Multi-Armed Bandits With Combinatorial Strategies Under Stochastic Bandits

Xiang-Yang Li, Senior Member, IEEE, Yaqin Zhou

Abstract—We consider the following linearly combinatorial multi-armed bandits (MABs) problem. In a discrete time system, there are \( K \) unknown random variables (RVs), i.e., arms, each evolving as an i.i.d stochastic process over time. At each time slot, we select a set of \( N (N < K) \) RVs, i.e., strategy, subject to an arbitrarily constraint. We then gain a reward that is a linear combination of observations on selected RVs. Our goal is to minimize the regret, defined as the difference between the summed reward obtained by an optimal static policy that knew the mean of each RV, and that obtained by a specified learning policy that does not know. A prior result for this problem has achieved zero regret (the expect of regret over time approaches zero when time goes to infinity), but dependent on probability distribution of strategies generated by the learning policy. The regret becomes arbitrarily large if the difference between the reward of the best and second best strategy approaches zero. Meanwhile, when there are exponential number of combinations, naive extension of a prior distribution-free policy would cause poor performance in terms of regret, computation and space complexity. We propose an efficient Distribution-Free Learning (DFL) policy that achieves zero regret without dependence on probability distribution of strategies. Our learning policy only requires time and space complexity \( O(K) \). When the linear combination is involved with NP-hard problems, our policy provides a flexible scheme to choose possible approximation algorithms to solve the problem efficiently while retaining zero regret.

Index Terms—Multi-armed bandits, online learning, combinatorial strategy.

I. INTRODUCTION

A multi-armed bandits problem is a basic sequential decision problem defined by a set of strategies against multiple unknown random variables. In the simplest form of MAB problems, i.e., single play, a strategy consists of one random variable. In the multi-play version, a strategy involves a combination of more than one random variables. At each time step, a decision maker selects a strategy, and then obtains an observable reward. The decision maker learns to maximize the total reward obtained in a sequence of decisions through history observation. MAB problems naturally capture the fundamental tradeoff between exploration and exploitation in sequential experiments. That is, the decision maker must exploit strategies that did well in the past on one hand, and explore strategies that might have higher gain on the other hand. MAB problems now play an important role in online computation under unknown environment, such as pricing and bidding in electronic commerce [3] [4]. Ad placement on web pages, source routing in dynamic networks, and opportunistic channel accessing in cognitive radio networks [5] [6]. Depending on the assumed nature of the reward process, MAB problems fall into three fundamental categories: stochastic, adversarial, and Markovian. In this paper, we focus on stochastic bandits.

Despite of many existing results on multi-play MAB problems against unknown stochastic environment [7] [8] [9], their adopted formulations block a broader application to deal with practical combinatorial optimization problems that usually include numerous or even exponential strategies. That is because their formulations ignore constraints among selected random variables, unsuited to practical applications that impose constraints on selected random variables. Especially in many domains, e.g., networking and communication, most related combinatorial problems are NP-hard, where aforementioned approaches would be confronted with inefficiency or even failures. Thus here we explore a more general formulation for constrained combinatorial bandit problems. Given \( K \) unknown random variables that are i.i.d over time, a strategy that consists of at most \( N \) random variables is selected under some general constraints at each time slot; all elements of this selected strategy are observed after decision, and the corresponding reward is a linear combination of these observed values. The objective is to maximize the total reward over time.

Recently Gai et.al [11] have also studied a similar formulation of linearly combinatorial MAB problem to ours, and proposed a policy called Learning with Linear Rewards (LLR) that has good performance in terms of regret as well as computation and space complexity. However, the upper bound on the expected regret of the proposed policy includes the term of \( \Delta_{\text{min}} \), the minimum distance between the best static strategy and any other strategy. We say such a bound is distribution-dependent [10], as the smaller \( \Delta_{\text{min}} \) is, the more time it takes to spot the best strategy. According to the expression of regret, its upper bound trends to infinity as \( \Delta_{\text{min}} \) trends to zero. The policy then becomes invalid if there are more than one best strategies, which is quite common in practice. Moreover, the premise of unknown environments indicates that we are usually unaware of the probability distribution of strategies. We also find that it is impossible to prove a good distribution-free result for the LLR policy as the regret grows faster than linearity of time.

Naturally, we then expect to design a distribution-free learning
policy that has zero regret for the linearly combinatorial MAB problem subject to arbitrary constraints. More specifically, the upper bound on regret is a supremum taken over all possible strategies of probability distribution on $[0, 1]$. Though there is a dependent-free policy (MOSS) for the simplest multi-armed bandits with single play, the achieved regret as well as computation and space complexity all grows linearly with the square root of the number of arms. There will be exponential number of strategies if one simply treats each strategy as an arm, which causes quite poor performance.

Meanwhile, we expect a good learning policy not only sound in theory, but also efficient in practical. In the case of numerous strategies, an efficient policy shall cost low overhead to make strategy decisions. Moreover, when the combinatorial optimization is NP-hard, the policy shall be robust enough to admit approximation algorithms to facilitate the learning process. Specifically, the resulting upper bound on regret shall be independent on all approximation algorithms once the approximation ratio is given. As different approximation algorithms with the same approximation ratio may have distinct advantages of practical interest, but possibly generate varying strategy sets that determine the regret. It would provide more flexible applications if the supremum regret taken over all possible approximation algorithms is sublinear with time.

Motivated by the above two objectives, we present a novel learning policy, named DFL, which requires only time and space complexity $O(K)$ to achieve zero regret that is distribution-free. When the underlying combinatorial optimization is NP-hard, our proposed approximation learning policy can also achieve zero regret through approximation algorithms. The results retain distribution-free once the approximation ratio is fixed. Additionally, we prove results on distribution dependency for both cases where optimal solutions or suboptimal solutions are available for the underlying combinatorial problem. Typical applications of the formulation and our proposed policies are discussed, including auction, shortest path and dynamic channel accessing problems. For those applications involving NP-hard problems, our analysis and results on approximation solutions enable flexible and efficient implementation of our proposed policy in practice.

The remainder of this paper is organized as follows. We first give a formal description of the linearly combinatorial multi-armed bandits problem in Section II. We present and analyze our new policy DFL in Section III. In Section IV, we give special analysis of our learning policy for the NP-hard combinatorial optimizations. In Section V, we present some applications of our policy. We evaluate our policy with the application of spectrum sensing in Section VI. We review related works in Section VII and conclude our work in Section VIII.

II. PROBLEM FORMULATION

We consider a time slotted system with $K$ arms/unknown random variables $\xi_k(t), 1 \leq k \leq K$, where $t$ is index of time slot. We assume that each of the $K$ variables evolves as an i.i.d stochastic process $\xi_k(t)$ normalized between $[0, 1]$ over time with mean $\mu_k$, which is unknown a prior. Table I summarizes the notations used in this paper.

At each time slot $t$, an $N$-dimensional strategy vector $s_x = \{s_{x,i} | i = 1, \ldots, N\}$ is selected under some policy from the feasible strategy set $F$. By feasible we mean that each strategy satisfies the underlying constraints imposed to $F$. Here $s_{x,i}$ is the index of random variables selected as the $i$th element of strategy $s_x$. We use $x = 1, \ldots, X$ to index strategies of feasible set $F$ in the decreasing order of average reward $\lambda_x = \sum_{i=1}^N \mu_{x,i}$. Note here we do not restrict each strategy to an exact length of $N$. A strategy may consist of less than $N$ random variables, as long as it satisfies the constraints. For that case we set $s_{x,i} = 0$ for the element without selection. Please also note that the uniformly linear combination of random variables in a strategy includes the weighted case, as we can easily take the product of each arm and its weight as a new random variable and normalize the new random variable to $[0, 1]$. If the unknown means were known, the static optimal strategy would be,$s_1 = \arg\max_{s_x \in F} \sum_{i=1}^N \mu_{x,i}$.s.t. $F$ is feasible strategy set. (1)

When a strategy $s_x$ is determined, one observes the value of $\xi_{s_{x,i}}(t)$, and then the total reward of strategy $s_x$ at $t$ is,

$$R_x(t) = \sum_{s_{x,i} \in s_x} \xi_{s_{x,i}}(t).$$ (2)

We evaluate policies using regret, which is defined as the difference between the expected reward obtained by an optimal strategy, and the expected reward obtained by the given policy. Let $R_1 \lambda_1$ be the optimum, then regret of a strategy $s_x$ over $n$ time slots can be expressed as

$$\mathcal{R}(n) = nR_1 - E \left[ \sum_{t=1}^n R_x(t) \right] = \sum_{x : R_x < R_1} \Delta_x E[T_x(n)]$$ (3)

As these random variables are unknown, and observed after decision, we have to learn to estimate the reward of each strategy. We denote the estimated value of strategy $s_x$ at time slot $t$ by weight $W_x(t) = \sum_{s_{x,i} \in s_x} \xi_{s_{x,i}}(t)$, where weight $w_{s_{x,i}}(t)$ is estimated value of random variable $\xi_{s_{x,i}}(t)$.

