V}_{h+1}^k \right)(s,a)}{H} - \frac{8H^2\iota}{N_h^k(s,a)} \le 0,
\end{aligned}
\tag{E.32}
$$

and

$$
\begin{aligned}
\underline{V}_h^k(s) =&(1-\rho)\underline{Q}_h^k(s, \overline{\pi}_h^k(s)) + \rho \underline{Q}_h^k(s, \underline{\pi}_h^k(s)) \\
\le&(1-\rho) Q_h^{\overline{\pi}^k}(s, \overline{\pi}_h^k(s)) + \rho \min_{a \in \mathcal{A}} \underline{Q}_h^k(s,a) \\
\le&(1-\rho) Q_h^{\overline{\pi}^k}(s, \overline{\pi}_h^k(s)) + \rho \underline{Q}_h^k(s, \arg\min_{a \in \mathcal{A}} Q_h^{\overline{\pi}^k}(s,a)) \\
\le&(1-\rho) Q_h^{\overline{\pi}^k}(s, \overline{\pi}_h^k(s)) + \rho \min_{a \in \mathcal{A}} Q_h^{\overline{\pi}^k}(s,a) = V_h^{\overline{\pi}^k}(s).
\end{aligned}
\tag{E.33}
$$

### E.2.3 Regret Analysis

**Proof of Lemma 18**

We consider the event $\mathcal{E}^R \cap \mathcal{E}^{PV}$. The following analysis will be done assuming the successful event $\mathcal{E}^R \cap \mathcal{E}^{PV}$ holds. By Lemma 17, the regret can be bounded by $Regret(K) := \sum_{k=1}^K (V_1^*(s_1^k) - V_1^{\overline{\pi}^k}(s_1^k)) \le \sum_{k=1}^K (\overline{V}_1^k(s_1^k) - \underline{V}_1^k(s_1^k))$.

By the update steps in Algorithm 6.1, we have

$$
\begin{aligned}
&\overline{V}_h^k(s_h^k) - \underline{V}_h^k(s_h^k)\\
=&(1-\rho)\overline{Q}_h^k(s_h^k, \overline{\pi}_h^k(s_h^k)) + \rho\overline{Q}_h^k(s_h^k, \underline{\pi}_h^k(s_h^k)) - (1-\rho)\underline{Q}_h^k(s_h^k, \overline{\pi}_h^k(s_h^k)) - \rho\underline{Q}_h^k(s_h^k, \underline{\pi}_h^k(s_h^k))\\
\leq&[\mathbb{D}_{\widetilde{\pi}_h^k}\hat{P}_h^k(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)](s_h^k) + 2\mathbb{D}_{\widetilde{\pi}_h^k}\theta_h(s_h^k)\\
=&[\mathbb{D}_{\widetilde{\pi}_h^k}\hat{P}_h^k(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)](s_h^k) - [\hat{P}_h^k(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)](s_h^k, a_h^k) + 2\mathbb{D}_{\widetilde{\pi}_h^k}\theta_h(s_h^k) + [\hat{P}_h^k(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)](s_h^k, a_h^k)\\
=&[\mathbb{D}_{\widetilde{\pi}_h^k}\hat{P}_h^k(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)](s_h^k) - [\hat{P}_h^k(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)](s_h^k, a_h^k) + 2\mathbb{D}_{\widetilde{\pi}_h^k}\theta_h(s_h^k)\\
&+ [\hat{P}_h^k(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)](s_h^k, a_h^k) - c_1 P_h(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_h^k, a_h^k)\\
&+ c_1 P_h(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_h^k, a_h^k) - c_2(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_{h+1}^k) + c_2(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_{h+1}^k)\\
=&[\mathbb{D}_{\widetilde{\pi}_h^k}\hat{P}_h^k(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)](s_h^k) - [\hat{P}_h^k(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)](s_h^k, a_h^k)\\
&+ [\hat{P}_h^k(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)](s_h^k, a_h^k) - c_1 P_h(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_h^k, a_h^k)\\
&+ c_1 P_h(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_h^k, a_h^k) - c_2(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_{h+1}^k) + c_2(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_{h+1}^k)\\
&+ 2(1-\rho)\sqrt{\frac{2\mathbb{V}_{\hat{P}_h^k}[(\overline{V}_{h+1}^k + \underline{V}_{h+1}^k)/2](s_h^k, \overline{\pi}_h^k(s_h^k))\iota}{N_h^k(s_h^k, \overline{\pi}_h^k(s_h^k))}} + 2(1-\rho)\sqrt{\frac{2\hat{r}_h^k(s_h^k, \overline{\pi}_h^k(s_h^k))\iota}{N_h^k(s_h^k, \overline{\pi}_h^k(s_h^k))}}\\
&+ (1-\rho)\hat{P}_h^k(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_h^k, \overline{\pi}_h^k(s_h^k))/H + \frac{2(1-\rho)(24H^2 + 7H + 7)\iota}{3N_h^k(s_h^k, \overline{\pi}_h^k(s_h^k)))}\\
&+ 2\rho\sqrt{\frac{2\mathbb{V}_{\hat{P}_h^k}[(\overline{V}_{h+1}^k + \underline{V}_{h+1}^k)/2](s_h^k, \underline{\pi}_h^k(s_h^k))\iota}{N_h^k(s_h^k, \underline{\pi}_h^k(s_h^k))}} + 2\rho\sqrt{\frac{2\hat{r}_h^k(s_h^k, \underline{\pi}_h^k(s_h^k))\iota}{N_h^k(s_h^k, \underline{\pi}_h^k(s_h^k))}}\\
&+ \rho\hat{P}_h^k(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_h^k, \underline{\pi}_h^k(s_h^k))/H + \frac{2\rho(24H^2 + 7H + 7)\iota}{3N_h^k(s_h^k, \underline{\pi}_h^k(s_h^k)))}\\
=&(1+1/H)[\mathbb{D}_{\widetilde{\pi}_h^k}\hat{P}_h^k(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)](s_h^k) - (1+1/H)[\hat{P}_h^k(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)](s_h^k, a_h^k)\\
&+ \underbrace{(1+1/H)[\hat{P}_h^k(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)](s_h^k, a_h^k) - c_1 P_h(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_h^k, a_h^k)}_{(a)}\\
&+ c_1 P_h(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_h^k, a_h^k) - c_2(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_{h+1}^k) + c_2(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_{h+1}^k)\\
&+ 2(1-\rho)\underbrace{\sqrt{\frac{2\mathbb{V}_{\hat{P}_h^k}[(\overline{V}_{h+1}^k + \underline{V}_{h+1}^k)/2](s_h^k, \overline{\pi}_h^k(s_h^k))\iota}{N_h^k(s_h^k, \overline{\pi}_h^k(s_h^k))}}}_{(b1)} + 2(1-\rho)\sqrt{\frac{2\hat{r}_h^k(s_h^k, \overline{\pi}_h^k(s_h^k))\iota}{N_h^k(s_h^k, \overline{\pi}_h^k(s_h^k))}}\\
&+ \frac{2(1-\rho)(24H^2 + 7H + 7)\iota}{3N_h^k(s_h^k, \overline{\pi}_h^k(s_h^k)))} + 2\rho\underbrace{\sqrt{\frac{2\mathbb{V}_{\hat{P}_h^k}[(\overline{V}_{h+1}^k + \underline{V}_{h+1}^k)/2](s_h^k, \underline{\pi}_h^k(s_h^k))\iota}{N_h^k(s_h^k, \underline{\pi}_h^k(s_h^k))}}}_{(b2)}\\
&+ 2\rho\sqrt{\frac{2\hat{r}_h^k(s_h^k, \underline{\pi}_h^k(s_h^k))\iota}{N_h^k(s_h^k, \underline{\pi}_h^k(s_h^k))}} + \frac{2\rho(24H^2 + 7H + 7)\iota}{3N_h^k(s_h^k, \underline{\pi}_h^k(s_h^k)))}.
\end{aligned}
\tag{E.34}
$$

**Bound of the error of the empirical probability estimator (a)** By Bennett's inequality, we have that w.p. $1 - S\delta$

$$|\hat{P}_h^k(s'|s,a) - P_h(s'|s,a)| \le \sqrt{\frac{2P_h(s'|s,a)\iota}{N_h^k(s,a)}} + \frac{\iota}{3N_h^k(s,a)} \tag{E.35}$$

holds for all $s, a, h, k, s'$.

Thus, we have that

$$\begin{aligned}
&(\hat{P}_h^k - P_h)(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s,a) \\
=&\sum_{s'}(\hat{P}_h^k(s'|s,a) - P_h(s'|s,a))(\overline{V}_{h+1}^k(s') - \underline{V}_{h+1}^k(s')) \\
\le&\sum_{s'}\sqrt{\frac{2P_h(s'|s,a)\iota}{N_h^k(s,a)}}(\overline{V}_{h+1}^k(s') - \underline{V}_{h+1}^k(s')) + \frac{SH\iota}{3N_h^k(s,a)} \\
\le&\sum_{s'}\left(\frac{P_h(s'|s,a)\iota}{H} + \frac{H}{2N_h^k(s,a)}\right)\left(\overline{V}_{h+1}^k(s') - \underline{V}_{h+1}^k(s')\right) + \frac{SH\iota}{3N_h^k(s,a)} \\
\le&P_h(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s,a)/H + \frac{SH^2}{2N_h^k(s,a)} + \frac{SH\iota}{3N_h^k(s,a)} \\
\le&P_h(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s,a)/H + \frac{SH^2\iota}{N_h^k(s,a)},
\end{aligned} \tag{E.36}$$

where the second inequality is due to AM-GM inequality.

**Bound of the error of the empirical variance estimator (b1) & (b2)** Here, we bound $\mathbb{V}_{\hat{P}_h^k}[(\overline{V}_{h+1}^k + \underline{V}_{h+1}^k)/2](s_h^k, a_h^k)$.

Recall that $C_h^{\pi,\pi',\rho}(s) = \mathbb{E}\left[\sum_{h'=h}^H R_{h'}(s_{h'}, a_{h'})|s_h = s, a_{h'} \sim \widetilde{\pi}_{h'}(\cdot|s_{h'})\right]$ in Appendix E.1. Set $\pi^{k*}$ here is the optimal policy towards the adversary policy $\underline{\pi}^k$ with $\pi_h^{k*}(s) = \arg\max_\pi C_h^{\pi,\underline{\pi}^k,\rho}(s)$. Similar to the proof in Appendix E.2.2, we can show that $\overline{V}_h^k(s) \ge C_h^{\pi^{k*},\underline{\pi}^k,\rho}(s)$. We also have that $C_h^{\pi^{k*},\underline{\pi}^k,\rho}(s) = \max_\pi C_h^{\pi,\underline{\pi}^k,\rho}(s) \ge C_h^{\overline{\pi}^k,\underline{\pi}^k,\rho}(s) \ge V_h^{\overline{\pi}^k}(s) \ge \underline{V}_h^k(s)$ . For any $(s, a, h, k) \in$

$\mathcal{S} \times \mathcal{A} \times [H] \times [K]$, under event $\mathcal{E}^R \cap \mathcal{E}^{PV}$,

$$\begin{aligned}
&\mathbb{V}_{\hat{P}_h^k}[(\overline{V}_{h+1}^k + \underline{V}_{h+1}^k)/2](s,a) - \mathbb{V}_{P_h} C_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}(s,a) \\
=&\hat{P}_h^k[(\overline{V}_{h+1}^k + \underline{V}_{h+1}^k)/2]^2(s,a) - [\hat{P}_h^k(\overline{V}_{h+1}^k + \underline{V}_{h+1}^k)/2]^2(s,a) \\
&- P_h(C_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho})^2(s,a) + (P_h C_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho})^2(s,a) \\
\le&[\hat{P}_h^k(\overline{V}_{h+1}^k)^2 - (\hat{P}_h^k \underline{V}_{h+1}^k)^2 - P_h(\underline{V}_{h+1}^k)^2 + (P_h \overline{V}_{h+1}^k)^2](s,a) \\
\le&|(\hat{P}_h^k - P_h)(\overline{V}_{h+1}^k)^2|(s,a) + |(P_h \underline{V}_{h+1}^k)^2 - (\hat{P}_h^k \underline{V}_{h+1}^k)^2|(s,a) \\
&+ P_h|(\overline{V}_{h+1}^k)^2 - (\underline{V}_{h+1}^k)^2|(s,a) + |(P_h \overline{V}_{h+1}^k)^2 - (P_h \underline{V}_{h+1}^k)^2|(s,a),
\end{aligned} \tag{E.37}$$

where the first inequality is due $\overline{V}_h^k(s) \ge C_h^{\pi^{k*},\underline{\pi}^k,\rho}(s) \ge \underline{V}_h^k(s)$. The result of [105] combined with a union bound on $N_h^k(s,a) \in [K]$ implies w.p $1-\delta$

$$\|\hat{P}_h^k(\cdot|s,a) - P_h(\cdot|s,a)\|_1 \le \sqrt{\frac{2S\iota}{N_h^k(s,a)}} \tag{E.38}$$

holds for all $s, a, h, k$.

These terms can be bounded separately by

$$\begin{aligned}
&|(\hat{P}_h^k - P_h)(\overline{V}_{h+1}^k)^2|(s,a) \le H^2 \sqrt{\frac{2S\iota}{N_h^k(s,a)}}, \\
&|(P_h \underline{V}_{h+1}^k)^2 - (\hat{P}_h^k \underline{V}_{h+1}^k)^2|(s,a) \le 2H|(P_h - \hat{P}_h^k)\underline{V}_{h+1}^k| \le 2H^2 \sqrt{\frac{2S\iota}{N_h^k(s,a)}}, \\
&P_h|(\overline{V}_{h+1}^k)^2 - (\underline{V}_{h+1}^k)^2|(s,a) \le 2HP_h(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s,a), \\
&|(P_h \overline{V}_{h+1}^k)^2 - (P_h \underline{V}_{h+1}^k)^2|(s,a) \le 2HP_h(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s,a),
\end{aligned} \tag{E.39}$$

where the first two inequality is due to (E.38). In addition, $3H^2\sqrt{\frac{2S\iota}{N_h^k(s,a)}} \le 1 + \frac{9SH^4\iota}{2N_h^k(s,a)}$. Thus, we have

$$
\begin{aligned}
&(1-\rho)\sqrt{\frac{\mathbb{V}_{\hat{P}_h^k}[(\overline{V}_{h+1}^k+\underline{V}_{h+1}^k)/2](s_h^k,\overline{\pi}_h^k(s_h^k))\iota}{N_h^k(s_h^k,\overline{\pi}_h^k(s_h^k))}}+\rho\sqrt{\frac{\mathbb{V}_{\hat{P}_h^k}[(\overline{V}_{h+1}^k+\underline{V}_{h+1}^k)/2](s_h^k,\underline{\pi}_h^k(s_h^k))\iota}{N_h^k(s_h^k,\underline{\pi}_h^k(s_h^k))}}\\
\le&(1-\rho)\sqrt{\frac{\mathbb{V}_{P_h}C_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}(s_h^k,\overline{\pi}_h^k(s_h^k))\iota}{N_h^k(s_h^k,\overline{\pi}_h^k(s_h^k))}}+\rho\sqrt{\frac{\mathbb{V}_{P_h}C_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}(s_h^k,\underline{\pi}_h^k(s_h^k))\iota}{N_h^k(s_h^k,\underline{\pi}_h^k(s_h^k))}}\\
&+(1-\rho)\sqrt{\frac{4HP_h(\overline{V}_{h+1}^k-\underline{V}_{h+1}^k)(s_h^k,\overline{\pi}_h^k(s_h^k))\iota}{N_h^k(s_h^k,\overline{\pi}_h^k(s_h^k))}}+\rho\sqrt{\frac{4HP_h(\overline{V}_{h+1}^k-\underline{V}_{h+1}^k)(s_h^k,\underline{\pi}_h^k(s_h^k))\iota}{N_h^k(s_h^k,\underline{\pi}_h^k(s_h^k))}}\\
&+(1-\rho)\sqrt{\frac{1}{N_h^k(s_h^k,\overline{\pi}_h^k(s_h^k))}}+\rho\sqrt{\frac{1}{N_h^k(s_h^k,\underline{\pi}_h^k(s_h^k))}}+\frac{(1-\rho)\sqrt{9SH^4\iota/2}}{N_h^k(s_h^k,\overline{\pi}_h^k(s_h^k))}+\frac{\rho\sqrt{9SH^4\iota/2}}{N_h^k(s_h^k,\underline{\pi}_h^k(s_h^k))}\\
\le&(1-\rho)\sqrt{\frac{\mathbb{V}_{P_h}C_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}(s_h^k,\overline{\pi}_h^k(s_h^k))\iota}{N_h^k(s_h^k,\overline{\pi}_h^k(s_h^k))}}+\rho\sqrt{\frac{\mathbb{V}_{P_h}C_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}(s_h^k,\underline{\pi}_h^k(s_h^k))\iota}{N_h^k(s_h^k,\underline{\pi}_h^k(s_h^k))}}\\
&+(1-\rho)\left(\frac{P_h(\overline{V}_{h+1}^k-\underline{V}_{h+1}^k)(s_h^k,\overline{\pi}_h^k(s_h^k))}{2\sqrt{2}H}+\frac{2\sqrt{2}H^2\iota}{N_h^k(s_h^k,\overline{\pi}_h^k(s_h^k))}\right)\\
&+\rho\left(\frac{P_h(\overline{V}_{h+1}^k-\underline{V}_{h+1}^k)(s_h^k,\underline{\pi}_h^k(s_h^k))}{2\sqrt{2}H}+\frac{2\sqrt{2}H^2\iota}{N_h^k(s_h^k,\underline{\pi}_h^k(s_h^k))}\right)\\
&+(1-\rho)\sqrt{\frac{1}{N_h^k(s_h^k,\overline{\pi}_h^k(s_h^k))}}+\rho\sqrt{\frac{1}{N_h^k(s_h^k,\underline{\pi}_h^k(s_h^k))}}\\
&+\frac{(1-\rho)\sqrt{9SH^4\iota/2}}{N_h^k(s_h^k,\overline{\pi}_h^k(s_h^k))}+\frac{\rho\sqrt{9SH^4\iota/2}}{N_h^k(s_h^k,\underline{\pi}_h^k(s_h^k))}\\
=&(1-\rho)\sqrt{\frac{\mathbb{V}_{P_h}C_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}(s_h^k,\overline{\pi}_h^k(s_h^k))\iota}{N_h^k(s_h^k,\overline{\pi}_h^k(s_h^k))}}+\rho\sqrt{\frac{\mathbb{V}_{P_h}C_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}(s_h^k,\underline{\pi}_h^k(s_h^k))\iota}{N_h^k(s_h^k,\underline{\pi}_h^k(s_h^k))}}\\
&+\frac{\mathbb{D}_{\widetilde{\pi}_h^k}P_h(\overline{V}_{h+1}^k-\underline{V}_{h+1}^k)(s_h^k)}{2\sqrt{2}H}+\frac{2\sqrt{2}(1-\rho)H^2\iota}{N_h^k(s_h^k,\overline{\pi}_h^k(s_h^k))}+\frac{2\sqrt{2}\rho H^2\iota}{N_h^k(s_h^k,\underline{\pi}_h^k(s_h^k))}\\
&+(1-\rho)\sqrt{\frac{1}{N_h^k(s_h^k,\overline{\pi}_h^k(s_h^k))}}+\rho\sqrt{\frac{1}{N_h^k(s_h^k,\underline{\pi}_h^k(s_h^k))}}\\
&+\frac{(1-\rho)\sqrt{9SH^4\iota/2}}{N_h^k(s_h^k,\overline{\pi}_h^k(s_h^k))}+\frac{\rho\sqrt{9SH^4\iota/2}}{N_h^k(s_h^k,\underline{\pi}_h^k(s_h^k))},
\end{aligned}
\tag{E.40}
$$

where the second inequality is due to AM-GM inequality.

**Recursing on $h$** Plugging (E.36) and (E.40) into (E.34)and setting $c_1 = 1 + 1/H$ and $c_2 = (1 + 1/H)^3$ , we have

$$
\begin{aligned}
&\overline{V}_h^k(s_h^k) - \underline{V}_h^k(s_h^k) \\
\leq &(1+1/H)[\mathbb{D}_{\widetilde{\pi}_h^k}\hat{P}_h^k(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)](s_h^k) - (1+1/H)[\hat{P}_h^k(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)](s_h^k, a_h^k) \\
&+ (1/H + 1/H^2)P_h(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_h^k, a_h^k) + \frac{(SH + SH^2)\iota}{N_h^k(s_h^k, a_h^k)} \\
&+ c_1 P_h(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_h^k, a_h^k) - c_2(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_{h+1}^k) + c_2(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_{h+1}^k) \\
&+ 2(1-\rho)\sqrt{\frac{2\hat{r}_h^k(s_h^k, \overline{\pi}_h^k(s_h^k))\iota}{N_h^k(s_h^k, \overline{\pi}_h^k(s_h^k))}} + \frac{2(1-\rho)(24H^2 + 7H + 7)\iota}{3N_h^k(s_h^k, \overline{\pi}_h^k(s_h^k)))} \\
&+ 2\rho\sqrt{\frac{2\hat{r}_h^k(s_h^k, \underline{\pi}_h^k(s_h^k))\iota}{N_h^k(s_h^k, \underline{\pi}_h^k(s_h^k))}} + \frac{2\rho(24H^2 + 7H + 7)\iota}{3N_h^k(s_h^k, \underline{\pi}_h^k(s_h^k)))} \\
&+ (1-\rho)\sqrt{\frac{8\mathbb{V}_{P_h} C_{h+1}^{\pi^{k*}, \underline{\pi}^k, \rho}(s_h^k, \overline{\pi}_h^k(s_h^k))\iota}{N_h^k(s_h^k, \overline{\pi}_h^k(s_h^k))}} + \rho\sqrt{\frac{8\mathbb{V}_{P_h} C_{h+1}^{\pi^{k*}, \underline{\pi}^k, \rho}(s_h^k, \underline{\pi}_h^k(s_h^k))\iota}{N_h^k(s_h^k, \underline{\pi}_h^k(s_h^k))}} \\
&+ \frac{\mathbb{D}_{\widetilde{\pi}_h^k} P_h(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_h^k)}{H} + \frac{8(1-\rho)H^2\iota}{N_h^k(s_h^k, \overline{\pi}_h^k(s_h^k))} + \frac{8\rho H^2\iota}{N_h^k(s_h^k, \underline{\pi}_h^k(s_h^k))} \\
&+ (1-\rho)\sqrt{\frac{8}{N_h^k(s_h^k, \overline{\pi}_h^k(s_h^k))}} + \rho\sqrt{\frac{8}{N_h^k(s_h^k, \underline{\pi}_h^k(s_h^k))}} + \frac{6(1-\rho)\sqrt{SH^4\iota}}{N_h^k(s_h^k, \overline{\pi}_h^k(s_h^k))} + \frac{6\rho\sqrt{SH^4\iota}}{N_h^k(s_h^k, \underline{\pi}_h^k(s_h^k))}.
\end{aligned}
\tag{E.41}
$$