For the case of finding the best strategy in $F$ is NP-hard, we define a weaker version of regret called $\beta$-regret, which is the difference between the expected reward that is $1/\beta$ of the optimum, and that obtained by an $\beta$-approximation policy which instead yields a strategy with learned weight at least $1/\beta$ of the maximum possible weight among all strategies. Define the set of strategies generated by the $\beta$-approximation policy as $F_\beta = \{s_{\beta,x} | \lambda_{s_{\beta,x}} \geq \lambda_1/\beta \} \subset F$. Let $R_{\beta,x}(t)$ be the reward of strategy $s_{\beta,x}$ generated by the $\beta$-approximation policy, the $\beta$-regret can be expressed as

$$\mathcal{R}_\beta(n) = nR_1/\beta - E \left[ \sum_{t=1}^n R_{\beta,x}(t) \right]$$ (4)

$$= \sum_{s_{\beta,x} : R_{\beta,x} < R_1/\beta} \Delta_{\beta,x} E[T_{\beta,x}(n)]$$ (5)

$$+ \sum_{s_{\beta,x} : R_{\beta,x} \geq R_1/\beta} \Delta_{\beta,x} E[T_{\beta,x}(n)]$$ (6)

$$\leq \sum_{s_{\beta,x} : R_{\beta,x} < R_1/\beta} \Delta_{\beta,x} E[T_{\beta,x}(n)]$$ (7)
TABLE I: Summary of notations

| Variable | meaning |
|----------|---------|
| \( K \) | number of arms/random variables |
| \( \xi_k \) | random variable (i.e., arm) with index \( k \) |
| \( \mu_k \) | mean of \( \xi_k \) |
| \( \tilde{\mu}_k \) | observed mean of \( \xi_k \) up to current time slot |
| \( m_k \) | number of times arm \( \xi_k \) has been observed so far |
| \( s_x \) | the \( x^{th} \) strategy in set \( F \), \( s_1 \) is the optimal strategy. |
| \( X \) | the maximum index of strategy in set \( F \) |
| \( N \) | length of strategy vector \( s_x \), \( N \leq \max \) |
| \( \lambda_x \) | mean reward achieved by \( s_x \) |
| \( \Delta_x \) | = \( \lambda_1 - \lambda_x \), the distance between \( s_1 \) and \( s_x \) |
| \( \Delta_{\min} \) | number of times strategy \( s_{\min} \) has been played by time slot \( n \) |
| \( \Delta_{\max} \) | number of times strategy \( s_{\max} \) has been played by time slot \( n \) |
| \( T_{\beta,x}(n) \) | weight (estimated reward) of strategy \( s_x \) at time slot \( n \) |
| \( W_{s_x}(n) \) | number of times strategy \( s_x \) has been played by time slot \( n \) |
| \( Z_x \) | index of the worst strategy \( s_{\min} \) with \( \lambda_{\min} \geq \lambda_1 / \beta \) |
| \( \Delta_{\min} \) | number of times that strategy \( s_{\min} \) has been played by time slot \( n \) |
| \( \Delta_{\max} \) | number of times strategy \( s_{\max} \) has been played by time slot \( n \) |

where \( T_{\beta,x}(n) \) is the number of times that strategy \( s_{\beta,x} \) has been played by time slot \( n \), and \( \Delta_{\beta,x} \) is the distance between \( R_1 / \beta \) and mean reward of strategy \( s_x \). Here in feasible strategy set of an \( \beta \)-approximation policy, strategies can be divided into two sets, i.e., a set of \( \beta \)-approximation strategies and a set of zero-\( \beta \)-approximation strategies. A \( \beta \)-approximation strategy is a strategy with mean reward of at least \( R_1 / \beta \), and a non-\( \beta \)-approximation strategy is one with mean reward less than \( R_1 / \beta \). Thus we have negative \( \Delta_{\beta,x} \) for \( \beta \)-approximation strategies and positive \( \Delta_{\beta,x} \) for non-\( \beta \)-approximation strategies.

In both cases, we expect regret \( \mathcal{R}(n) \) (or \( \mathcal{R}_3(n) \)) to be as small as possible. Intuitively, if the regret is \( o(n) \), sublinear with time \( n \), then the time averaged regret will approach 0, indicating time averaged reward to be maximum. Though previous learning policies are zero-regret, the proved regret heavily depends on the distribution of strategies in feasible set \( F \) (or \( F_3 \)). That is, the upper bound of regret \( \mathcal{R}(n) \) (or \( \beta \)-regret \( \mathcal{R}_3(n) \)) including a factor of \( \frac{1}{\Delta_{\min}} \) (or \( \frac{1}{\Delta_{3,\min}} \)) that becomes vacuous if \( \Delta_{\min} \) (or \( \Delta_{3,\min} \)) \( \to 0 \). We expect to design a zero-regret policy without dependency on \( \Delta_{\min} \) or \( \Delta_{3,\min} \).

III. DISTRIBUTION-FREE LEARNING POLICY

A naive method for a distribution-free policy of our combinatorial NP-hard MAB problem is to treat each strategy \( s_x \in F \) as an arm, by which we can directly use the MOSS policy achieving regret without \( \Delta_{\min} \).

Theorem 1: [2] MOSS satisfies \( \sup \mathcal{R}(n) \leq 49 \sqrt{nK} \), where the supremum is taken over all \( \kappa \)-tuple of probability distributions on \([0, 1] \).

Here \( \kappa \) is actually the number of strategies available (i.e., \( \kappa \approx \Theta(K^N) \) for combinatorial strategies) as MOSS is proposed for single-play bandit. MOSS yields regret growing linearly with the square root of the number of strategies, which is inefficient when the feasible strategy set \( F \) has exponential number of unknown strategies. Meanwhile, it requires high cost of \( \Theta(K^N) \) in computation and space to update and store observed information of all strategies. Consequently, the MOSS policy will perform poorly on our problem in terms of regret, computation and space complexity. When the combinatorial problem is NP-hard, it does not admit efficient approximation algorithms on strategy decision as well.

We now present a novel policy DFL that is a distribution-free zero-regret learning policy for combinatorial strategies (described in Algorithm 1) with low cost to store and update observed information by exploiting dependencies among correlated strategies.

Algorithm 1 Learning policy DFL

1: For each round \( t = 0, 1, \ldots, n \) Select a strategy \( s_x \) by maximizing \( \max_{s_x \in F} \sum_{x_{-1} \in s_x} (\tilde{\mu}_{s_x,t}(t) + \sqrt{\frac{\max(\ln n^{2/3}, 0)}{m_{s_x,t}}}) \) (8)

For brevity, let weight \( w_{s_x,t}(t + 1) = \tilde{\mu}_{s_x,t}(t) + \sqrt{\frac{\max(\ln n^{2/3}, 0)}{m_{s_x,t}}} \) (9) be estimated reward of \( \xi_{s_x,t}(t + 1) \) and weight \( W_x(t + 1) = \sum_{s_{-1} \in s_x} w_{s_x,t}(t + 1) \) (10) denote estimated reward of strategy \( s_x \). As shown in Algorithm 1, our proposed learning policy requires storage linear with \( K \) to update observed reward.

Theorem 2: Algorithm 1 has time and space complexity of \( O(K) \), even though the number of strategies may grow exponentially to \( \Theta(K^N) \).

Here we have assumed that we can instantly find a strategy with maximum reward in (8). We will give more computation efficient algorithm that also achieves zero-regret later in the paper. Bellow we give results and analysis on regret of Algorithm 1.

Theorem 3: The regret of policy DFL satisfies \( \sup \mathcal{R}(n) \leq NK + \left( \sqrt{eK} + 8(1 + N^3)N^\frac{2}{3} \right)n^\frac{2}{3} + (1 + \frac{4\sqrt{K}N^2}{e})N^2Kn^\frac{2}{3} \) (11) without dependency on \( \Delta_{\min} \). The supremum is taken over all \( X \)-tuple of probability distributions on \([0, 1] \).

Proof: See Appendix A for its proof.

Remark: By careful calculation of \( \max(\ln n^{2/3}, 0) \) with the term \( n^a (a < 1) \) to replace the term \( n^{2/3} \), we can get an upper
bound slightly better than presented results in Theorem 3. The power of $n$ in (11) is between $\left(\frac{2}{3} \cdot \frac{5}{6}\right]$.

The regret bound with dependency on $\Delta_{\text{min}}$ is presented in the theorem below.

**Theorem 4:** DFL has distribution-dependent regret

$$\mathcal{R}(n) \leq \left( e + (17 + 8.53N)N^3 + 8N^4 K n^{\frac{2}{3}} \right)^{K \max \left( \frac{n^2}{1 - 1/e}, 1 \right)}.$$

**Proof:** See Appendix B for its proof. $lacksquare$

**Remark:** On results with $\Delta_{\text{min}}$, we note that $\mathcal{R}(n)$ includes an order of $O(n^{\frac{2}{3}} \ln n)$, a bit worse than existing results with an order of $O(\ln n)$. The term $\frac{n^{\frac{2}{3}}}{1 - 1/e}$ is caused by the term of $\ln(n^{\frac{2}{3}})$ in (8). Actually we can replace the power $2/3$ by 1 and achieve bound without $n^{\frac{1}{2}}$ in proof of Theorem 4. But a power smaller than 1 is necessary for distribution-free result in Theorem 3 or the regret will grow to infinity.

IV. $\beta$-APPROXIMATION DISTRIBUTION-FREE LEARNING POLICY

As many problems in Expression (8) are NP-hard due to complex constraints imposed to the maximum problem, it is necessary to analyze the regret bound for the case of solving (8) with approximation algorithms. Without loss of generality, given an algorithm with approximation factor $\beta$ to solve problem in (8), the learning policy DFL becomes $\beta$-approximation policy DFL. We consider an upper bound of all $\beta$-approximation DFL policies. In that case we may have $\cup F_\beta \subseteq F$. Thus we drop superscript $\beta$ for $F_\beta$, $s_\beta, x$, $T_\beta, x$ according to the context.

**Theorem 5:** The $\beta$-approximation DFL policy satisfies

$$\sup \mathcal{R}_\beta(n) \leq \frac{1}{\beta} NK + \left( \sqrt{eK} + \frac{16}{e\beta} (1 + N)N^3 \right) n^{\frac{2}{3}}$$

$$+ \frac{1}{\beta} \left( 1 + \frac{4\sqrt{eK}N^2}{e\beta^2} \right) N^2 Kn^{\frac{2}{3}}$$

without dependency on $\Delta_{\beta, \text{min}}$. The supremum is taken over all $X$-tuple of probability distributions on $[0, 1]$.