We set $\Theta_h^k(s,a) = \sqrt{\frac{8\mathbb{V}_{P_h} C_{h+1}^{\pi^{k*}, \underline{\pi}^k, \rho}(s,a)\iota}{N_h^k(s,a)}} + \sqrt{\frac{32}{N_h^k(s,a)}} + \frac{46\sqrt{SH^4\iota}}{N_h^k(s,a)}$. Since $r_h^k(s,a) \leq 1$, by organizing the items, we have that

$$
\begin{aligned}
&\overline{V}^k_h(s^k_h) - \underline{V}^k_h(s^k_h) \\
\leq &(1+1/H)[\mathbb{D}_{\widetilde{\pi}^k_h}\hat{P}^k_h(\overline{V}^k_{h+1} - \underline{V}^k_{h+1})](s^k_h) - (1+1/H)[\hat{P}^k_h(\overline{V}^k_{h+1} - \underline{V}^k_{h+1})](s^k_h, a^k_h) \\
&+ (1/H + 1/H^2)P_h(\overline{V}^k_{h+1} - \underline{V}^k_{h+1})(s^k_h, a^k_h) + \frac{(SH + SH^2)\iota}{N^k_h(s^k_h, a^k_h)} \\
&+ c_1 P_h(\overline{V}^k_{h+1} - \underline{V}^k_{h+1})(s^k_h, a^k_h) - c_2(\overline{V}^k_{h+1} - \underline{V}^k_{h+1})(s^k_{h+1}) + c_2(\overline{V}^k_{h+1} - \underline{V}^k_{h+1})(s^k_{h+1}) \\
&+ \frac{\mathbb{D}_{\widetilde{\pi}^k_h} P_h(\overline{V}^k_{h+1} - \underline{V}^k_{h+1})(s^k_h, \overline{\pi}^k_h(s^k_h))}{H} + \mathbb{D}_{\widetilde{\pi}^k_h}\Theta^k_h(s^k_h) \\
\leq &(1+1/H)[\mathbb{D}_{\widetilde{\pi}^k_h}\hat{P}^k_h(\overline{V}^k_{h+1} - \underline{V}^k_{h+1})](s^k_h) - (1+1/H)[\hat{P}^k_h(\overline{V}^k_{h+1} - \underline{V}^k_{h+1})](s^k_h, a^k_h) \\
&+ \frac{1}{H}[\mathbb{D}_{\widetilde{\pi}^k_h} P_h(\overline{V}^k_{h+1} - \underline{V}^k_{h+1})(s^k_h) - P_h(\overline{V}^k_{h+1} - \underline{V}^k_{h+1})(s^k_h, a^k_h)] \qquad\qquad \text{(E.42)} \\
&+ (1 + 3/H + 1/H^2)P_h(\overline{V}^k_{h+1} - \underline{V}^k_{h+1})(s^k_h, a^k_h) - c_2(\overline{V}^k_{h+1} - \underline{V}^k_{h+1})(s^k_{h+1}) \\
&+ c_2(\overline{V}^k_{h+1} - \underline{V}^k_{h+1})(s^k_{h+1}) + \frac{(SH + SH^2)\iota}{N^k_h(s^k_h, a^k_h)} + \mathbb{D}_{\widetilde{\pi}^k_h}\Theta^k_h(s^k_h) \\
\leq &(1+1/H)[\mathbb{D}_{\widetilde{\pi}^k_h}\hat{P}^k_h(\overline{V}^k_{h+1} - \underline{V}^k_{h+1})](s^k_h) - (1+1/H)[\hat{P}^k_h(\overline{V}^k_{h+1} - \underline{V}^k_{h+1})](s^k_h, a^k_h) \\
&+ \frac{1}{H}[\mathbb{D}_{\widetilde{\pi}^k_h} P_h(\overline{V}^k_{h+1} - \underline{V}^k_{h+1})(s^k_h) - P_h(\overline{V}^k_{h+1} - \underline{V}^k_{h+1})(s^k_h, a^k_h)] \\
&+ c_2 P_h(\overline{V}^k_{h+1} - \underline{V}^k_{h+1})(s^k_h, a^k_h) - c_2(\overline{V}^k_{h+1} - \underline{V}^k_{h+1})(s^k_{h+1}) \\
&+ c_2(\overline{V}^k_{h+1} - \underline{V}^k_{h+1})(s^k_{h+1}) + \frac{(SH + SH^2)\iota}{N^k_h(s^k_h, a^k_h)} + \mathbb{D}_{\widetilde{\pi}^k_h}\Theta^k_h(s^k_h).
\end{aligned}
$$

By induction of (E.34) on $h = 1, \cdots, H$ and $\overline{V}^k_{h+1} = \underline{V}^k_{h+1} = 0$, we have that

$$
\begin{aligned}
Regret(K) \leq 21 \sum_{k=1}^{K}\sum_{h=1}^{H} &(\mathbb{D}_{\widetilde{\pi}^k_h}\hat{P}^k_h(\overline{V}^k_{h+1} - \underline{V}^k_{h+1})(s^k_h) - \hat{P}^k_h(\overline{V}^k_{h+1} - \underline{V}^k_{h+1})(s^k_h, a^k_h) \\
&+ \frac{1}{H}[\mathbb{D}_{\widetilde{\pi}^k_h} P_h(\overline{V}^k_{h+1} - \underline{V}^k_{h+1})(s^k_h) - P_h(\overline{V}^k_{h+1} - \underline{V}^k_{h+1})(s^k_h, a^k_h)] \\
&+ P_h(\overline{V}^k_{h+1} - \underline{V}^k_{h+1})(s^k_h, a^k_h) - (\overline{V}^k_{h+1} - \underline{V}^k_{h+1})(s^k_{h+1}) \\
&+ \frac{(SH + SH^2)\iota}{N^k_h(s^k_h, a^k_h)} + \mathbb{D}_{\widetilde{\pi}^k_h}\Theta^k_h(s^k_h)).
\end{aligned}
\tag{E.43}
$$

Here we use $(1+1/H)^{3H} < 21$.

**Proof of Lemma 19**

Recall that $M_1 = \sum_{k=1}^K \sum_{h=1}^H [\mathbb{D}_{\widetilde{\pi}_h^k} \hat{P}_h^k (\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_h^k) - \hat{P}_h^k(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_h^k, a_h^k)]$.

Since $\mathbb{E}_{a_h^k \sim \mathbb{D}_{\widetilde{\pi}_h^k}}[\hat{P}_h^k(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_h^k, a_h^k)] = \mathbb{D}_{\widetilde{\pi}_h^k} \hat{P}_h^k(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_h^k)$, we have that $\mathbb{D}_{\widetilde{\pi}_h^k} \hat{P}_h^k(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_h^k) - \hat{P}_h^k(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_h^k, a_h^k)$ is a martingale difference sequence. By the Azuma-Hoeffding inequality, with probability $1-\delta$, we have

$$\left| \sum_{k=1}^K \sum_{h=1}^H [\mathbb{D}_{\widetilde{\pi}_h^k} \hat{P}_h^k(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_h^k) - \hat{P}_h^k(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_h^k, a_h^k)] \right| \le H\sqrt{2HK\iota}. \tag{E.44}$$

**Proof of Lemma 20**

Recall that $M_2 = \sum_{k=1}^K \sum_{h=1}^H \frac{1}{H}[\mathbb{D}_{\widetilde{\pi}_h^k} P_h(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_h^k) - P_h(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_h^k, a_h^k)]$.

Since $\mathbb{E}_{a_h^k \sim \mathbb{D}_{\widetilde{\pi}_h^k}}[P_h(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_h^k, a_h^k)] = \mathbb{D}_{\widetilde{\pi}_h^k} P_h(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_h^k)$, we have that $\mathbb{D}_{\widetilde{\pi}_h^k} P_h(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_h^k) - P_h(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_h^k, a_h^k)$ is a martingale difference sequence. By the Azuma-Hoeffding inequality, with probability $1-\delta$, we have

$$\left| \sum_{k=1}^K \sum_{h=1}^H [\mathbb{D}_{\widetilde{\pi}_h^k} P_h(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_h^k) - P_h(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_h^k, a_h^k)] \right| \le H\sqrt{2HK\iota}. \tag{E.45}$$

**Proof of Lemma 21**

Recall that $M_3 = \sum_{k=1}^K \sum_{h=1}^H (P_h^k(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_h^k, a_h^k) - (\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)(s_{h+1}^k))$.

Let the one-hot vector $\hat{\mathbb{1}}_h^k(\cdot|s_h^k, a_h^k)$ to satisfy that $\hat{\mathbb{1}}_h^k(s_{h+1}^k|s_h^k, a_h^k) = 1$ and $\hat{\mathbb{1}}_h^k(s|s_h^k, a_h^k) = 0$ for $s \neq s_{h+1}^k$. Thus, $[(P_h^k - \hat{\mathbb{1}}_h^k)(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)](s_h^k, a_h^k)$ is a martingale difference sequence. By the Azuma-Hoeffding inequality, with probability $1-\delta$, we have

$$\left| \sum_{k=1}^K \sum_{h=1}^H [(P_h^k - \hat{\mathbb{1}}_h^k)(\overline{V}_{h+1}^k - \underline{V}_{h+1}^k)](s_h^k, a_h^k) \right| \le H\sqrt{2HK\iota}. \tag{E.46}$$

**Proof of Lemma 22**

We bounded $M_4 = \sum_{k=1}^{K} \sum_{h=1}^{H} [\frac{(SH+SH^2)\iota}{N_h^k(s_h^k, a_h^k)} + \mathbb{D}_{\widetilde{\pi}_h^k} \Theta_h^k(s_h^k)]$ by separately bounding the four items.

**Bound** $\sum_{k=1}^{K} \sum_{h=1}^{H} \frac{(SH+SH^2)\iota}{N_h^k(s_h^k, a_h^k)}$ We regroup the summands in a different way.

$$\sum_{k=1}^{K} \sum_{h=1}^{H} \frac{(SH+SH^2)\iota}{N_h^k(s_h^k, a_h^k)} = (SH+SH^2)\iota \sum_{h=1}^{H} \sum_{(s,a)\in\mathcal{S}\times\mathcal{A}} \sum_{n=1}^{N_h^K(s,a)} \frac{1}{n} \leq (SH+SH^2)SAH\iota^2. \tag{E.47}$$

Recall that $\Theta_h^k(s,a) = \sqrt{\frac{8\mathbb{V}_{P_h} C_{h+1}^{\pi^{k*}, \underline{\pi}^k, \rho}(s,a)\iota}{N_h^k(s,a)}} + \sqrt{\frac{32}{N_h^k(s,a)}} + \frac{46\sqrt{SH^4\iota}}{N_h^k(s,a)}$.

**Bound** $\sum_{k=1}^{K} \sum_{h=1}^{H} [(1-\rho)\sqrt{\frac{32\iota}{N_h^k(s_h^k, \overline{\pi}_h^k(s_h^k))}} + \rho\sqrt{\frac{32\iota}{N_h^k(s_h^k, \underline{\pi}_h^k(s_h^k))}}]$ We regroup the summands in a different way. For any policy $\pi$, we have

$$\sum_{k=1}^{K} \sum_{h=1}^{H} \sqrt{\frac{32\iota}{N_h^k(s_h^k, \pi(s_h^k))}} = \sum_{h=1}^{H} \sum_{(s,a)\in\mathcal{S}\times\mathcal{A}} \sum_{n=1}^{N_h^K(s,a)} \sqrt{\frac{32\iota}{n}} \leq 8H\sqrt{SAK\iota}. \tag{E.48}$$

**Bound** $\sum_{k=1}^{K} \sum_{h=1}^{H} [(1-\rho)\frac{46SH^2\iota}{N_h^k(s_h^k, \overline{\pi}_h^k(s_h^k))} + \rho\frac{46SH^2\iota}{N_h^k(s_h^k, \underline{\pi}_h^k(s_h^k))}]$ We regroup the summands in a different way. For any policy $\pi$, we have

$$\sum_{k=1}^{K} \sum_{h=1}^{H} \frac{46\sqrt{SH^4\iota}}{N_h^k(s_h^k, \pi(s_h^k))} = 46\sqrt{SH^4\iota} \sum_{h=1}^{H} \sum_{(s,a)\in\mathcal{S}\times\mathcal{A}} \sum_{n=1}^{N_h^K(s,a)} \frac{1}{n} \leq 46S^{\frac{3}{2}}AH^3\iota^2. \tag{E.49}$$

**Bound** $\sum_{k=1}^{K}\sum_{h=1}^{H}\left[(1-\rho)\sqrt{\frac{8\mathbb{V}_{P_h}C_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}(s_h^k,\overline{\pi}_h^k(s_h^k))\iota}{N_h^k(s_h^k,\overline{\pi}_h^k(s_h^k))}}+\rho\sqrt{\frac{8\mathbb{V}_{P_h}C_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}(s_h^k,\underline{\pi}_h^k(s_h^k))\iota}{N_h^k(s_h^k,\underline{\pi}_h^k(s_h^k))}}\right]$ By Cauchy-Schwarz inequality,

$$
\begin{aligned}
&\sum_{k=1}^{K}\sum_{h=1}^{H}\sqrt{\frac{\mathbb{V}_{P_h}C_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}(s_h^k,\overline{\pi}_h^k(s_h^k))\iota}{N_h^k(s_h^k,\overline{\pi}_h^k(s_h^k))}}\\
\leq&\sqrt{\sum_{k=1}^{K}\sum_{h=1}^{H}\mathbb{V}_{P_h}C_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}(s_h^k,\overline{\pi}_h^k(s_h^k))\cdot\sum_{k=1}^{K}\sum_{h=1}^{H}\frac{\iota}{N_h^k(s_h^k,\overline{\pi}_h^k(s_h^k))}} \qquad \text{(E.50)}\\
\leq&\sqrt{SAH\iota^2\sum_{k=1}^{K}\sum_{h=1}^{H}\mathbb{V}_{P_h}C_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}(s_h^k,\overline{\pi}_h^k(s_h^k))}.
\end{aligned}
$$

Similarly,

$$
\begin{aligned}
&\sum_{k=1}^{K}\sum_{h=1}^{H}\sqrt{\frac{\mathbb{V}_{P_h}C_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}(s_h^k,\underline{\pi}_h^k(s_h^k))\iota}{N_h^k(s_h^k,\underline{\pi}_h^k(s_h^k))}}\\
\leq&\sqrt{SAH\iota^2\sum_{k=1}^{K}\sum_{h=1}^{H}\mathbb{V}_{P_h}C_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}(s_h^k,\underline{\pi}_h^k(s_h^k))}.
\end{aligned} \qquad \text{(E.51)}
$$

By $(1-\rho)a^2+\rho b^2\geq((1-\rho)a+\rho b)^2$,

$$
\begin{aligned}
&(1-\rho)\sqrt{\sum_{k=1}^{K}\sum_{h=1}^{H}\mathbb{V}_{P_h}C_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}(s_h^k,\overline{\pi}_h^k(s_h^k))}+\rho\sqrt{\sum_{k=1}^{K}\sum_{h=1}^{H}\mathbb{V}_{P_h}C_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}(s_h^k,\underline{\pi}_h^k(s_h^k))}\\
\leq&\sqrt{\sum_{k=1}^{K}\sum_{h=1}^{H}[(1-\rho)\mathbb{V}_{P_h}C_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}(s_h^k,\overline{\pi}_h^k(s_h^k))+\rho\mathbb{V}_{P_h}C_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}(s_h^k,\underline{\pi}_h^k(s_h^k))]}.
\end{aligned} \qquad \text{(E.52)}
$$

Now we bound the total variance. Let $\mathbb{D}_{\widetilde{\pi}_h^k}P_h(s'|s)=(1-\rho)P_h(s'|s,\overline{\pi}_h^k(s)+\rho P_h(s'|s,\underline{\pi}_h^k(s))$,

$$
[\mathbb{D}_{\widetilde{\pi}_h^k}P_hV_{h+1}](s)=\sum_{s'}[(1-\rho)P_h(s'|s,\overline{\pi}_h^k(s))+\rho P_h(s'|s,\underline{\pi}_h^k(s))]V_{h+1}(s'), \qquad \text{(E.53)}
$$

and

$$\begin{aligned}\mathbb{V}_{[\mathbb{D}_{\widetilde{\pi}_h^k}P_h]}V_{h+1}(s) = &\sum_{s'}[(1-\rho)P_h(s'|s,\overline{\pi}_h^k(s)) + \rho P_h(s'|s,\underline{\pi}_h^k(s))][V_{h+1}(s')]^2 \\ &- [\sum_{s'}\big((1-\rho)P_h(s'|s,\overline{\pi}_h^k(s)) + \rho P_h(s'|s,\underline{\pi}_h^k(s))\big)V_{h+1}(s')]^2.\end{aligned} \tag{E.54}$$

We have that

$$\begin{aligned}&\mathbb{V}_{[\mathbb{D}_{\widetilde{\pi}_h^k}P_h]}C_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}(s_h^k) \\ =&\sum_{s'}[(1-\rho)P_h(s'|s_h^k,\overline{\pi}_h^k(s_h^k)) + \rho P_h(s'|s_h^k,\underline{\pi}_h^k(s_h^k))][C_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}(s')]^2 \\ &- [\sum_{s'}\big((1-\rho)P_h(s'|s_h^k,\overline{\pi}_h^k(s_h^k)) + \rho P_h(s'|s_h^k,\underline{\pi}_h^k(s_h^k))\big)C_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}(s')]^2 \\ \geq&(1-\rho)\mathbb{V}_{P_h}C_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}(s_h^k,\overline{\pi}_h^k(s_h^k)) + \rho\mathbb{V}_{P_h}C_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}(s_h^k,\underline{\pi}_h^k(s_h^k)) \\ &+ (1-\rho)[P_hC_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}(s_h^k,\overline{\pi}_h^k(s_h^k))]^2 + \rho P_h[C_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}(s_h^k,\underline{\pi}_h^k(s_h^k))]^2 \\ &- [\sum_{s'}(1-\rho)P_h(s'|s_h^k,\overline{\pi}_h^k(s_h^k))C_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}(s') + \rho P_h(s'|s_h^k,\underline{\pi}_h^k(s_h^k))C_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}(s')]^2 \\ \geq&(1-\rho)\mathbb{V}_{P_h}C_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}(s_h^k,\overline{\pi}_h^k(s_h^k)) + \rho\mathbb{V}_{P_h}C_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}(s_h^k,\underline{\pi}_h^k(s_h^k)),\end{aligned} \tag{E.55}$$

where the last inequality is due to $(1-\rho)a^2 + \rho b^2 \geq ((1-\rho)a + \rho b)^2$.

With probability $1-2\delta$, we also have that

$$
\begin{aligned}
&\sum_{k=1}^{K}\sum_{h=1}^{H}\mathbb{V}_{[\mathbb{D}_{\widetilde{\pi}_h^k}P_h]}C_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}(s_h^k)\\
=&\sum_{k=1}^{K}\sum_{h=1}^{H}\left([\mathbb{D}_{\widetilde{\pi}_h^k}P_h(C_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho})^2](s_h^k)-\left([\mathbb{D}_{\widetilde{\pi}_h^k}P_hC_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}](s_h^k)\right)^2\right)\\
=&\sum_{k=1}^{K}\sum_{h=1}^{H}\left([\mathbb{D}_{\widetilde{\pi}_h^k}P_h(C_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho})^2](s_h^k)-\left(C_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}(s_{h+1}^k)\right)^2\right)\\
&+\sum_{k=1}^{K}\sum_{h=1}^{H}\left(\left(C_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}(s_{h+1}^k)\right)^2-\left([\mathbb{D}_{\widetilde{\pi}_h^k}P_hC_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}](s_h^k)\right)^2\right)\\
\leq& H^2\sqrt{2HK\iota}+\sum_{k=1}^{K}\sum_{h=1}^{H}\left((C_h^{\pi^{k*},\underline{\pi}^k,\rho}(s_h^k))^2-\left([\mathbb{D}_{\widetilde{\pi}_h^k}P_hC_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}](s_h^k)\right)^2\right)-\sum_{k=1}^{K}(C_1^{\pi^{k*},\underline{\pi}^k,\rho}(s_1^k))^2\\
\leq& H^2\sqrt{2HK\iota}+2H\sum_{k=1}^{K}\sum_{h=1}^{H}|C_h^{\pi^{k*},\underline{\pi}^k,\rho}(s_h^k)-\mathbb{D}_{\widetilde{\pi}_h^k}P_hC_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}(s_h^k)|\\
\leq& H^2\sqrt{2HK\iota}+2H\sum_{k=1}^{K}\left(C_1^{\pi^{k*},\underline{\pi}^k,\rho}(s_1^k)+\sum_{h=1}^{H}\left(C_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}(s_{h+1}^k)-\mathbb{D}_{\widetilde{\pi}_h^k}P_hC_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}(s_h^k,a_h^k)\right)\right)\\
\leq& H^2\sqrt{2HK\iota}+2H^2K+2H^2\sqrt{2HK\iota}\\
\leq& 3H^2K+9H^3\iota/2,
\end{aligned}
\tag{E.56}
$$

where the first inequality holds with probability $1-\delta$ by Azuma-Hoeffding inequality, the second inequality is due to the bound of V-values, the third inequality is due to Lemma 17 so that $C_h^{\pi^{k*},\underline{\pi}^k,\rho}(s_h^k)\geq\mathbb{D}_{\widetilde{\pi}_h^k}D_h^{\pi^{k*},\underline{\pi}^k,\rho}(s_h^k)\geq\mathbb{D}_{\widetilde{\pi}_h^k}P_hC_{h+1}^{\pi^{k*},\underline{\pi}^k,\rho}(s_h^k)$, the fourth inequality holds with probability $1-\delta$ by Azuma-Hoeffding inequality, and the last inequality holds with $2ab\leq a^2+b^2$.

In summary, with probability at least $1-\delta$, we have $\sum_{k=1}^{K}\sum_{h=1}^{H}\mathbb{V}_{P_h}V_{h+1}^{\underline{\pi}^k}(s_h^k,a_h^k)\leq(H^2K+H^3\iota)$.

In summary,

$$\begin{aligned}\sum_{k=1}^{K}\sum_{h=1}^{H}\mathbb{D}_{\widetilde{\pi}_h^k}\Theta_h^k(s_h^k) &\leq 8\sqrt{SAH^2K\iota}+46S^{\frac{3}{2}}AH^3\iota^2+\sqrt{24SAH^3K\iota^2+36SAH^5\iota^2} \\ &\leq 8\sqrt{SAH^2K\iota}+46S^{\frac{3}{2}}AH^3\iota^2+\sqrt{24SAH^3K}\iota+6\sqrt{SAH^5}\iota.\end{aligned} \tag{E.57}$$

## E.3 Proof for model-free algorithm

In this section, we prove Theorem 21. Recall that we use $\overline{Q}_h^k$,$\overline{V}_h^k$,$\underline{Q}_h^k$,$\underline{V}_h^k$ and $N_h^k$ to denote the values of $\overline{Q}_h$,$\overline{V}_h$,$\underline{Q}_h$,$\underline{V}_h$ and $\max\{N_h,1\}$ at the beginning of the $k$-th episode.

**Property of Learning Rate $\alpha_t$** We refer the readers to the setting of the learning rate $\alpha_t := \frac{H+1}{H+t}$ and the Lemma 4.1 in [41]. For notational convenience, define $\alpha_t^0 := \prod_{j=1}^{t}(1-\alpha_t)$ and $\alpha_t^i := \alpha_i\prod_{j=i+1}^{t}(1-\alpha_t)$. Here, we introduce some useful properties of $\alpha_t^i$ which were proved in [41]:

(1) $\sum_{i=1}^{t}\alpha_t^i = 1$ and $\alpha_t^0 = 0$ for $t \geq 1$;

(2) $\sum_{i=1}^{t}\alpha_t^i = 0$ and $\alpha_t^0 = 1$ for $t = 0$;

(3) $\frac{1}{\sqrt{t}} \leq \sum_{i=1}^{t}\frac{\alpha_t^i}{\sqrt{t}} \leq \frac{2}{\sqrt{t}}$ for every $t \geq 1$;

(4) $\sum_{i=1}^{t}(\alpha_t^i)^2 \leq \frac{2H}{t}$ for every $t \geq 1$;

(5) $\sum_{t=i}^{\infty}\alpha_t^i \leq (1+\frac{1}{H})$ for every $i \geq 1$.