**Proof:** See Appendix C. $lacksquare$

For the regret result with dependency on $\Delta_{\beta, \text{min}}$ of a given $\beta$-approximation DFL policy, we have

**Theorem 6:** The $\beta$-approximation DFL policy satisfies

$$\mathcal{R}_\beta(n) \leq \left( e/\beta + (17 + 8.53N)N^3 + 8N^4 K n^{\frac{2}{3}}/\beta^2 \right)^{K \max \left( \frac{n^2}{1 - 1/e}, 1 \right)}$$

$$+ \frac{N^2 K}{1 - 1/e} \frac{K \max \left( \frac{n^2}{\beta^2}N^2, 1 \right)}{\beta \Delta_{\beta, \text{min}}}.$$

**Proof:** See Appendix D. $lacksquare$

Based on Theorem 5 one can design efficient algorithms on strategy decision even though the number of strategies may grow exponentially. In unknown stochastic environment, many network optimization problems can be formulated as a linearly combinatorial MAB problem with a maximum objective function, e.g., the shortest path problem, matching problem, maximum weighted independent set of vertices problem and other practical problems in wireless communication. Regarding to these problems without deterministic polynomial solutions, our uniform results provide a flexible scheme to alternative approximation learning methods with bounded $\beta$-regret.

V. APPLICATIONS

In this section we describe some typical applications of our proposed learning policy for combinatorial optimization problems that may have exponential number of strategies.

A. Auctions

Recently, auctions have been employed in a variety of industries, such as truckload transportation, bidding for ad placement, as well as allocation of radio spectrum for wireless communications services and virtual machine instances for clouds.

1. Ad placement

We first start with a simple example in auctions, i.e., bidding for ad placements. Ad placement is the process of deciding which advertisement to display on the web page delivered to the next visitor of a website. Suppose there is a sequence of $N$ places on the web page for display of ads, and a pool of $K$ available ads (bandit arms), where $K \geq N$. The payoff of an ad depends on valid click-throughs or other visitors’ behaviors that are unknown and need to learn. The payoff of ad $k$ evolves as an unknown stochastic process $\xi_k$ with mean $\mu_k$. The provider of ad $k$ bids for a placement in the website. And the website wants to maximize its social welfare, summed payoff of the winners. The simplest form of the problem is that there are no constraints among different ads. In this case, the optimum solution is selecting the $N$-best ads among $K$ ones.

2. Virtual machine instance allocation
In the auction problem for virtual machine instance allocation, the service provider has $M$ types of virtual machines, each type with $a_j$ instances. A set of $K$ long-term tenants has a bundle of requested instances $B_i = \{b_{i,j} | j = 1, \ldots, M\}$ where $b_{i,j}$ is the number of instances required for type $j$ by user $i$. For user $i$, the valuation of his bundle is $v_i$. The service provider has to select a set $s_x$ of winners among the $K$ tenants to maximize its long-term social welfare

$$\max_{s_x \in F} \sum_{i \in s_x} v_i$$

$$s.t. \sum_{j \in s_x} b_{i,j} \leq a_j, \forall 1 \leq j \leq M$$

$$F$$ is a set of all possible $s_x$ satisfying (15).

Note the formulation of virtual machine allocation problem could be much more complicated if taking more practical constraints into consideration, e.g., energy constraint and migration of resources in physical machines. Even for simplest form of auctions for virtual machine allocation in (15), however, the maximum problem turns out to be a multidimensional knapsack problem that is a strongly NP-hard combinatorial optimization problem.

In practical applications, the valuation of a bundle may be unknown for the bidder. For example, if the tenants rent instances of virtual machine for deployment of websites, the payoff may be determined by viewer’s behaviors. In some computation-intensive applications, the payoff may be determined by unknown pure utility earned online. When valuation of each bundle evolves as an i.i.d stochastic process $\xi_i(t)$ over time, with bounded region but unknown mean, an online learning policy is necessary for the service provider.

**Algorithm 2 Learning policy for VM allocation**

1. For each time slot $t = 0, 1, \ldots, n$ Select a strategy $s_x$ by maximizing

$$\max_{s_x \in F, s_x, i \in s_x} \left( \tilde{\mu}_{s_x,i}(t) + \sqrt{\frac{\ln \frac{2}{\delta_x} K}{m_{s_x,i}}} \right)$$

Applying our $\beta$-approximation DFL learning policy, we can design an efficient learning policy with low time and space complexity. Here the valuation of user’s bundle corresponds to unknown random variables in our linearly combinatorial MAB formulation, and a set of winners corresponds to a strategy $s_x$ in $F$. The general framework is shown in Algorithm 2. There are numerous greedy benchmark algorithms or simple heuristics [11] [12] for the multidimensional knapsack problem in literature, one can flexibly select approximation algorithms or heuristics with good experimental performance to solve the multidimensional knapsack problem according to requirements of practical system design. As the special design of DFL policy, the regret trends to zero no matter how strategies in resulting feasible set by the selected algorithm distribute.

**B. The shortest path problem**

We now consider the shortest path problem that has polynomial computation time to find the deterministic optimization among exponential possible paths. Consider a network $G = (V, E)$ with a set $V$ of vertices connected by edges of $E$. A sequence of packets must be routed from a distinguished vertex, called source, to another distinguished vertex, called destination. At each time slot a packet is sent along a specific source-destination path by a routing protocol or a decision maker. Depending on the congestion of the edges in the chosen path, each edge in the network may have a different delay. The delays may change from one time slot to the next one. The goal is to find the route that the sum of the total delays is minimum among all possible paths. For this problem, we can look upon delay of each edge as a bandit. The shortest path problem involves a minimum problem that is the opposite of maximum problems in our paper. Thus we can transform it into a maximum problem by replacing the loss of delay with a gain that is defined as the difference between the maximum delay and observed delay.

Let delay of each edge be an i.i.d stochastic process $\xi_k(t)$ over time with mean $\mu_k$. For simplicity, we assume $\xi_k(t)$ is normalized to $[0, 1]$. Define $\vartheta_k(t) = 1 - \xi_k(t)$ with mean $1 - \mu_k$. We suppose each source-destination path $p_x \in F$ consists of a sequence of edges $\{p_{x,i} | p_{x,i} \leq |E|\}$ where $p_{x,i}$ is index of edges. Thus the solution to shortest path problem solves the following maximum problem actually,

$$\max_{p_x \in F} \sum_{p_{x,i} \in p_x} (1 - \mu_{p_{x,i}})$$

$$s.t. F$$ is a set of all source-destination paths. (17)

Taking $\vartheta_k(t)$ as unknown random variables, and $p_x$ as strategies, we instantly get the maximum reward version of combinatorial multi-armed bandit formulation. The modified DFL policy for the shortest path problem is shown in Algorithm 3. For the shortest path problem in (18) where estimation of delay on each edge is $\tilde{\mu}_{p_{x,i}}(t) + \sqrt{\frac{\ln \frac{2}{\delta_x} K}{m_{p_{x,i}}}}$, there exist efficient implementations of these classical solutions (i.e., Dijkstra’s algorithm [13] [14] and Bellman-Ford algorithm [15]).

**Algorithm 3 Learning policy for shortest path problem**

1. For each time slot $t = 0, 1, \ldots, n$ Select a path $p_x$ by minimizing

$$\min_{p_x \in F} \sum_{p_{x,i} \in p_x} \left( \tilde{\mu}_{p_{x,i}}(t) + \sqrt{\frac{\ln \frac{2}{\delta_x} K}{m_{p_{x,i}}}} \right)$$

(18)
C. Dynamic channel accessing in multi-hop cognitive radio networks

Here we consider the dynamic channel accessing problem in multi-hop cognitive radio networks. Given a cognitive radio network described by conflict graph $G = (V, E, C)$ with a set $V = \{v_i | i = 1, \ldots, N\}$ of $N$ users, a set $E$ of edges, and a set $C = \{c_j | j = 1, \ldots, M\}$ of $M$ channels. Conflicts happen if any two adjacent users access the same channel simultaneously. At each time slot $t$, user $v_i$ has $M$ choices of channels, each having data rate drawn from i.i.d stochastic process $\xi_{i,j}(t)$ over time with an unknown mean $\mu_{i,j} \in [0, 1]$. Without loss of generality, we assume the same channel may demonstrate different channel qualities for different users. For the same channel $c_j$, the random process $\xi_{i,j}(t)$ is independent from $\xi_{i',j}(t)$ if $i \neq i'$. The objective of the dynamic channel accessing problem is to find an optimal allocation of channels for users so that the time averaged throughput of network is maximized.

We then show how the dynamic channel accessing problem can be formulated into a networked multi-armed bandit problem. We remodel the network conflict graph $G$ as an extended conflict graph $H$, and show that the problem can be reformulated as the maximum weighted independent set of vertexes in extended conflict graph $H$. Define virtual nodes $v_{i,j}$, $j = 1, \ldots, M$ for each user $v_i$, and connect $v_{i,j}$ with $v_{i,k}$ ($j \neq k$) for all $j, k$. We also connect $v_{i,j}$ with $v_{p,j}$ if $v_i$ and $v_p$ has an edge in original network $G$. Then we get a new graph $H$ with $MN$ nodes. We give an illustration of this procedure in Fig. 2 where the original conflict graph $G$ has $M = 2$ available channels for each of $N = 4$ user. The feasible strategy set $F$ consists of all maximal independent set (MIS) of nodes in $H$. Here note that the cardinality of MIS is less than $N$ if the chromatic number of $G$ is greater than $M$, and is $N$ otherwise. Let $\xi_{i,j}(t)$ be weight of virtual node $v_{i,j}$. If the mean of $\xi_{i,j}(t)$ is known, the optimum strategy is to find a maximum weighted independent set of nodes among $K = MN$ nodes of $H$ as choices selected by users in $G$, i.e.,

$$
\max_{s_x \in F} \sum_{i=1}^{N} \mu_{i,s_{x,i}}
$$

subject to $s_x$ is an independent set of vertexes in $H$, (19)

where $s_{x,i}$ is the index of channel selected by user $v_i$ in strategy $s_x$.