**Recursion on $Q$** As shown in [41], at any $(s,a,h,k) \in \mathcal{S}\times\mathcal{A}\times[H]\times[K]$, let $t = N_h^k(s,a)$ and suppose $(s,a)$ was previously taken by the agent at step $h$ of episodes $k_1,k_2,\ldots,k_t < k$. By the update equations in Algorithm 6.2 and the definition of $\alpha_t^i$, we have

$$\begin{aligned}\overline{Q}_h^k(s,a) &= \alpha_t^0(H-h+1)+\sum_{i=1}^{t}\alpha_t^i\left(r_h^{k_i}+\overline{V}_{h+1}^{k_i}(s_{h+1}^{k_i})+b_i\right); \\ \underline{Q}_h^k(s,a) &= \sum_{i=1}^{t}\alpha_t^i\left(r_h^{k_i}+\underline{V}_{h+1}^{k_i}(s_{h+1}^{k_i})-b_i\right).\end{aligned} \tag{E.58}$$

Thus,

$$\begin{aligned}(\overline{Q}_h^k - Q_h^*)(s,a) =&\alpha_t^0(H-h+1) + \sum_{i=1}^t \alpha_t^i \left(r_h^{k_i} + \overline{V}_{h+1}^{k_i}(s_{h+1}^{k_i}) + b_i\right) \\ &- \left(\alpha_t^0 Q_h^*(s,a) + \sum_{i=1}^t \alpha_t^i \left(R_h(s,a) + P_h V_{h+1}^*(s,a)\right)\right) \\ =&\alpha_t^0(H-h+1-Q_h^*(s,a)) + \sum_{i=1}^t \alpha_t^i \left((\overline{V}_{h+1}^{k_i} - V_{h+1}^*)(s_{h+1}^{k_i})\right) \\ &+ \sum_{i=1}^t \alpha_t^i \left((r_h^{k_i} - R_h(s,a)) + V_{h+1}^*(s_{h+1}^{k_i}) - P_h V_{h+1}^*(s,a) + b_i\right),\end{aligned} \tag{E.59}$$

and similarly

$$\begin{aligned}(\underline{Q}_h^k - Q_h^{\overline{\pi}^k})(s,a) =&\sum_{i=1}^t \alpha_t^i \left(r_h^{k_i} + \underline{V}_{h+1}^{k_i}(s_{h+1}^{k_i}) - b_i\right) \\ &- \left(\alpha_t^0 Q_h^{\overline{\pi}^k}(s,a) + \sum_{i=1}^t \alpha_t^i \left(R_h(s,a) + P_h V_{h+1}^{\overline{\pi}^k}(s,a)\right)\right) \\ =&-\alpha_t^0 Q_h^{\overline{\pi}^k}(s,a) + \sum_{i=1}^t \alpha_t^i \left([P_h(\underline{V}_{h+1}^{k_i} - V_{h+1}^{\overline{\pi}^k})](s,a)\right) \\ &+ \sum_{i=1}^t \alpha_t^i \left((r_h^{k_i} - R_h(s,a)) + \underline{V}_{h+1}^{k_i}(s_{h+1}^{k_i}) - P_h \underline{V}_{h+1}^{k_i}(s,a) - b_i\right).\end{aligned} \tag{E.60}$$

In addition, for any $k' \leq k$, let $t' = N_h^{k'}(s,a)$. Thus, $(s,a)$ was previously taken by the agent at step $h$ of episodes $k_1, k_2, \ldots, k_{t'} < k'$. We have

$$\begin{aligned}(\underline{Q}_h^{k'} - Q_h^{\overline{\pi}^k})(s,a) =&-\alpha_t^0 Q_h^{\overline{\pi}^k}(s,a) + \sum_{i=1}^{t'} \alpha_{t'}^i \left([P_h(\underline{V}_{h+1}^{k_i} - V_{h+1}^{\overline{\pi}^k})](s,a)\right) \\ &+ \sum_{i=1}^{t'} \alpha_{t'}^i \left((r_h^{k_i} - R_h(s,a)) + \underline{V}_{h+1}^{k_i}(s_{h+1}^{k_i}) - P_h \underline{V}_{h+1}^{k_i}(s,a) - b_i\right).\end{aligned} \tag{E.61}$$

**Confidence Bounds** By the Azuma-Hoeffding inequality, with probability $1-\delta$, we have that for all $s$, $a$, $h$ and $t \le K$,

$$\left|\sum_{i=1}^{t} \alpha_t^i \left((r_h^{k_i} - R_h(s,a)) + \underline{V}_{h+1}^{k_i}(s_{h+1}^{k_i}) - P_h \underline{V}_{h+1}^{k_i}(s,a)\right)\right| \le H\sqrt{\sum_{i=1}^{t} (\alpha_t^i)^2 \iota/2} \le \sqrt{H^3\iota/t}. \tag{E.62}$$

At the same time, with probability $1-\delta$, we have that for all $s$, $a$, $h$ and $t \le K$,

$$\left|\sum_{i=1}^{t} \alpha_t^i \left((r_h^{k_i} - R_h(s,a)) + V_{h+1}^{*}(s_{h+1}^{k_i}) - P_h V_{h+1}^{*}(s,a)\right)\right| \le \sqrt{H^3\iota/t}. \tag{E.63}$$

In addition, we have $\sqrt{H^3\iota/t} \le \sum_{i=1}^{t} \alpha_t^i b_i \le 2\sqrt{H^3\iota/t}$.

**Monotonicity** Now we prove that $\overline{V}_h^k(s) \ge V_h^*(s) \ge V_h^{\overline{\pi}^k}(s) \ge \underline{V}_h^k(s)$ and $\overline{Q}_h^k(s,a) \ge Q_h^*(s,a) \ge Q_h^{\overline{\pi}^k}(s,a) \ge \underline{Q}_h^k(s,a)$ for all $(s,a,h,k) \in S \times A \times [H] \times [K]$.

At step $H+1$, we have $\overline{V}_{H+1}^k(s) = V_{H+1}^*(s) = V_{H+1}^{\overline{\pi}^k}(s) = \underline{V}_{H+1}^k(s) = 0$ and $\overline{Q}_{H+1}^k(s,a) = Q_{H+1}^*(s,a) = Q_{H+1}^{\overline{\pi}^k}(s,a) = \underline{Q}_{H+1}^k(s,a) = 0$ for all $(s,a,k) \in S \times A \times [K]$.

Consider any step $h \in [H]$ in any episode $k \in [K]$, and suppose that the monotonicity is satisfied for all previous episodes as well as all steps $h' \ge h+1$ in the current episode, which is

$$\begin{aligned}
&\overline{V}_{h'}^{k'}(s) \ge V_{h'}^*(s) \ge V_{h'}^{\overline{\pi}^{k'}}(s) \ge \underline{V}_{h'}^{k'}(s) \; \forall (k',h',s) \in [k-1] \times [H+1] \times \mathcal{S}, \\
&\overline{Q}_{h'}^{k'}(s,a) \ge Q_{h'}^*(s,a) \ge Q_{h'}^{\overline{\pi}^{k'}}(s,a) \ge \underline{Q}_{h'}^{k'}(s,a) \; \forall (k',h',s,a) \in [k-1] \times [H+1] \times \mathcal{S} \times \mathcal{A}, \\
&\overline{V}_{h'}^{k}(s) \ge V_{h'}^*(s) \ge V_{h'}^{\overline{\pi}^{k}}(s) \ge \underline{V}_{h'}^{k}(s) \; \forall h' \ge h+1 \text{ and } s \in \mathcal{S}, \\
&\overline{Q}_{h'}^{k}(s,a) \ge Q_{h'}^*(s,a) \ge Q_{h'}^{\overline{\pi}^{k}}(s,a) \ge \underline{Q}_{h'}^{k}(s,a) \; \forall h' \ge h+1 \text{ and } (s,a) \in \mathcal{S} \times \mathcal{A}.
\end{aligned} \tag{E.64}$$

We first show the monotonicity of $Q$ values. We have

$$(\overline{Q}_h^k - Q_h^*)(s,a) \ge \alpha_t^0 (H - h + 1 - Q_h^*(s,a)) + \sum_{i=1}^{t} \alpha_t^i \left((\overline{V}_{h+1}^{k_i} - V_{h+1}^*)(s_{h+1}^{k_i})\right) \ge 0, \tag{E.65}$$

and, by to the update rule of $\underline{V}$ values (line 13) in Algorithm 6.2,

$$\begin{aligned}(\underline{Q}_h^k - Q_h^{\overline{\pi}^k})(s,a) \le & -\alpha_t^0 Q_h^{\overline{\pi}^k}(s,a) + \sum_{i=1}^{t} \alpha_t^i \left([P_h(\underline{V}_{h+1}^{k_i} - V_{h+1}^{\overline{\pi}^k})](s,a)\right) \\ \le & -\alpha_t^0 Q_h^{\overline{\pi}^k}(s,a) + \sum_{i=1}^{t} \alpha_t^i \left([P_h(\underline{V}_{h+1}^{k} - V_{h+1}^{\overline{\pi}^k})](s,a)\right) \le 0.\end{aligned} \tag{E.66}$$

In addition, for any $k' \le k$,

$$\begin{aligned}(\underline{Q}_h^{k'} - Q_h^{\overline{\pi}^k})(s,a) \le & -\alpha_t^0 Q_h^{\overline{\pi}^k}(s,a) + \sum_{i=1}^{t'} \alpha_{t'}^i \left([P_h(\underline{V}_{h+1}^{k_i} - V_{h+1}^{\overline{\pi}^k})](s,a)\right) \\ \le & -\alpha_t^0 Q_h^{\overline{\pi}^k}(s,a) + \sum_{i=1}^{t'} \alpha_{t'}^i \left([P_h(\underline{V}_{h+1}^{k} - V_{h+1}^{\overline{\pi}^k})](s,a)\right) \le 0.\end{aligned} \tag{E.67}$$

Then, we show the monotonicity of $V$ values. We have that

$$\begin{aligned}&(1-\rho)\max_a \overline{Q}_h^k(s,a) + \rho\overline{Q}_h^k(s, \arg\min_a \underline{Q}_h^k(s,a)) \\ \ge&(1-\rho)\max_a \overline{Q}_h^k(s,a) + \rho Q_h^*(s, \arg\min_a \underline{Q}_h^k(s,a)) \\ \ge&(1-\rho)\overline{Q}_h^k(s,\pi_h^*(s)) + \rho\min_{a\in\mathcal{A}} Q_h^*(s,a) \\ \ge&(1-\rho)Q_h^*(s,\pi_h^*(s)) + \rho\min_{a\in\mathcal{A}} Q_h^*(s,a) = V_h^*(s).\end{aligned} \tag{E.68}$$

By the update rule of $\overline{V}$ values (line 12) in Algorithm 6.2,

$$\overline{V}_h^k(s) = \min\{\overline{V}_h^{k-1}(s), (1-\rho)\max_a \overline{Q}_h^k(s,a) + \rho\overline{Q}_h^k(s, \arg\min_a \underline{Q}_h^k(s,a))\} \ge V_h^*(s). \tag{E.69}$$

Here, we need use the update rule of policy $\underline{\pi}$ (line 11-16) in Algorithm 6.2. Define $\tau(k,h,s) := \max\{k' : k' < k \text{ and } \underline{V}_h^{k'+1}(s) = (1-\rho)\underline{Q}_h^{k'+1}(s, \arg\max_a \overline{Q}_h^{k'+1}(s,a)) + \rho\min_a \underline{Q}_h^{k'+1}(s,a)\}$, which denotes the last episode (before the beginning of the episode $k$), in which the $\overline{\pi}$ and $\underline{V}$ was updated at $(h,s)$. For notational simplicity, we use $\tau$ to denote $\tau(k,h,s)$ here. After the end of episode $\tau$ and before the beginning of the episode $k$, the agent policy $\overline{\pi}$ was not updated and $\underline{V}$ was not updated at $(h, s)$, i.e. $\underline{V}_h^k(s) = \underline{V}_h^{\tau+1}(s) = (1-\rho)\underline{Q}_h^{\tau+1}(s, \overline{\pi}_h^{\tau+1}(s)) + \rho \min_a \underline{Q}_h^{\tau+1}(s, a)$ and $\overline{\pi}_h^k(s) = \overline{\pi}_h^{\tau+1}(s) = \arg\max_a \overline{Q}_h^{\tau+1}(s, a))$. Thus,

$$\begin{aligned}\underline{V}_h^k(s) =& (1-\rho)\underline{Q}_h^{\tau+1}(s, \overline{\pi}_h^{\tau+1}(s)) + \rho \min_a \underline{Q}_h^{\tau+1}(s, a) \\ \leq& (1-\rho) Q_h^{\overline{\pi}^k}(s, \overline{\pi}_h^{\tau+1}(s)) + \rho \min_a \underline{Q}_h^{\tau+1}(s, a) \\ \leq& (1-\rho) Q_h^{\overline{\pi}^k}(s, \overline{\pi}_h^k(s)) + \rho \underline{Q}_h^{\tau+1}(s, \arg\min_{a \in \mathcal{A}} Q_h^{\overline{\pi}^k}(s, a)) \\ \leq& (1-\rho) Q_h^{\overline{\pi}^k}(s, \overline{\pi}_h^k(s)) + \rho \min_{a \in \mathcal{A}} Q_h^{\overline{\pi}^k}(s, a) = V_h^{\overline{\pi}^k}(s).\end{aligned} \tag{E.70}$$

By induction from $h = H + 1$ to 1 and $k = 1$ to $K$, we can conclude that $\overline{V}_h^k(s) \geq V_h^*(s) \geq V_h^{\overline{\pi}^k}(s) \geq \underline{V}_h^k(s)$ and $\overline{Q}_h^k(s, a) \geq Q_h^*(s, a) \geq Q_h^{\overline{\pi}^k}(s, a) \geq \underline{Q}_h^k(s, a)$ for all $(s, a, h, k) \in S \times A \times [H] \times [K]$.