Similarly, the dynamic channel accessing policy in Algorithm 4 needs to find a strategy that has maximum estimated weight at each time slot, i.e., solving the problem of (20),

$$
\sum_{s_x : v_{s_x,i} \in H} (\tilde{\mu}_{s_x,i}(t) + \sqrt{\max (\frac{\ln \frac{2}{\delta}}{K m_{s_x,i}}, 0)})
$$

is estimated weight of virtual node $v_{s_x,i}$. As the involved MWIS problem is NP-hard, we can not directly use the DFL policy to solve (20). We then turn to $\beta$-approximation DFL policy to solve (20) with low complexity approximation algorithms for MWIS. For MWIS problem, there exist some simple PTAS that can be implemented in a distributed manner, such as robust PTAS in [16] and shifting approach in [17].

Herein the above applications discussed in this section give basic frameworks on bandits formulation of these problems, practical considerations may generate more complicated constraints on feasible sets $F$, which lead to even harder NP problems that have no existing efficient solutions. Additionally, many more details and implementation issues need to be addressed when applying our proposed policies to specific applications. For instance, in the application of dynamic channel accessing, it would be necessary to design a local or distributed implementation of our policy, involving consideration on low cost on strategy decision, as well as message collection and broadcast. These issues are not trivial, but of significance when putting our theoretical results into practice. It demands a careful tradeoff among theoretical guarantee, implementation manners, storage, computation and extra communication complexity as well as their potential impact on the actually achievable performance. Hence, combination of practical implementation with our proposed learning policy in specific domain especially demands more elegant design, which is also an interesting work.

VI. SIMULATION

In this section we present some simulation results for our learning policy, using the application to the dynamic channel accessing problem in Section V-C. We consider a small network with 5 users, each of which has 5 available channels. The conflict relationship is below,

$$
\begin{pmatrix}
1 & 1 & 1 & 1 & 0 \\
1 & 1 & 1 & 0 & 1 \\
1 & 1 & 1 & 1 & 0 \\
1 & 0 & 1 & 1 & 0 \\
0 & 1 & 0 & 0 & 1
\end{pmatrix}
$$

where an element $e_{i,j} = 1$ denotes conflict and $e_{i,j} = 0$ denotes independency between users $v_i$ and $v_j$. The average data rate

![Fig. 2: Original conflict graph G to extended conflict graph H](image-url)
on the 5 channels of each user is shown in the following matrix,

\[
\begin{pmatrix}
631.98 & 369.81 & 128.43 & 191.70 & 155.64 \\
432.00 & 53.93 & 598.08 & 30.93 & 551.52 \\
199.55 & 26.00 & 1175.17 & 524.34 & 147.69 \\
127.38 & 53.73 & 68.34 & 937.44 & 117.62 \\
311.04 & 101.28 & 171.95 & 436.45 & 62.19 \\
\end{pmatrix}
\]

where each row \(i\) denotes data rates of user \(v_i, i = 1, 2, 3, 4, 5\). The optimal static throughput of this network, i.e., the maximum possible weight of ISLs in the corresponding extended conflict graph is 3732.56.

Fig. 3 plots comparison of the time-averaged regret/\(\beta\)-regret by our proposed DFL policy and LLR policy. Fig. 3(a) shows that DFL policy requires much less time on learning for better strategies, thus produces much smaller regret. The time-averaged regret by DFL policy converges to 0 around time slot 400, while regret by LLR policy is more than 100. \(\beta\)-regret in Fig. 3(b) shows negative value, which indicates that the achievable throughput by the two learning algorithms is better than \(1/\beta\) of the optimal throughput when utilizing \(\beta\)-approximation algorithms to solve the NP-hard MWIS problem.

![Regret](image1)

(a) Regret

![\(\beta\)-regret](image2)

(b) \(\beta\)-regret

Fig. 3: Regret/\(\beta\)-regret: comparison with LLR learning policy

VII. Related Work

We now make a review on multi-armed bandits problems. Depending on the assumed reward process, we have stochastic, adversarial, and Markovian bandits. In the adversarial bandits, the reward of each arm is nonstochastic, and in the Markovian bandits, each arm is associated with a Markov process of its own state space. Regrading to each single-play bandit problem with a specific reward model of the three, we respectively have the following three classical learning policies: the UCB algorithm for stochastic case [18], the Exp3 for adversary case [19], and the Gittin’s indices for the Markovian case [20]. A throughout review on stochastic and nonstochastic bandit problems is available in [10], while a textbook by Gittin [21] for Markovian bandits. On the other hand, according to feedback of the observed information on random variables, the bandits problem can fall into categories of full information, semi-bandit, and full bandit. The decision maker observes value of all random variables in the case of full information, and value of these selected random variables in the case of semi-bandit. While in full bandit, only the instant reward of the selected strategy is fed back.

We mainly focus on stochastic bandits where value of selected random variables can be observed. The simplest form of bandits is single-play bandits where \(N = 1\) arm is selected among \(K\) ones. The analysis of the stochastic bandit is pioneered in the seminal paper of Lai and Robbins [18] where the UCB algorithm is proposed to solve the single play version. Many papers follow its basic idea to provide improved bounds on regret or simpler upper confidence bound policies for single-play version [22], or extend it to multi-play variants where \(N > 1\) arms are selected at a time. In [22], it proposes a simple sample-mean based method with regret logarithmic uniformly over time, and [23] presents variants of Agrawal’s work to achieve logarithmic regret in finite time.

All these aforementioned UCB-type policies are distribution-dependent. In [2], Audibert and Bubeck propose a learning policy called MOSS that has a distribution-free upper bound on regret with order of \(\sqrt{N}\). Our work is inspired by MOSS, but considers a more general formulation which includes a set of multiple arms that has to satisfy an arbitrarily given constraint. The MOSS policy is proposed to solve single-play bandits, and can not be directly used to solve multi-play version where the exact value of \(N\) may be even unknown. The policy will be highly inefficient if taking each combination of random variables as an arm, as it will cost exponential computation and storage in terms of \(K^N\). Furthermore, for the NP-hard combinatorial optimization problems, it is too expensive to find the best strategies by learning all possible strategies. Thus, our learning policy provides an efficient policy in regret, storage and computation for such problems.

For the variant with multi-play, Anantharam et al. [7] firstly consider the problem that exactly \(N\) arms are selected simultaneously. Gai et al. recently extend this version to a more general problem with arbitrary constraints [11]. The model is also relaxed to a linear combination of no more than \(N\) arms.
As we pointed out, the results are distribution-dependent. An arbitrary small $\Delta_{\text{min}}$ will invalidate the zero-regret result. In our work, we have analyzed both the cases, with or without dependent on $\Delta_{\text{min}}$. But we can not get similar results by analyzing the policy proposed in [1], as the policy is no longer zero-regret in the case of distribution free. Our leaning policy is thus more robust and general than [1].

In a very recent work, Chen et al. [24] study a similar combinatorial MAB problem that admit nonlinear reward function under two assumptions. The objective is to minimize a so-called $(\alpha, \beta)$-approximation regret, which is the difference in total expected reward between the $\alpha\beta$ fraction of the expected reward when always playing the optimal fixed arm, and the expected reward of the playing arms output by an assumed oracle that could compute an arm whose expected reward is at least $\alpha$ fraction of the optimum with probability $\beta$. The regret bound achieves distribution free for some reward functions if the two assumptions on the expected reward are satisfied, i.e., monotonicity and bound smoothness. Our work differs from theirs in several important aspects. First and for most, our regret analysis covers all forms of linear combinations without any assumption. Second, we analyze the regret bounds for both optimal solution and approximation solution for the NP combinatorial problems. Third, we discuss various applications to typical network optimization problems.

Some recent works [25] [9] have studied distributed learning among multiple users under the original multi-play model as in [7]. Though there is no communication overhead, both of the approaches basically require exponential time in a single learning round. While with communication among multiple users, Kalathil et al. [8] propose an online index-based learning policy that achieves nearly zero regret.

Recently, the bandits have attracted much attention from researchers in cognitive radio networks. This line of works starts from single-user play [6] [26], where each channel evolves as independent and identically distributed Markov processes with good or bad state. Due to distributed nature of wireless networks as well as limited computation, storage and energy of wireless nodes, efficient distributed implementation among multiple users then becomes the main focus of policy design [8], [25], [27], [28], [29]. These works basically assume channel quality evolving with i.i.d stochastic process over time, and a single-hop network setting where conflict happens if any pair of users choose the same channel simultaneously. Under nonstochastic channel quality, Li et al. [5] propose an throughput efficient allocation approach with central control. This approach only costs computation and space complexity $O(MN)$ by exploiting dependency among strategies.

VIII. Conclusion

In this paper we propose a distribution-free policy for the linearly combinatorial multi-armed bandits with general constraints that may cause exponential number of strategies. We have taken care of efficiency issues on storage, computation and practical applications. We expect that our works would broaden applications of multi-armed bandits in practice.

Though we have provided a distribution-free upper bound for the problem we considered in this paper, we find that the bound is kind of loose compared to some existing results that require high complexity in computation and storage. We conjecture that an upper bound with $O(\sqrt{n})$ may be available if we give tighter analysis and use more strict conditions in probabilistic analysis when counting the number of times that non-optimal or non-$\beta$-approximation strategies have been played. We leave this as a future work.

In our paper we have actually studied a simpler bandit model of stochastic rewards, compared to adversary or Markovian bandits. The assumption on i.i.d stochastic process has mitigated difficulties on concentration analysis through Hoeffding’s results. The problem becomes more challenging in the adversary case where we can not use these tools. For instance, many results with tight regret bounds of $\sqrt{nK}$ have been gained for linear combination of bandits in the adversary case in literature, but not yet computation and storage efficient. We expect to tackle this challenge in future works.

We also note that many works as well as ours have studied weak regret that is compared to a static optimal policy. It would be interesting to analyze models using strong regret that is compared to a dynamic optimal policy. In this case, one has to track the best dynamic policy through estimating the random process and computing the approximate optimal policy, instead of only estimating sample mean in static case. Similar with the case of weak regret, there would be challenges on reduction of time and space complexity, as well as distributed implementation issues among multi-users.

REFERENCES

[1] Y. Gai, B. Krishnamachari, and R. Jain, “Combinatorial network optimization with unknown variables: Multi-armed bandits with linear rewards and individual observations,” IEEE/ACM Transactions on Networking, vol. 20, no. 5, pp. 1466-1478, 2012.
[2] J.-Y. Audibert, S. Bubeck et al., “Minimax policies for adversarial and stochastic bandits,” in Proceedings of the 22th annual conference on Learning theory, 2009.
[3] M. Babaioff, S. Dughmi, R. Kleinberg, and A. Slivkins, “Dynamic pricing with limited supply,” in Proceedings of the 13th ACM Conference on Electronic Commerce, 2012, pp. 74–91.
[4] M. Babaioff, R. D. Kleinberg, and A. Slivkins, “Truthful mechanisms with implicit payment computation,” in Proceedings of the 11th ACM conference on Electronic Commerce, 2010, pp. 43–52.
[5] X.-Y. Li, P. Yang, Y. Yan, L. You, S. Tang, and Q. Huang, “Almost optimal accessing of nonstochastic channels in cognitive radio networks,” in Proc. IEEE Infocom, 2012, pp. 2291–2299.
[6] Q. Zhao, B. Krishnamachari, and K. Liu, “On myopic sensing for multi-channel opportunistic access: Structure, optimality, and performance,” IEEE Transactions on Wireless Communications, vol. 7, no. 12, pp. 5431–5440, 2008.
[7] V. Anantharam, P. Varaiya, and J. Walrand, “Asymptotically efficient allocation rules for the multiarmed bandit problem with multiple plays: I: iid rewards,” IEEE Transactions on Automatic Control, vol. 32, no. 11, pp. 968–976, 1987.
[8] D. Kalathil, N. Nuyyar, and R. Jain, “Decentralized learning for multi-player multi-armed bandits,” in Proc. IEEE CDC, 2012, pp. 3960–3965.
[9] C. Tekin and M. Liu, “Online learning in decentralized multiuser resource sharing problems,” arXiv preprint arXiv:1210.5544, 2012.
To prove the theorem, we will use Chernoff-Hoeffding bound and the maximal inequality by Hoeffding [30].

**Lemma 1:** (Chernoff-Hoeffding Bound [30]) \( \xi_1, \ldots, \xi_n \) are random variables within range \([0, 1]\), and \( E[\xi_i] = \xi_1, \ldots, \xi_{i-1} = \mu, \forall 1 \leq t \leq n \). Let \( S_n = \sum_{i=1}^{n} \xi_i \), then for all \( a > 0 \)

\[
P(S_n \geq n\mu + a) \leq \exp\left(-2a^2/n\right),
\]

\[
P(S_n \leq n\mu - a) \leq \exp\left(-2a^2/n\right).
\] (21)

**Lemma 2:** (Maximal inequality) \( \xi_1, \ldots, \xi_n \) are i.i.d random variables with expect \( \mu \), then for any \( y > 0 \) and \( n > 0 \),

\[
P\left(\exists t \in 1, \ldots, n, \sum_{t=1}^{r} (\mu - \xi_t) > y\right) < \exp\left(-\frac{2y^2}{n}\right).
\] (22)

Recall that we have assumed \( \lambda_1 \geq \cdots \geq \lambda_X \). As strategy \( s_1 \) is the optimal strategy, we have \( \Delta_x = \lambda_1 - \lambda_x \). We further define \( W_1 = \min_{1 \leq t \leq n} W_1(t) \). We may assume the first time slot \( z = \arg \min_{1 \leq t \leq n} W_1(t) \).

1. **Rewrite regret in terms of arms**

Separating the strategies in two sets by \( \Delta_x > \Delta_{x_0} \). We define \( x_0 \) later in the proof, we have

\[
\mathfrak{R}(n) = \sum_{x=1}^{x_0} \Delta_x E[T_x(n)] + \sum_{x=x_0+1}^{X} \Delta_x E[T_x(n)] \leq \Delta_{x_0} n + \sum_{x=x_0+1}^{X} \Delta_x E[T_x(n)].
\] (23)

We then analyze the second term of (23). As there may be exponential number of strategies, counting \( T_x(n) \) of each strategy by the traditional UCB based analysis yields regret growing linearly with the number of strategies. Note that each strategy consists of \( N \) arms at most, we can rewrite the regret in terms of arms instead of strategies. We then introduce a set of counters \( \{\tilde{T}_k(n)\} \). At each time slot, either 1) a strategy with \( \Delta_x \leq \Delta_{x_0} \) or 2) a strategy with \( \Delta_x > \Delta_{x_0} \) is played. In the first case, \( \tilde{T}_k(n) \) will get updated. In the second case, we increase \( \tilde{T}_k(n) \) by 1 for any arm \( k = \arg \min_{x \neq x_0} \{m_{x_{x,j}} : x \} \). Thus whenever a strategy with \( \Delta_x > \Delta_{x_0} \) is chosen, exactly one element in \( \{\tilde{T}_k(n)\} \) increases by 1. This implies that the total number that strategies with \( \Delta_x > \Delta_{x_0} \) have been played is equal to sum of all counters in \( \{\tilde{T}_k(n)\} \), i.e., \( \sum_{x=x_0+1}^{X} E[T_x(n)] = \sum_{k=1}^{K} \tilde{T}_k(n) \). Thus, we can rewrite the second term of (23) as

\[
\sum_{x=x_0+1}^{X} \Delta_x E[T_x(n)] \leq \Delta_x \sum_{x=x_0+1}^{X} E[T_x(n)] \leq \Delta_x \sum_{k=1}^{K} \tilde{T}_k(n).
\] (24)

Let \( I_k(t) \) be the indicator function that equals 1 if \( \tilde{T}_k(n) \) is updated at time slot \( t \). Define the indicator function \( 1\{y\} = 1 \) if the event \( y \) happens and 0 otherwise. When \( I_k(t) = 1 \), a strategy \( s_{x_{x,j}} > x_0 \) has been played for which \( m_k = \min_{m_{x_{x,j}} : x \} \). Then

\[
\tilde{T}_k(n) = \sum_{t=1}^{n} 1\{I_k(t) = 1\}
\] (25)

\[
\leq \sum_{t=1}^{n} 1\{W_t(t) \leq W_x(t)\}
\] (26)

\[
\leq \sum_{t=1}^{n} 1\{W_t \leq W_x\} \quad \text{(27)}
\]

\[
\leq \sum_{t=1}^{n} 1\{W_t \leq W_x(t), W_t \geq Z_x\}
\] (28)
\[ T_k^1(n) = \sum_{t=1}^{n} 1\{W_x(t) \geq Z_x\} \geq \sum_{x,j} \mu_{x,j} + \Delta_x \frac{K}{2N}. \] (31)

Using union bound one directly obtains:

\[ \hat{T}_k^1(n) \leq l_0 + \sum_{x,j \in s_x} P\left\{ \hat{\mu}_{x,j} \geq \frac{\mu_{x,j} + \Delta_x}{2N} \right\}. \] (32)

Now we let \( l_0 = 16N^2 \left[ \ln(\frac{n^{2/3}}{K}) \right]/\sqrt{2\Delta_x} \) with \([y]\) the smallest integer larger than \(y\). We further set \( \delta_0 = e^{1/2} \sqrt{K/n^{2/3}} \) and set \( x_0 \) such that \( \Delta_x \leq \delta_0 < \Delta_{x_0+1} \). As \( m_{x,j} \leq l_0 \),

\[ \ln(\frac{n^{2/3}}{K}) \leq \ln(\frac{n^{2/3}}{K}) \leq \ln(n^{2/3}/Kl_0) \]

\[ \ln(\frac{n^{2/3}}{K}) \leq \ln(n^{2/3}/Kl_0) \]

\[ \leq \ln(n^{2/3}/K) \leq \frac{\Delta_x}{2N}. \] (33)

Hence we have,

\[ \Delta_x - \frac{\ln(n^{2/3}/K)}{m_{x,j}} \geq \Delta_x - \frac{\Delta_x}{2N} = c\Delta_x. \] (34)

with \( c = \frac{1}{2N} - \frac{1}{\sqrt{16N^2}} = \frac{1}{4N}. \)

Therefore, using Hoeffding’s inequality and Equation (37), and then plugging into the value of \( l_0 \), we get,

\[ \hat{T}_k^1(n) \leq l_0 + \sum_{x,j \in s_x} \sum_{m_{x,j}=l_0}^{\infty} P\left\{ \hat{\mu}_{x,j} - \mu_{x,j} \geq c\Delta_x \right\} \]

\[ \leq l_0 + \sum_{x,j \in s_x} \sum_{m_{x,j}=l_0}^{\infty} \exp(-2m_{x,j}(c\Delta_x)^2) \]

\[ \leq l_0 + \sum_{x,j \in s_x} \exp(-2l_{m_{x,j}}(c\Delta_x)^2) \]

\[ = 1 + 16N^2 \sum_{x,j \in s_x} \frac{1}{1 - \exp(-2l_{m_{x,j}}(c\Delta_x)^2)}. \] (39)

For the second term of (40), we have \( y = \sqrt{eK/n^{2/3}} \) takes the maximum of \( y \to y^{-2} \ln(e/K/n^{2/3}) \). As \( \delta_0 = e^{1/2} \sqrt{K/n^{2/3}} \) and \( \Delta_x \geq \delta_0 \) in (40), we can get the second term bounded by \( 16N^2 e\Delta_x^2 n^{2/3}. \)