**Regret Analysis** According to the monotonicity, the regret can be bounded by

$$Regret(K) := \sum_{k=1}^K (V_1^*(s_1^k) - V_1^{\overline{\pi}^k}(s_1^k)) \leq \sum_{k=1}^K (\overline{V}_1^k(s_1^k) - \underline{V}_1^k(s_1^k)). \tag{E.71}$$

By the update rules in Algorithm 6.2, we have

$$\begin{aligned}& \overline{V}_h^k(s_h^k) - \underline{V}_h^k(s_h^k) \\ \leq& (1-\rho)\overline{Q}_h^k(s_h^k, \arg\max_a \overline{Q}_h^k(s_h^k, a)) + \rho \overline{Q}_h^k(s_h^k, \arg\min_a \underline{Q}_h^k(s_h^k, a)) \\ & - (1-\rho)\underline{Q}_h^k(s_h^k, \arg\max_a \overline{Q}_h^k(s_h^k, a)) + \rho \underline{Q}_h^k(s_h^k, \arg\min_a \underline{Q}_h^k(s_h^k, a)) \\ =& (1-\rho)[\overline{Q}_h^k - \underline{Q}_h^k](s_h^k, \overline{a}_h^k) + \rho[\overline{Q}_h^k - \underline{Q}_h^k](s_h^k, \underline{a}_h^k) \\ =& [\overline{Q}_h^k - \underline{Q}_h^k](s_h^k, a_h^k) + [\mathbb{D}_{\widetilde{\pi}_h^k}(\overline{Q}_h^k - \underline{Q}_h^k)](s_h^k) - [\overline{Q}_h^k - \underline{Q}_h^k](s_h^k, a_h^k).\end{aligned} \tag{E.72}$$

Set $n_h^k = N_h^k(s_h^k, a_h^k)$ and where $k_i(s_h^k, a_h^k)$ is the episode in which $(s_h^k, a_h^k)$ was taken at step $h$ for the $i$-th time. For notational simplicity, we set $\phi_h^k = \overline{V}_h^k(s_h^k) - \underline{V}_h^k(s_h^k)$ and $\xi_h^k = [\mathbb{D}_{\widetilde{\pi}_h^k}(\overline{Q}_h^k -$

$\underline{Q}_h^k)](s_h^k) - [\overline{Q}_h^k - \underline{Q}_h^k](s_h^k, a_h^k)$. According to the update rules,

$$\begin{aligned}\phi_h^k =& \overline{V}_h^k(s_h^k) - \underline{V}_h^k(s_h^k)\\ \le& \alpha^0_{n_h^k}(H-h+1) + \sum_{i=1}^{n_h^k} \alpha^i_{n_h^k}\left(\overline{V}_{h+1}^{k_i(s_h^k,a_h^k)}(s_{h+1}^{k_i(s_h^k,a_h^k)}) - \underline{V}_{h+1}^{k_i(s_h^k,a_h^k)}(s_{h+1}^{k_i(s_h^k,a_h^k)}) + 2b_i\right)\\ &+ [\mathbb{D}_{\widetilde{\pi}_h^k}(\overline{Q}_h^k - \underline{Q}_h^k)](s_h^k) - [\overline{Q}_h^k - \underline{Q}_h^k](s_h^k, a_h^k)\\ =& \alpha^0_{n_h^k}(H-h+1) + \sum_{i=1}^{n_h^k} \alpha^i_{n_h^k}(\phi_{h+1}^{k_i(s_h^k,a_h^k)} + 2b_i) + \xi_h^k\\ \le& \alpha^0_{n_h^k}(H-h+1) + \sum_{i=1}^{n_h^k} \alpha^i_{n_h^k}\phi_{h+1}^{k_i(s_h^k,a_h^k)} + \xi_h^k + 4\sqrt{H^3\iota/n_h^k}.\end{aligned} \tag{E.73}$$

We add $\overline{V}_h^k(s_h^k) - \underline{V}_h^k(s_h^k)$ over $k$ and regroup the summands in a different way. Note that for any episode $k$, the term $\sum_{i=1}^{n_h^k} \alpha^i_{n_h^k}\phi_{h+1}^{k_i(s_h^k,a_h^k)}$ takes all the prior episodes $k_i < k$ where $(s_h^k, a_h^k)$ was taken into account. In other words, for any episode $k'$, the term $\phi_{h+1}^{k'}$ appears in the summands at all posterior episodes $k > k'$ where $(s_h^{k'}, a_h^{k'})$ was taken. The first time it appears we have $n_h^k = n_h^{k'} + 1$, and the second time it appears we have $n_h^k = n_h^{k'} + 2$, and so on. Thus, we have

$$\begin{aligned}&\sum_{k=1}^K(\overline{V}_h^k(s_h^k) - \underline{V}_h^k(s_h^k))\\ \le& \sum_{k=1}^K \alpha^0_{n_h^k}(H-h+1) + \sum_{k=1}^K\sum_{i=1}^{n_h^k} \alpha^i_{n_h^k}\phi_{h+1}^{k_i(s_h^k,a_h^k)} + \sum_{k=1}^K \xi_h^k + \sum_{k=1}^K 4\sqrt{H^3\iota/n_h^k}\\ =& \sum_{k=1}^K \alpha^0_{n_h^k}(H-h+1) + \sum_{k'=1}^K \phi_{h+1}^{k'} \sum_{t=n_h^{k'}+1}^{n_h^K} \alpha_t^{n_h^{k'}} + \sum_{k=1}^K \xi_h^k + \sum_{k=1}^K 4\sqrt{H^3\iota/n_h^k}\\ \le& \sum_{k=1}^K \alpha^0_{n_h^k}(H-h+1) + (1+1/H)\sum_{k=1}^K \phi_{h+1}^k + \sum_{k=1}^K \xi_h^k + \sum_{k=1}^K 4\sqrt{H^3\iota/n_h^k}\end{aligned} \tag{E.74}$$

where the final inequality uses the property $\sum_{t=i}^{\infty} \alpha_t^i \le (1 + \frac{1}{H})$ for every $i \ge 1$.

Taking the induction from $h = 1$ to $H$, we have

$$\begin{aligned}
&\sum_{k=1}^{K}(\overline{V}_1^k(s_1^k) - \underline{V}_1^k(s_1^k)) \\
\leq& 3\sum_{h=1}^{H}\sum_{k=1}^{K}\alpha^0_{n_h^k}(H-h+1) + 3\sum_{h=1}^{H}\sum_{k=1}^{K}\xi_h^k + \sum_{h=1}^{H}\sum_{k=1}^{K}12\sqrt{H^3\iota/n_h^k}
\end{aligned} \tag{E.75}$$

where we use the fact that $(1 + 1/H)^H < 3$ and $\phi^k_{H+1} = 0$ for all $k$.

We bound the three items separately.

(1) We have $\sum_{h=1}^{H}\sum_{k=1}^{K}\alpha^0_{n_h^k}(H-h+1) = \sum_{h=1}^{H}\sum_{k=1}^{K}\mathbb{1}[n_h^k = 0](H-h+1) \leq SAH^2$.

(2) Similar to Lemma 19, by the Azuma-Hoeffding inequality, with probability $1-\delta$, we have $\sum_{h=1}^{H}\sum_{k=1}^{K}\xi_h^k \leq H\sqrt{2HK\iota}$.

(3) We have $\sum_{h=1}^{H}\sum_{k=1}^{K}12\sqrt{H^3\iota/n_h^k} = \sum_{h=1}^{H}\sum_{(s,a)}\sum_{n=1}^{N_h^K(s,a)}\sqrt{H^3\iota/n} \leq H\sqrt{2H^3SAK\iota}$.

In summary,

$$Regret(K) = \sum_{k=1}^{K}(V_1^*(s_1^k) - V_1^{\overline{\pi}^k}(s_1^k)) \leq \mathcal{O}(\sqrt{SAH^5K\iota} + SAH^2)$$

and

$$\begin{aligned}
V_1^*(s_1) - V_1^{\pi^{out}}(s_1) \leq& \overline{V}_1^{K+1}(s_1) - \underline{V}_1^{K+1}(s_1) \\
=& \min_{k\in[K+1]}(\overline{V}_1^k(s_1^k) - \underline{V}_1^k(s_1^k)) \\
\leq& O\left(\frac{\sqrt{SAH^5\iota}}{K} + \frac{SAH^2}{K}\right).
\end{aligned} \tag{E.76}$$

# Bibliography

[1] Yasin Abbasi-yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. In *Advances in Neural Information Processing Systems*, volume 24, pages 2312–2320, Granada, Spain, Dec. 2011.

[2] Shipra Agrawal and Navin Goyal. Thompson sampling for contextual bandits with linear payoffs. In *Proc. of International Conference on Machine Learning*, volume 28, pages 127–135, Atlanta, GA, Jun. 2013.

[3] Jean-Baptiste Alayrac, Jonathan Uesato, Po-Sen Huang, Alhussein Fawzi, Robert Stanforth, and Pushmeet Kohli. Are labels required for improving adversarial robustness? In *Advances in Neural Information Processing Systems*, volume 32, pages 12214–12223, Vancouver, Canada, Dec. 2019.

[4] S. Alfeld, X. Zhu, and P. Barford. Data poisoning attacks against autoregressive models. In *Proc. of AAAI Conference on Artificial Intelligence*, pages 1452–1458, Phoenix, AZ, Feb. 2016.

[5] Mohammad Gheshlaghi Azar, Ian Osband, and Rémi Munos. Minimax regret bounds for reinforcement learning. In *Proc. of International Conference on Machine Learning*, volume 70, pages 263–272, Sydney, Australia, Aug. 2017.

[6] Yu Bai and Chi Jin. Provable self-play algorithms for competitive reinforcement learning. In *Proc. of International Conference on Machine Learning*, volume 119, pages 551–560, Jul. 2020.

[7] Kiarash Banihashem, Adish Singla, and Goran Radanovic. Defense against reward poisoning attacks in reinforcement learning. *Transactions on Machine Learning Research*, 2023.

[8] Vahid Behzadan and Arslan Munir. Vulnerability of deep reinforcement learning to policy induction attacks. In *International Conference on Machine Learning and Data Mining in Pattern Recognition*, pages 262–275, New York, NY, Jul. 2017.

[9] Battista Biggio, Blaine Nelson, and Pavel Laskov. Poisoning attacks against support vector machines. In *Proc. of International Conference on Machine Learning*, pages 1467–1474, Edinburgh, Scotland, Jun, 2012.

[10] Ilija Bogunovic, Arpan Losalka, Andreas Krause, and Jonathan Scarlett. Stochastic linear bandits robust to adversarial attacks. In *Proc. of International Conference on Artificial Intelligence and Statistics*, volume 130, page 991–999, Apr 2021.

[11] Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. Openai gym. *arXiv preprint arXiv:1606.01540*, 2016.

[12] Sébastien Bubeck, Nicolo Cesa-Bianchi, et al. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations and Trends® in Machine Learning*, 5(1):1–122, 2012.