For the last term of (40) using \( 1 - \exp(-y) \geq y - y^2/2 \) for any \( y \) we have

\[ \frac{1}{1 - \exp(-y)} \leq \frac{1}{y} \leq \frac{1}{eKc(1 - c^2 N)} - \frac{1}{2Kc^2(1 - c^2 N)^{n^{2/3}}}. \] (40)

Finally we get

\[ \hat{T}_k^1(n) \leq 1 + \frac{16N^2}{eKc(1 - c^2 N)} n^{2/3}. \] (41)

3. Bounding \( \hat{T}_k^2(n) \)

\[ \hat{T}_k^2(n) = \sum_{t=1}^{n} 1\{W_x(t) \leq Z_x\} \]

\[ \leq \sum_{t=1}^{n} P\{W_x(t) < Z_x\} \leq nP\{W_x < Z_x\}. \] (42)

Remember that at time slot \( z \), we have \( W_1 = \min W_1(t) \). For the probability \( \{W_1 < Z_x\} \) of fixed \( x \), we have

\[ P\{W_1 < \lambda_1 - \frac{\Delta_x}{2}\} \]

\[ = P\left\{ \sum_{s_{1,j} \in s_{x,j}} w_{s_{1,j}}(z) < \lambda_1 - \frac{\Delta_x}{2}\right\}. \] (44)

\[ \leq \sum_{s_{1,j} \in s_{x,j}} P\{w_{s_{1,j}}(z) < \mu_{s_{1,j}} - \frac{\Delta_x}{2}\}. \] (45)

We define function \( f(u) = e \ln(\sqrt{2\pi u}/u^3) \) for \( u \in [\delta_0, N] \).

Then we have,

\[ P\{w_{s_{1,j}}(z) < \mu_{s_{1,j}} - \frac{\Delta_x}{2}\} = P\left\{ 31 \leq i \leq n : \sum_{t=1}^{l} (\xi_{s_{1,j}}(\tau) + \sqrt{\ln(\frac{n^{2/3}}{K})}) \right\}. \] (46)
\[
< \mu_{s_i,j} - \frac{l \Delta_s}{2N} \]

\[
\leq \Pr \left\{ \exists 1 \leq l \leq n : \sum_{\tau=1}^{l} (\mu_{s_i,j} - \xi_{s_i,j}(\tau)) > \sqrt{l \ln \left( \frac{2^3}{Kl} \right)} \right\}
\]

\[
\leq \Pr \left\{ \exists 1 \leq l \leq f(\Delta_s) : \sum_{\tau=1}^{l} (\mu_{s_i,j} - \xi_{s_i,j}(\tau)) > \sqrt{l \ln \left( \frac{2^3}{Kl} \right)} \right\}
\]

\[
+ \Pr \left\{ \exists f(\Delta_s) < l \leq n : \sum_{\tau=1}^{l} (\mu_{s_i,j} - \xi_{s_i,j}(\tau)) > \frac{l \Delta_s}{2N} \right\}.
\] (47)

For the first term we use a peeling argument with a geometric grid of the form \(0 \leq f(\Delta_s) \leq l \leq \frac{1}{2^g} f(\Delta_s)\):

\[
P \left\{ \exists 1 \leq l \leq f(\Delta_s) : \sum_{\tau=1}^{l} (\mu_{s_i,j} - \xi_{s_i,j}(\tau)) > \sqrt{l \ln \left( \frac{2^3}{Kl} \right)} \right\}
\]

\[
\leq \sum_{g=0}^{\infty} P \left\{ \exists \frac{1}{2^{g+1}} f(\Delta_s) \leq l \leq \frac{1}{2^g} f(\Delta_s) : \sum_{\tau=1}^{l} (\mu_{s_i,j} - \xi_{s_i,j}(\tau)) > \sqrt{\frac{f(\Delta_s) \ln \left( \frac{2^{2g} 3}{K f(\Delta_s)} \right)}{2^g}} \right\}
\]

\[
\leq \sum_{g=0}^{\infty} \exp \left( -2^g f(\Delta_s) \sqrt{\frac{\ln \left( \frac{2^3}{K f(\Delta_s)} \right)}{2^g}} \right)
\]

\[
\leq \sum_{g=0}^{\infty} \left[ \frac{\frac{K f(\Delta_s)}{n^{2/3}}}{2^g} \right] \leq \frac{2K f(\Delta_s)}{n^{2/3}}.
\] (48)

where in the second inequality we use Lemma 2.

As the special design of function \(f(u)\), we have \(f(u)\) takes maximum of \(e^{1/3} \sqrt{K/n^{1/3}}\) when \(u = e^{1/3} \sqrt{K/n^{1/3}}\). For \(\Delta_s > e^{1/3} \sqrt{K/n^{1/3}}\), we have

\[
\frac{2K f(\Delta_s)}{n^{2/3}} \leq \frac{2}{3\sqrt{K}} n^{-1/6}.
\] (49)

For the second term we also use a peeling argument but with a geometric grid of the form \(2^g f(\Delta_s) \leq l < 2^{g+1} f(\Delta_s)\):

\[
P \left\{ \exists f(\Delta_s) < l \leq n : \sum_{\tau=1}^{l} (\mu_{s_i,j} - \xi_{s_i,j}(\tau)) > \frac{l \Delta_s}{2N} \right\}
\]

\[
\leq \sum_{g=0}^{\infty} P \left\{ \exists 2^g f(\Delta_s) \leq l \leq 2^{g+1} f(\Delta_s) : \sum_{\tau=1}^{l} (\mu_{s_i,j} - \xi_{s_i,j}(\tau)) > \frac{2^g - 1 f(\Delta_s)}{N} \right\}
\]

\[
\leq \sum_{g=0}^{\infty} \exp \left( -2^g f(\Delta_s) \frac{\Delta_s^2}{4N^2} \right)
\]

\[
\leq \sum_{g=0}^{\infty} \exp \left( -(g+1) f(\Delta_s) \frac{\Delta_s^2}{4N^2} \right)
\]

\[
= \frac{1}{\exp(f(\Delta_s) \frac{\Delta_s^2}{4N^2})} - 1.
\] (50)

We note that \(f(u)u^2\) has a minimum of \(\frac{1}{\sqrt{K}} n^{1/6}\) when \(u = x_0\). Thus for (50), we further have,

\[
\frac{1}{\exp(\frac{f(\Delta_s) \Delta_s^2}{4N})} - 1 \leq \frac{1}{\exp(\frac{1}{\sqrt{K}} n^{1/6})} \leq 4\sqrt{K} N^2 n^{-\frac{1}{6}}.
\] (51)

Combining (46) and (43), we then have

\[
\tilde{T}_k(n) \leq \frac{2Nn^{5/6}}{3\sqrt{K}} + 4\sqrt{N^3 K^4} n^{5/6} \leq (1 + 4\sqrt{K} N^2) N n^{\frac{5}{6}} \tilde{T}_k(n).
\] (52)

4) Results without dependency on \(\Delta_{\min}\)

Summing \(\tilde{T}_k(n)\) and \(\tilde{T}_k^2(n)\), we have

\[
\tilde{T}_k(n) \leq \tilde{T}_k^1(n) + \tilde{T}_k^2(n)
\]

\[
= 1 + 16N^2 \left( \frac{8N}{15} \right) n^{\frac{5}{6}} + (1 + 4\sqrt{K} N^2) K n^{\frac{5}{6}}
\]

and using \(\Delta_X \leq N\) and \(\Delta_Z \leq \delta_0\) for \(x \leq x_0\), we have

\[
\gamma(n) \leq \sqrt{K} en^{\frac{5}{6}} + KN \left( 1 + 16N^2 \left( \frac{8N}{15} \right) n^{\frac{5}{6}} + (1 + 4\sqrt{K} N^2) K n^{\frac{5}{6}} \right)
\]

\[
\leq N K \left( \sqrt{K} + (8N + N^3) \right) n^{\frac{5}{6}}
\]

\[
+(1 + 4\sqrt{K} N^2) K n^{\frac{5}{6}}.
\] (54)

Appendix B

Proof of Theorem 4

Recall that we have \(\lambda_1 \geq \cdots \geq \lambda_X\), and \(Z_x = \lambda_1 - \frac{\Delta_s}{\gamma}\). This time we set \(\Delta_{x_0} \leq \delta_0 = \sqrt{K} n \Delta_{x_0} + 1\). Splitting strategy set \(F\) into two disjoint sets again by \(\Delta_{x_0}\), and plugging (24) into (23), we begin with a weak vision of (23).

\[
\gamma(n) \leq \Delta_{x_0} + \Delta_X \sum_{k=1}^{11} \tilde{T}_k(n).
\] (55)

Here we have the same form of \(\tilde{T}_k(n)\) as that in (30), i.e.,

\[
\tilde{T}_k(n) = \tilde{T}_k^1(n) + \tilde{T}_k^2(n)
\]

\[
\tilde{T}_k^1(n) \leq \sum_{t=1}^{n} I\{W_1 \leq W_x(t), W_1 \geq Z_x\},
\] (57)

\[
\tilde{T}_k^2(n) \leq \sum_{t=1}^{n} I\{W_1 \leq W_x(t), W_1 < Z_x\}.
\] (58)

From (40), we have

\[
\tilde{T}_k^1(n) \leq 1 + 16N^2 \ln \left( \frac{2^{2/3} \Delta_s}{N} \right) + \sum_{s_x,j} \frac{1}{\Delta_x^2} 1 - \exp(-2(c\Delta_x)^2)
\]

\[
\leq 1 + 16N^2 \ln \left( \frac{2^{2/3} \Delta_s}{N} \right) + \sum_{s_x,j} 2c^2 \Delta_x^2 (1 - c^2 \Delta_x^2)
\]

\[
\leq 1 + 16N^2 \ln \left( \frac{2^{2/3} \Delta_s}{N} \right) + \frac{128N^3}{15\Delta_s^2}.
\] (59)
where we use $\frac{K}{n} \Delta_{x}^{2} > e$ for $x > x_{0}$ in the third term, and $\Delta_{\min} \leq \Delta_{x} \leq N$ in the last term.