[13] Yair Carmon, Aditi Raghunathan, Ludwig Schmidt, John C Duchi, and Percy S Liang. Unlabeled data improves adversarial robustness. In *Advances in Neural Information Processing Systems*, volume 32, page 11192–11203, Vancouver, Canada, Dec. 2019.

[14] Yifang Chen, Simon Du, and Kevin Jamieson. Improved corruption robust algorithms for episodic reinforcement learning. In *Proc. of International Conference on Machine Learning*, volume 139, pages 1561–1570, Jul. 2021.

[15] Wang Chi Cheung, David Simchi-Levi, and Ruihao Zhu. Reinforcement learning for non-stationary markov decision processes: The blessing of (more) optimism. In *Proc. of International Conference on Machine Learning*, volume 119, pages 1843–1854, Jul. 2020.

[16] Ferdinando Cicalese, Eduardo Laber, Marco Molinaro, et al. Teaching with limited information on the learner's behaviour. In *Proc. of International Conference on Machine Learning*, volume 119, pages 2016–2026, Jul. 2020.

[17] Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. Certified adversarial robustness via randomized smoothing. In *Proc. of International Conference on Machine Learning*, volume 97, pages 1310–1320, Long Beach, CA, Jun. 2019.

[18] Christoph Dann, Tor Lattimore, and Emma Brunskill. Unifying pac and regret: Uniform pac bounds for episodic reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 30, page 5717–5727, Long Beach, CA, Dec. 2017.

[19] Christoph Dann, Lihong Li, Wei Wei, and Emma Brunskill. Policy certificates: Towards accountable reinforcement learning. In *Proc. of International Conference on Machine Learning*, volume 97, pages 1507–1516, Long Beach, CA, Jun. 2019.

[20] Sanjoy Dasgupta, Daniel Hsu, Stefanos Poulis, and Xiaojin Zhu. Teaching a black-box learner. In *Proc. of International Conference on Machine Learning*, volume 97, pages 1547–1555, Long Beach, CA, Jun. 2019.

[21] Qin Ding, Cho-Jui Hsieh, and James Sharpnack. Robust stochastic linear contextual bandits under adversarial attacks. In *Proc. of International Conference on Artificial Intelligence and Statistics*, volume 151, pages 7111–7123, Mar. 2022.

[22] Elvis Dohmatob. Generalized no free lunch theorem for adversarial robustness. In *Proc. of International Conference on Machine Learning*, volume 97, pages 1646–1654, Long Beach, CA, Jun. 2019.

[23] Yingjie Fei, Zhuoran Yang, Zhaoran Wang, and Qiaomin Xie. Dynamic regret of policy optimization in non-stationary environments. In *Advances in Neural Information Processing Systems*, volume 33, pages 6743–6754, Dec. 2020.

[24] Zhe Feng, David Parkes, and Haifeng Xu. The intrinsic robustness of stochastic bandits to strategic manipulation. In *Proc. of International Conference on Machine Learning*, volume 119, pages 3092–3101, Jul. 2020.

[25] David A Freedman. On tail probabilities for martingales. *the Annals of Probability*, pages 100–118, 1975.

[26] Evrard Garcelon, Baptiste Roziere, Laurent Meunier, Jean Tarbouriech, Olivier Teytaud, Alessandro Lazaric, and Matteo Pirotta. Adversarial attacks on linear contextual bandits. In *Advances in Neural Information Processing Systems*, pages 14362–14373, Dec. 2020.

[27] Adam Gleave, Michael Dennis, Cody Wild, Neel Kant, Sergey Levine, and Stuart Russell. Adversarial policies: Attacking deep reinforcement learning. In *International Conference on Learning Representations*, Apr. 2020.

[28] Ken Goldberg, Theresa Roeder, Dhruv Gupta, and Chris Perkins. Eigentaste: A constant time collaborative filtering algorithm. *Information Retrieval*, 4(2):133–151, 2001.

[29] Ian Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*, San Diego, CA, May 2015.

[30] Z. Guan, K. Ji, D. Bucci, T. Hu, J. Palombo, M. Liston, and Y. Liang. Robust stochastic bandit algorithms under probabilistic unbounded adversarial attack. In *Proc. of the AAAI Conference on Artificial Intelligence*, volume 34, pages 4036–4043, New York City, NY, Feb. 2020.

[31] Kaiyang Guo, Shao Yunfeng, and Yanhui Geng. Model-based offline reinforcement learning with pessimism-modulated dynamics belief. In *Advances in Neural Information Processing Systems*, volume 35, pages 449–461, New Orleans, LA, Dec. 2022.

[32] Wenbo Guo, Xian Wu, Sui Huang, and Xinyu Xing. Adversarial policy learning in two-player competitive games. In *Proc. of International Conference on Machine Learning*, volume 139, pages 3910–3919, Jul. 2021.

[33] Anupam Gupta, Tomer Koren, and Kunal Talwar. Better algorithms for stochastic bandits with adversarial corruptions. In *Proc. of Conference on Learning Theory*, volume 99, pages 1562–1578, Phoenix, AZ, Jun. 2019.

[34] F Maxwell Harper and Joseph A Konstan. The movielens datasets: History and context. *Acm Transactions on Interactive Intelligent Systems (TIIS)*, 5(4):1–19, 2015.

[35] Jiafan He, Dongruo Zhou, and Quanquan Gu. Logarithmic regret for reinforcement learning with linear function approximation. In *Proc. of International Conference on Machine Learning*, volume 139, pages 4171–4180, Jul. 2021.

[36] Jiafan He, Dongruo Zhou, and Quanquan Gu. Nearly minimax optimal reinforcement learning for discounted mdps. *Advances in Neural Information Processing Systems*, 34:22288–22300, Dec. 2021.

[37] Sandy Huang, Nicolas Papernot, Ian J. Goodfellow, Yan Duan, and Pieter Abbeel. Adversarial attacks on neural network policies. In *International Conference on Learning Representations Workshop Track Proceedings*, Toulon, France, Apr. 2017.

[38] Yunhan Huang and Quanyan Zhu. Deceptive reinforcement learning under adversarial manipulations on cost signals. In *International Conference on Decision and Game Theory for Security*, pages 217–237, Stockholm, Sweden, Oct. 2019.

[39] Garud N. Iyengar. Robust dynamic programming. *Mathematics of Operations Research*, 30(2):257–280, May 2005.

[40] Thomas Jaksch, Ronald Ortner, and Peter Auer. Near-optimal regret bounds for reinforcement learning. *Journal of Machine Learning Research*, 11:1563–1600, Aug. 2010.

[41] Chi Jin, Zeyuan Allen-Zhu, Sebastien Bubeck, and Michael I Jordan. Is q-learning provably efficient? *Advances in Neural Information Processing Systems*, 31:4868–4878, Dec. 2018.

[42] Chi Jin, Qinghua Liu, Yuanhao Wang, and Tiancheng Yu. V-learning–a simple, efficient, decentralized algorithm for multiagent rl. *Mathematics of Operations Research*, Nov. 2023.

[43] Kwang-Sung Jun, Lihong Li, Yuzhe Ma, and Jerry Zhu. Adversarial attacks on stochastic bandits. In *Advances in Neural Information Processing Systems*, pages 3644–3653, Montréal, Canada, Dec. 2018.

[44] Michael Kearns and Satinder Singh. Near-optimal reinforcement learning in polynomial time. *Machine learning*, 49:209–232, 2002.

[45] Richard Klima, Daan Bloembergen, Michael Kaisers, and Karl Tuyls. Robust temporal difference learning for critical domains. In *Proc. of International Conference on Autonomous Agents and MultiAgent Systems*, pages 350–358, Montréal, Canada, May 2019.

[46] Alexey Kurakin, Ian J. Goodfellow, and Samy Bengio. Adversarial machine learning at scale. In *International Conference on Learning Representations*, Toulon, France, Apr. 2017.

[47] Erwan Lecarpentier and Emmanuel Rachelson. Non-stationary markov decision processes, a worst-case approach using model-based reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 32, pages 7216–7225, Vancouver, Canada, Dec. 2019.

[48] Xian Yeow Lee, Sambit Ghadai, Kai Liang Tan, Chinmay Hegde, and Soumik Sarkar. Spatiotemporally constrained action space attacks on deep reinforcement learning agents.

In *Proc. of the AAAI Conference on Artificial Intelligence*, pages 4577–4584, New York City, NY, Feb. 2020.

[49] B. Li, Y. Wang, A. Singh, and Y. Vorobeychik. Data poisoning attacks on factorization-based collaborative filtering. In *Advances in Neural Information Processing Systems*, pages 1885–1893, Barcelona Spain, Dec. 2016.

[50] Fuwei Li, Lifeng Lai, and Shuguang Cui. On the adversarial robustness of LASSO based feature selection. *IEEE Transactions on Signal Processing*, 69:5555–5567, 2021.

[51] Fuwei Li, Lifeng Lai, and Shuguang Cui. Optimal feature manipulation attacks against linear regression. *IEEE Transactions on Signal Processing*, 69:5580–5594, 2021.

[52] Gen Li, Yuting Wei, Yuejie Chi, Yuantao Gu, and Yuxin Chen. Sample complexity of asynchronous q-learning: Sharper analysis and variance reduction. In *Advances in Neural Information Processing Systems*, volume 33, pages 7031–7043, Dec. 2020.

[53] L. Li, W. Chu, J. Langford, and R. Schapire. A contextual-bandit approach to personalized news article recommendation. In *Proc. of International Conference on World Wide Web*, pages 661–670, Raleigh, NC, Apr. 2010.

[54] Lihong Li, Yu Lu, and Dengyong Zhou. Provably optimal algorithms for generalized linear contextual bandits. In *Proc. of International Conference on Machine Learning*, pages 2071–2080, Sydney, Australia, Aug. 2017.

[55] Yen-Chen Lin, Zhang-Wei Hong, Yuan-Hong Liao, Meng-Li Shih, Ming-Yu Liu, and Min Sun. Tactics of adversarial attack on deep reinforcement learning agents. In *Proc. of International Joint Conference on Artificial Intelligence*, page 3756–3762, Melbourne, Australia, Aug. 2017.

[56] Fang Liu and Ness Shroff. Data poisoning attacks on stochastic bandits. In *Proc. of International Conference on Machine Learning*, pages 4042–4050, Long Beach, CA, Aug. 2019.

[57] Guanin Liu, Zhihan Zhou, Han Liu, and Lifeng Lai. Efficient action robust reinforcement learning with probabilistic policy execution uncertainty. *arXiv preprint arXiv:2307.07666*, 2023.

[58] Guanlin Liu and Lifeng Lai. Action-manipulation attacks against stochastic bandits: Attacks and defense. *IEEE Transactions on Signal Processing*, 68:5152–5165, 2020.

[59] Guanlin Liu and Lifeng Lai. Action-manipulation attacks on stochastic bandits. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3112–3116, May 2020.

[60] Guanlin Liu and Lifeng Lai. Provably efficient black-box action poisoning attacks against reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 34, pages 12400–12410, Dec. 2021.

[61] Guanlin Liu and Lifeng Lai. Action poisoning attacks on linear contextual bandits. *Transactions on Machine Learning Research*, 2022.

[62] Guanlin Liu and Lifeng Lai. Efficient adversarial attacks on online multi-agent reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 36, New Orleans, LA, Dec. 2023.

[63] Qinghua Liu, Tiancheng Yu, Yu Bai, and Chi Jin. A sharp analysis of model-based reinforcement learning with self-play. In *Proc. of International Conference on Machine Learning*, pages 7001–7010, Aug. 2021.

[64] Xiao-Yang Liu, Hongyang Yang, Qian Chen, Runjia Zhang, Liuqing Yang, Bowen Xiao, and Christina Dan Wang. Finrl: A deep reinforcement learning library for automated stock trading in quantitative finance. *Deep RL Workshop, NeurIPS 2020*, Dec. 2020.

[65] Thodoris Lykouris, Vahab Mirrokni, and Renato Paes Leme. Stochastic bandits robust to adversarial corruptions. In *Proc. of Annual ACM SIGACT Symposium on Theory of Computing*, pages 114–122, Los Angeles, CA, Jun. 2018.