As to $\tilde{T}_{k}^{2}(n) = nP\{W_{1} < Z_{x}\}$, according to (46),

$$P\{W_{1} < Z_{x}\} \leq \sum_{s_{1},j \in s_{1}} P\left\{ w_{s_{1},j}(z) < \mu_{s_{1},j} - \frac{\Delta_{x}}{2N} \right\}.$$  \hspace{1cm} (60)

Then the probability of $\{ w_{s_{1},j}(z) < \mu_{s_{1},j} - \frac{\Delta_{x}}{2N} \}$ can be divided into two elements by introducing a function $f(\Delta_{x}) < n$. Here we again follow a similar scheme as done in proof of Theorem 3. We reset the function $f(\Delta_{x}) = 4N^{2} \ln(n\Delta_{x}^{2}/K)/\Delta_{x}^{2}$, and let

$$P_{1} = P\left\{ \exists 1 \leq l \leq f(\Delta_{x}) : \sum_{\tau=1}^{l} (\mu_{s_{1},j} - \xi_{s_{1},j}(\tau)) > \sqrt{\ln n} \left( \frac{2\beta}{K}\right) \right\}.$$  \hspace{1cm} (61)

For $P_{1}$ we use a peeling argument with a geometric grid of the form $\frac{1}{n} f(\Delta_{x}) \leq l < \frac{1}{2} f(\Delta_{x})$, then we have the following by using similar technique of (48):

$$P_{1} \leq \frac{2K f(\Delta_{x})}{n^{2/3}} = \frac{8N^{2} K \ln \left( \frac{n}{\Delta_{x}^{2}} \right)}{n^{2/3}}.$$  \hspace{1cm} (62)

For $P_{2}$ we also use a peeling argument but with a geometric grid of the form $2^{\alpha} f(\Delta_{x}) \leq l < 2^{\alpha+1} f(\Delta_{x})$, then we have the following by using similar technique of (50).

$$P_{2} \leq \frac{1}{\exp \left( \frac{\sqrt{\Delta_{x}^{2}}}{4N^{2}} \right) - 1} \leq \frac{1}{K \Delta_{x}^{2} - 1} \leq \frac{K}{(1 - \frac{1}{e}) \Delta_{\min}n}.$$  \hspace{1cm} (63)

where once again we use $\frac{n \Delta_{x}^{2}}{K} > 1$ with $x > x_{0}$, and $\frac{1}{K \Delta_{\min}n} \geq 1$ in the last step.

Recall that $\tilde{T}_{k}^{2}(n) \leq n \sum_{s_{1},j \in s_{1}} (P_{1} + P_{2})$, thus combining previous analysis we have

$$\tilde{T}_{k}^{2}(n) \leq \sum_{s_{1},j \in s_{1}} P\left\{ w_{s_{1},j}(z) < \mu_{s_{1},j} - \frac{\Delta_{x}}{2N} \right\}.$$  \hspace{1cm} (64)

Since $\Delta_{x_{0}} \leq \sqrt{eK/n}$, putting (59) and (63) in (55), we obtain

$$\mathcal{R}(n) \leq \frac{eK}{\Delta_{x_{0}}} + (17 + 8.53N) K \ln \left( \frac{n}{\Delta_{x}^{2}} \right) \frac{K N}{\Delta_{\min}^{2}} + \left( \frac{8N^{3} K n^{-\frac{1}{2}}} {1 - 1/e} \right) N K \ln \left( \frac{n}{\Delta_{x}^{2}} \right) \frac{K N}{\Delta_{\min}^{2}} \leq \left( e + (17 + 8.53N) N^{3} + 8N^{4} K n^{-\frac{1}{2}} \right) \frac{K N}{1 - 1/e} \frac{\ln \left( \frac{n}{\Delta_{x}^{2}} \right)}{\Delta_{\min}^{2}}.$$  \hspace{1cm} (65)

**APPENDIX C**

**PROOF OF THEOREM 3**

Here we still assume feasible strategy set $F$ in analysis of lower bound for all $\beta$-approximation policies. Then we follow the same notations in analysis of Theorem 3 if not specified. We have

$$\lambda_{1} \geq \lambda_{2} \geq \cdots \geq \lambda_{x_{\beta}} \geq \cdots \geq \lambda_{x},$$

where $x_{\beta}$ is the greatest index of strategies satisfying $\lambda_{x} - \lambda_{1}/\beta \geq 0$.

Remember that we have defined $\Delta_{\beta,x}$ as the distance between $R_{1}/\beta$ and reward of strategy $s_{x}$. Similar to proof of Theorem 3 we introduce a split $x_{\beta,0}$ with $\lambda_{x_{\beta,0}} < R_{1}/\beta$ (or $x_{\beta,0} > x_{\beta}$) to divide the strategies into two disjoint sets. Then the regret caused by non-$\beta$-approximation strategies can be written as:

$$\mathbb{R}_{\beta}(n) \leq \sum_{x : x_{\beta} < x \leq x_{\beta,0}} \Delta_{\beta,x} E[T_{x}(n)] \leq n \Delta_{x_{\beta,0}} + \sum_{x = x_{\beta,0} + 1}^{x} \Delta_{\beta,x} E[T_{x}(n)] \leq n \Delta_{x_{\beta,0}} + \frac{1}{\beta} \Delta_{\max} \sum_{x = x_{\beta,0} + 1}^{x} E[T_{x}(n)].$$  \hspace{1cm} (66)

In the last term we use the fact that $\frac{1}{\beta} \Delta_{\max} \geq \Delta_{\beta,\max}$.

Using a set of counters $\{ \tilde{T}_{k}(n) \mid k = 1, \ldots, K \}$ to count the number of times that strategies of index $x > x_{\beta,0}$ have been played up to time slot $n$, update $\tilde{T}_{k}(n)$ if a strategy $s_{x}$ of index $x > x_{\beta,0}$ is played for which $m_{k} = \min_{s_{x,j} \in s_{x}} \{ m_{s_{x,j}} \}$, we have

$$\sum_{x = x_{\beta,0} + 1}^{x} E[T_{x}(n)] = \sum_{k=1}^{K} E[\tilde{T}_{k}(n)].$$  \hspace{1cm} (67)

Let indicator function $I_{k}(t) = 1$ denoting the event that $\tilde{T}_{k}(n)$ get updates at time $t$, we have

$$\tilde{T}_{k}(n) = \sum_{t=1}^{n} I_{k}(t) = 1$$

$$\leq \sum_{t=1}^{n} I_{k}(t) \{ W_{1} \leq \beta W_{x}(t), W_{1} \geq \beta Z_{\beta,x} \} \leq \sum_{t=1}^{n} I_{k}(t) \{ W_{1} \leq \beta W_{x}(t), W_{1} < \beta Z_{\beta,x} \} \leq \sum_{t=1}^{n} I_{k}(t) \{ W_{1} \leq \beta W_{x}(t), W_{1} \geq Z_{\beta,x} \} \leq \tilde{T}_{k}^{1}(n) + \tilde{T}_{k}^{2}(n).$$  \hspace{1cm} (68)

For (66), the event $\{ W_{1} \leq \beta W_{x}(t) \}$ and $\{ W_{1} \geq \beta Z_{\beta,x} \}$ implies $\{ W_{x}(t) \geq Z_{\beta,x} \}$. Taking similar approaches in proof of Theorem 3 then for any positive integer $l_{\beta,0} > 0$ we have
the following for $\tilde{T}_k(n)$:

$$
\sum_{t=1}^{n} 1\{W_1 \leq \beta W_x(t), W_1 \geq \beta Z_{\beta,x}\}
\leq \sum_{t=1}^{n} 1\{W_x(t) \geq Z_{\beta,x}\}
\leq \sum_{t=1}^{n} 1\{W_x(t) \geq \lambda_x + \frac{\Delta_{\beta,x}}{2}\}
\leq l_{\beta,0} + \sum_{t=1}^{n} P\left\{ \sum_{s_{x,j} \in s_x} \left( \bar{\mu}_{s_{x,j}} + \sqrt{\frac{\ln \left( \frac{K^{1/3}}{K^2 s_{x,j}, x}\right)}{l_{\beta,0}}} \right) \right\}
\geq \sum_{s_{x,j} \in s_x} \mu_{s_{x,j}} + \frac{\Delta_{\beta,x}}{2}.
$$

(72)

The expression of (73) then becomes quite the same with that of (34) in proof of Theorem 3. Replacing $\Delta_{\beta,x}$ by $\Delta_{\beta,x}$, setting $l_{\beta,0} = 16N^2[\ln((\pi/2)^2/\Delta_{\beta,x}^2)]$, and utilizing $\Delta_{x,0} \geq \delta_{\beta,0} = e^{1/2}/\sqrt{K/n^{2/3}}$, we have

$$
\tilde{T}_k(n) \leq 1 + 16N^2 \frac{\ln(n^{1/3}/\Delta_{\beta,x}^2)}{\Delta_{\beta,x}^2} + \sum_{s_{x,j} \in s_x} \frac{1}{1 - \exp(-2(e\Delta_{\beta,x}))}
\leq 1 + 16N^2 \left( \frac{N}{Ke} + \frac{2Ke^2(1 - e^2N^2)}{Ke} \right) n^{2/3}
= 1 + 16N^2 eK/(1 + 8N/15)n^{2/3}.
$$

(73)

For (77), we have

$$
\tilde{T}_k(n) \leq \sum_{t=1}^{n} 1\{W_1 \leq \beta W_x(t), W_1 \geq \beta Z_{\beta,x}\}
\leq nP(W_1 < \beta Z_{\beta,x})
= nP(W_1 < \lambda_x - \frac{\beta \Delta_{\beta,x}}{2})
= nP\left\{ \sum_{s_{x,j} \in s_x, j=1}^{N} w_{s_{x,j}}(z) < \lambda_x - \frac{\beta \Delta_{\beta,x}}{2} \right\}
\leq nP\left\{ \exists s_{x,j} \in s_x : w_{s_{x,j}}(z) < \mu_{s_{x,j}} - \frac{\beta \Delta_{\beta,x}}{2N} \right\}
\leq n \sum_{s_{x,j} \in s_x, j=1}^{N} P\left\{ w_{s_{x,j}}(z) < \mu_{s_{x,j}} - \frac{\beta \Delta_{\beta,x}}{2N} \right\}.
$$

(77)

For the probability of $P\left\{ w_{s_{x,j}}(z) < \mu_{s_{x,j}} - \frac{\beta \Delta_{\beta,x}}{2N} \right\}$, we use function $f(u) = e \ln(\sqrt{n^{1/3}/K}u)/u^3$ for $u \in [\delta_{\beta,0}, N]$. Let $A_1 = P\left\{ \exists 1 \leq l \leq f(\Delta_{\beta,x}): \sum_{\tau=1}^{l} (\mu_{s_{x,j}} - \zeta_{s_{x,j}}(\tau)) > \sqrt{\ln(\frac{K^{1/3}}{lK^2})} \right\}$, we have

$$
A_1 = P\left\{ \exists 1 \leq l \leq f(\Delta_{\beta,x}): \sum_{\tau=1}^{l} (\mu_{s_{x,j}} - \zeta_{s_{x,j}}(\tau)) > \sqrt{\ln(\frac{K^{1/3}}{lK^2})} \right\}.
$$

(78)

and

$$
A_2 = P\left\{ \exists f(\Delta_{\beta,x}) < l \leq n : \sum_{\tau=1}^{l} (\mu_{s_{x,j}} - \zeta_{s_{x,j}}(\tau)) > \frac{\Delta_{\beta,x}}{2N} \right\}.
$$

(79)

Using a peeling argument with a geometric grid of the form $\sum_{\tau=1}^{l} f(\Delta_{\beta,x}) \leq l \leq \frac{1}{\beta} f(\Delta_{\beta,x})$, we have

$$
A_1 \leq \frac{2Kf(\Delta_{\beta,x})}{n^{2/3}} \leq \frac{2}{\sqrt{K}} n^{5/6}
$$

(81)

Using a peeling argument with a geometric grid of the form $\sum_{\tau=1}^{l} f(\Delta_{\beta,x}) \leq l \leq \frac{1}{\beta} f(\Delta_{\beta,x})$, then we have the following by using similar technique of (50).

$$
A_2 \leq \exp(\frac{f(\Delta_{\beta,x})\Delta_{\beta,x}^2\beta/4N^2}{\beta^2}) - 1 \leq \frac{4\sqrt{K}N^2}{e\beta^2} n^{5/6}
$$

(82)

Following the approaches from Equation (46) for proof of Theorem 3 we can get $\tilde{T}_k(n)$ bounded by

$$
\tilde{T}_k(n) \leq \frac{2N}{3\sqrt{K}} n^{5/6} + \frac{4\sqrt{K}N^3}{e\beta^2} n^{5/6} \leq (1 + \frac{4\sqrt{K}N^2}{e\beta^2}) N n^{5/6}
$$

(83)

Plugging (70), (72) into $\tilde{T}_k(n) \leq \tilde{T}_k(n) + \tilde{T}_k(n)$, we have

$$
\tilde{T}_k(n) \leq 1 + \frac{16N^2 eK}{(1 + 8N/15)n^{2/3}} + \left(1 + \frac{4\sqrt{K}N^2}{e\beta^2}\right)n n^{5/6}
$$

(84)

With the above result and $\Delta_{x,0} \geq \delta_{\beta,0} = e^{1/2}/\sqrt{K/n^{2/3}}$, we have the following for regret in Equation (64).

$$
\mathbb{R}_\beta(n) \leq N K/\beta + \left(\sqrt{eK} + 16(1 + N)N^3/(e\beta)\right) n^{3/2}
+ \left(1 + \frac{4\sqrt{K}N^2}{e\beta^2}\right) N^2 K/\beta \ n^{1/2}.
$$

(85)

**APPENDIX D**

**PROOF OF THEOREM 6**

We then prove the results for $\beta$-approximation policy with dependency on $\Delta_{\beta,\min}$. Without loss of generality, we still assume strategy set $P$ with $A_1 \geq \cdots \geq A_\lambda$. Recall that $\Delta_{\beta,x} = \lambda_x/\beta - \lambda_x$, and $Z_{\beta,x} = \lambda_x/\beta - \Delta_{\beta,x}/2$. Define index $x_0$ satisfying $\Delta_{\beta,x_0} \leq \delta_{\beta,0} < \Delta_{\beta,x_0} + 1$ where this time we set $\delta_{\beta,0} = \sqrt{eK/n}$. Let $\mathbb{R}_\beta(n) \leq \sum_{x : \Delta_{\beta,x} \leq R_1/\beta} \Delta_{\beta,x} E[T_x(n)]$

$$
\leq n \Delta_{\beta,0} + \sum_{x = \delta_{\beta,0} + 1}^{\Delta_{\beta,x}} \Delta_{\beta,x} E[T_x(n)]
\leq n \Delta_{\beta,0} + \Delta_{\beta,0} \max_{x = \delta_{\beta,0} + 1} \sum_{k=1}^{K} E[T_k(n)]
\leq n \Delta_{\beta,0} + \frac{1}{\beta} \Delta_{\beta,0} \max_{x = \delta_{\beta,0} + 1} \sum_{k=1}^{K} E[T_k(n)]
$$

(86)
where the last step is from (65), and \( \{ T_k(n) \}_{k=1, \ldots, K} \) denotes the number of times that strategies of index \( x > x_{\beta, 0} \) have been played up to time slot \( n \).

We rewrite \( \hat{T}_k(n) = \hat{T}_k^1(n) + \hat{T}_k^2(n) \) from (66) and (67), each denoting,

\[
\hat{T}_k^1(n) = \sum_{t=1}^{n} 1 \{ W_t \leq \beta W_x(t), W_t \geq \beta Z_{\beta, x} \},
\]

\[
\hat{T}_k^2(n) = \sum_{t=1}^{n} 1 \{ W_t \leq \beta W_x(t), W_t < \beta Z_{\beta, x} \}.
\]

For the first term above, we directly get the following from (72),

\[
\hat{T}_k^1(n) \leq 1 + 16N^2 \frac{\ln(\frac{\tau^{2/3}}{\Delta_{\beta, x}})}{\Delta_{\beta, x}} + \sum_{s_x, j \in s_x} \frac{1}{1 - \exp(-2(c\Delta_{\beta, x}^2))}
\]

\[
\leq (17 + 8.53N)N^2 \frac{\ln \left( \frac{\tau}{\beta N^2} \right)}{\Delta_{\beta, \min}^2}.
\]

by \( \frac{n\Delta_{\beta, x}^2}{K^2} \geq 1 \) when \( x > x_{\beta, 0} \).

For the second term, we follow (77),

\[
\hat{T}_k^2(n) \leq n \sum_{s_x, j \in s_x} \mathbb{P} \left( w_{s_x, j}(z) < \mu_{s_x, j} - \frac{\beta \Delta_{\beta, x}}{2N} \right).
\]

And we reset the function \( f(u) = 4N^2 \frac{\ln(u^2/K)}{\beta^2 u^2} \), and let

\[
A_1 = \mathbb{P} \left\{ \exists 1 \leq l \leq f(\Delta_{\beta, x}) : \sum_{\tau=1}^{l}(\mu_{s_x, j} - \xi_{s_x, j}(\tau)) > \sqrt{\ln(\frac{\tau^{2/3}}{K})} \right\},
\]

\[
A_2 = \mathbb{P} \left\{ \exists f(\Delta_{\beta, x}) < l \leq n : \sum_{\tau=1}^{l}(\mu_{s_x, j} - \xi_{s_x, j}(\tau)) > \frac{\Delta_{\beta, x}}{2N} \right\}.
\]

We have \( \mathbb{P} \left( w_{s_x, j}(z) < \mu_{s_x, j} - \frac{\beta \Delta_{\beta, x}}{2N} \right) \leq A_1 + A_2 \).

Using a peeling argument with a geometric grid of the form \( \frac{1}{2^{2+l}} f(\Delta_{\beta, x}) \leq 1 \leq \frac{1}{2^l} f(\Delta_{\beta, x}) \), we have

\[
A_1 \leq \frac{2K f(\Delta_{\beta, x})}{n^{2/3}} \leq \frac{8N^2 K \ln \left( \frac{\tau^{2/3}}{K} \right)}{\beta^2 n^{2/3}} \Delta_{\beta, x}^2
\]

\[
\leq \frac{8N^2 K \ln \left( \frac{\tau}{\beta N^2} \right)}{\beta^2 n^{2/3}} \Delta_{\beta, \min}^2.
\]

Using a peeling argument with a geometric grid of the form \( 2^l f(\Delta_{\beta, x}) \leq l < 2^{l+1} f(\Delta_{\beta, x}) \), then we have the following by using similar technique of (60),

\[
A_2 \leq \frac{1}{n \Delta_{\beta, x}^2 / K - 1} \leq \frac{K}{(1 - 1/e) \Delta_{\beta, \min}^2 n}.
\]

Thus we have

\[
\hat{T}_k^2(n) \leq 8N^3 Kn \frac{\ln \left( \frac{\tau}{\beta N^2} \right)}{\beta^2 \Delta_{\beta, \min}^2} + \frac{NK}{(1 - 1/e) \Delta_{\beta, \min}^2}
\]

\[
\leq \left( 8N^3 Kn \frac{\ln \left( \frac{\tau}{\beta N^2} \right)}{\beta^2} + \frac{NK}{(1 - 1/e) \Delta_{\beta, \min}^2} \right) \Delta_{\beta, \min}.
\]