[66] Thodoris Lykouris, Max Simchowitz, Alex Slivkins, and Wen Sun. Corruption-robust exploration in episodic reinforcement learning. In *Conference on Learning Theory*, pages 3242–3245, Boulder, CO, Aug. 2021.

[67] Yuzhe Ma, Kwang-Sung Jun, Lihong Li, and Xiaojin Zhu. Data poisoning attacks in contextual bandits. In *International Conference on Decision and Game Theory for Security*, pages 186–204, Seattle, WA, Oct. 2018.

[68] Yuzhe Ma, Young Wu, and Xiaojin Zhu. Game redesign in no-regret game playing. In *Proc. of International Joint Conference on Artificial Intelligence*, pages 3321–3327, Vienna, Austria, Jul. 2022.

[69] Yuzhe Ma, Xuezhou Zhang, Wen Sun, and Jerry Zhu. Policy poisoning in batch reinforcement learning and control. In *Advances in Neural Information Processing Systems*, volume 32, Vancouver, Canada, Dec. 2019.

[70] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, Vancouver, Canada, Apr. 2018.

[71] Weichao Mao and Tamer Başar. Provably efficient reinforcement learning in decentralized general-sum markov games. *Dynamic Games and Applications*, pages 1–22, 2022.

[72] Weichao Mao, Kaiqing Zhang, Ruihao Zhu, David Simchi-Levi, and Tamer Basar. Near-optimal model-free reinforcement learning in non-stationary episodic mdps. In *Proc. of International Conference on Machine Learning*, volume 139, pages 7447–7458, Jul. 2021.

[73] Andreas Maurer and Massimiliano Pontil. Empirical bernstein bounds and sample-variance penalization. In *Annual Conference Computational Learning Theory*, Montréal, Canada, Jun. 2009.

[74] S. Mei and X. Zhu. Using machine teaching to identify optimal training-set attacks on machine learners. In *Proc. of AAAI Conference on Artificial Intelligence*, pages 2871–2877, Austin, TX, Jan. 2015.

[75] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi, and Pascal Frossard. Universal adversarial perturbations. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1765–1773, Honolulu, HI, Jul. 2017.

[76] MohammadReza Nazari, Afshin Oroojlooy, Lawrence Snyder, and Martin Takac. Reinforcement learning for solving the vehicle routing problem. In *Advances in Neural Information Processing Systems*, volume 31, Montréal, Canada, Dec. 2018.

[77] Arnab Nilim and Laurent El Ghaoui. Robust control of markov decision processes with uncertain transition matrices. *Operations Research*, 53(5):780–798, 2005.

[78] Matthew O' Kelly, Aman Sinha, Hongseok Namkoong, Russ Tedrake, and John C Duchi. Scalable end-to-end autonomous vehicle testing via rare-event simulation. In *Advances in Neural Information Processing Systems*, volume 31, Montréal, Canada, Dec. 2018.

[79] Kishan Panaganti and Dileep Kalathil. Sample complexity of robust reinforcement learning with a generative model. In *Proc. of International Conference on Artificial Intelligence and Statistics*, pages 9582–9602, Mar. 2022.

[80] Rafael Pinot, Laurent Meunier, Alexandre Araujo, Hisashi Kashima, Florian Yger, Cedric Gouy-Pailler, and Jamal Atif. Theoretical evidence for adversarial robustness through randomization. In *Advances in Neural Information Processing Systems*, volume 32, Vancouver, Canada, Dec. 2019.

[81] Lerrel Pinto, James Davidson, Rahul Sukthankar, and Abhinav Gupta. Robust adversarial reinforcement learning. In *Proc.of International Conference on Machine Learning*, pages 2817–2826, Sydney, Australia, Aug. 2017.

[82] Doina Precup. Eligibility traces for off-policy policy evaluation. In *Proc. of International Conference on Machine Learning*, page 759–766, San Francisco, CA, Jun. 2000.

[83] Amin Rakhsha, Goran Radanovic, Rati Devidze, Xiaojin Zhu, and Adish Singla. Policy teaching via environment poisoning: Training-time adversarial attacks against reinforcement learning. In *Proc. of International Conference on Machine Learning*, pages 7974–7984, Jul. 2020.

[84] Amin Rakhsha, Xuezhou Zhang, Xiaojin Zhu, and Adish Singla. Reward poisoning in reinforcement learning: Attacks against unknown learners in unknown environments. *arXiv preprint arXiv:2102.08492*, 2021.

[85] Anshuka Rangi, Haifeng Xu, Long Tran-Thanh, and Massimo Franceschetti. Understanding the limits of poisoning attacks in episodic reinforcement learning. In *Proc. of International Joint Conference on Artificial Intelligence*, pages 3394–3400, Vienna, Austria, Jul. 2022.

[86] Lloyd S Shapley. Stochastic games. *Proceedings of the national academy of sciences*, 39(10):1095–1100, 1953.

[87] C. Shen. Universal best arm identification. *IEEE Transactions on Signal Processing*, 67(17):4464–4478, Sep. 2019.

[88] Laixi Shi and Yuejie Chi. Distributionally robust model-based offline reinforcement learning with near-optimal sample complexity. *arXiv preprint arXiv:2208.05767*, 2022.

[89] Aaron Sidford, Mengdi Wang, Lin Yang, and Yinyu Ye. Solving discounted stochastic two-player games with near-optimal time and sample complexity. In *Proc. of International Conference on Artificial Intelligence and Statistics*, pages 2992–3002, Aug. 2020.

[90] Max Simchowitz and Kevin G Jamieson. Non-asymptotic gap-dependent regret bounds for tabular mdps. In *Advances in Neural Information Processing Systems*, volume 32, Vancouver, Canada, Dec. 2019.

[91] Alexander L Strehl, Lihong Li, Eric Wiewiora, John Langford, and Michael L Littman. Pac model-free reinforcement learning. In *Proc. of International Conference on Machine Learning*, pages 881–888, Pittsburgh, PA, Jun. 2006.

[92] Yanchao Sun, Da Huo, and Furong Huang. Vulnerability-aware poisoning mechanism for online rl with unknown dynamics. In *International Conference on Learning Representations*, Vienna, Austria, May 2021.

[93] Yanchao Sun, Ruijie Zheng, Yongyuan Liang, and Furong Huang. Who is the strongest enemy? towards optimal and efficient evasion attacks in deep RL. In *International Conference on Learning Representations*, Apr. 2022.

[94] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

[95] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *International Conference on Learning Representations*, Banff, Canada, Apr. 2014.

[96] Aviv Tamar, Shie Mannor, and Huan Xu. Scaling up robust mdps using function approximation. In *Proc. of International Conference on Machine Learning*, volume 32, pages 181–189, Beijing, China, Jun. 2014.

[97] Chen Tessler, Yonathan Efroni, and Shie Mannor. Action robust reinforcement learning and applications in continuous control. In *Proc. of International Conference on Machine Learning*, pages 6215–6224, Atlanta, GA, Jun. 2019.

[98] Philip Thomas, Georgios Theocharous, and Mohammad Ghavamzadeh. High-confidence off-policy evaluation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29, Austin, TX, Jan. 2015.

[99] Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5026–5033, Vilamoura, Portugal, Oct. 2012.

[100] Joel Tropp et al. Freedman's inequality for matrix martingales. *Electronic Communications in Probability*, 16:262–270, 2011.

[101] Yisen Wang, Xingjun Ma, James Bailey, Jinfeng Yi, Bowen Zhou, and Quanquan Gu. On the convergence and robustness of adversarial training. In *Proc. of International Conference on Machine Learning*, volume 97, pages 6586–6595, Long Beach, CA, Jun. 2019.

[102] Yizhen Wang, Somesh Jha, and Kamalika Chaudhuri. Analyzing the robustness of nearest neighbors to adversarial examples. In *Proc. of International Conference on Machine Learning*, pages 5133–5142, Stockholm Sweden, Jul. 2018.

[103] Yue Wang and Shaofeng Zou. Online robust reinforcement learning with model uncertainty. In *Advances in Neural Information Processing Systems*, volume 34, pages 7193–7206, Dec. 2021.

[104] Chen-Yu Wei, Christoph Dann, and Julian Zimmert. A model selection approach for corruption robust reinforcement learning. In *International Conference on Algorithmic Learning Theory*, pages 1043–1096, Mar. 2022.

[105] Tsachy Weissman, Erik Ordentlich, Gadiel Seroussi, Sergio Verdu, and Marcelo J Weinberger. Inequalities for the l1 deviation of the empirical distribution. *Hewlett-Packard Labs, Tech. Rep*, 2003.

[106] Tianhao Wu, Yunchang Yang, Simon Du, and Liwei Wang. On reinforcement learning with adversarial corruption and its application to block mdp. In *Proc. of International Conference on Machine Learning*, pages 11296–11306. PMLR, Jul. 2021.

[107] Young Wu, Jeremy McMahan, Xiaojin Zhu, and Qiaomin Xie. Reward-poisoning attacks on offline multi-agent reinforcement learning. In *Proc. of AAAI Conference on Artificial Intelligence*, volume 37, pages 10426–10434, Washington DC, Jun. 2023.

[108] H. Xiao, B. Biggio, G. Brown, G. Fumera, C. Eckert, and F. Roli. Is feature selection secure against training data poisoning? In *Proc. of International Conference on Machine Learning*, volume 37, pages 1689–1698, Lille, France, July 2015.

[109] Qiaomin Xie, Yudong Chen, Zhaoran Wang, and Zhuoran Yang. Learning zero-sum simultaneous-move markov games using function approximation and correlated equilibrium. In *Conference on learning theory*, pages 3674–3682, Graz, Austria, Jul. 2020.

[110] Zaiyan Xu, Kishan Panaganti, and Dileep Kalathil. Improved sample complexity bounds for distributionally robust reinforcement learning. In *Proc. of International Conference on Artificial Intelligence and Statistics*, pages 9728–9754, Valencia, Spain, Apr. 2023.

[111] Kunhe Yang, Lin Yang, and Simon Du. Q-learning with logarithmic regret. In *Proc. of International Conference on Artificial Intelligence and Statistics*, pages 1576–1584, Apr 2021.

[112] Huan Zhang, Hongge Chen, Chaowei Xiao, Bo Li, Mingyan Liu, Duane Boning, and Cho-Jui Hsieh. Robust deep reinforcement learning against adversarial perturbations on state observations. In *Advances in Neural Information Processing Systems*, volume 33, pages 21024–21037, Dec. 2020.

[113] Kaiqing Zhang, Sham Kakade, Tamer Basar, and Lin Yang. Model-based multi-agent rl in zero-sum markov games with near-optimal sample complexity. In *Advances in Neural Information Processing Systems*, volume 33, pages 1166–1178, Dec. 2020.

[114] Tianyuan Zhang and Zhanxing Zhu. Interpreting adversarially trained convolutional neural networks. In *Proc. of International Conference on Machine Learning*, pages 7502–7511, Long Beach, CA, Jun. 2019.

[115] Xuezhou Zhang, Yiding Chen, Xiaojin Zhu, and Wen Sun. Robust policy gradient against strong data corruption. In *Proc. of International Conference on Machine Learning*, pages 12391–12401, Jul. 2021.

[116] Xuezhou Zhang, Yuzhe Ma, Adish Singla, and Xiaojin Zhu. Adaptive reward-poisoning attacks against reinforcement learning. In *Proc. of International Conference on Machine Learning*, volume 119, pages 11225–11234, Jul. 2020.

[117] Zihan Zhang, Xiangyang Ji, and Simon Du. Is reinforcement learning more difficult than bandits? a near-optimal algorithm escaping the curse of horizon. In *Conference on Learning Theory*, pages 4528–4531, Boulder, CO, Aug. 2021.

[118] Zihan Zhang, Yuan Zhou, and Xiangyang Ji. Almost optimal model-free reinforcement learningvia reference-advantage decomposition. In *Advances in Neural Information Processing Systems*, volume 33, pages 15198–15207, Dec. 2020.

[119] Xiangyu Zhao, Long Xia, Liang Zhang, Zhuoye Ding, Dawei Yin, and Jiliang Tang. Deep reinforcement learning for page-wise recommendations. In *Proc. of ACM Conference on Recommender Systems*, page 95–103, Vancouver, Canada, Oct. 2018.