Online Learning for Adaptive Probing and Scheduling in Dense WLANs

Tianyi Xu[a], Ding Zhang[b], Zizhan Zheng[a]
[a] Department of Computer Science, Tulane University, New Orleans, LA, USA
[b] Department of Computer Science, George Mason University, Fairfax, VA, USA

***Abstract*—Existing solutions to network scheduling typically assume that the instantaneous link rates are completely known before a scheduling decision is made or consider a bandit setting where the accurate link quality is discovered only after it has been used for data transmission. In practice, the decision maker can obtain (relatively accurate) channel information, e.g., through beamforming in mmWave networks, right before data transmission. However, frequent beamforming incurs a formidable overhead in densely deployed mmWave WLANs. In this paper, we consider the important problem of throughput optimization with joint link probing and scheduling. The problem is challenging even when the link rate distributions are pre-known (the offline setting) due to the necessity of balancing the information gains from probing and the cost of reducing the data transmission opportunity. We develop an approximation algorithm with guaranteed performance when the probing decision is non-adaptive and a dynamic programming-based solution for the more challenging adaptive setting. We further extend our solutions to the online setting with unknown link rate distributions and develop a contextual-bandit based algorithm and derive its regret bound. Numerical results using data traces collected from real-world mmWave deployments demonstrate the efficiency of our solutions.**

***Index Terms*—wireless scheduling, online learning, adaptive probing**

I. Introduction

An important obstacle to efficient resource allocation in wireless networks is the uncertain and time-varying nature of link quality. Traditionally, it is common to assume that the decision maker can observe the time-varying channel condition at the beginning of a time slot before a scheduling decision is made [1], leading to the so called *opportunistic scheduling* problems that can be solved using various stochastic optimization techniques. More recently, online learning algorithms have been intensively studied for wireless resource allocation, including link scheduling [2], [3], rate adaptation [4] and beam selection [5] where a typical assumption is that the channel condition of a link is unknown before a scheduling decision is made and is discovered only after the link has been used for data transmission (i.e., the *bandit* setting).

In practice, a network controller can often obtain (relatively) accurate channel conditions before transmitting data. For example, access points (APs) in 60 GHz millimeter-wave (mmWave) WLANs can use beamforming to identify the link quality to its clients. However, frequent beamforming between a large number of APs and clients incurs a formidable overhead. For example, it takes approximately 5ms to train the downlink of an AP with 64 Tx sectors to its clients with 16 Rx sectors as shown in [6], and the overhead increases linearly with the number of APs. Thus, it is crucial to develop joint beamforming and scheduling schemes that can achieve the optimal tradeoff between the reduction of uncertainty and the extra overhead introduced by beamforming. This is especially important in densely deployed mmWave WLANs that are increasingly being used to provide high capacity and reliability to clients. However, neither traditional stochastic optimization approaches nor bandit-based online learning methods can be directly applied as the former ignores the overhead associated with beamforming while the latter does not utilize the side information provided by beamforming.

In this work, we propose a joint probing and scheduling framework for throughput optimization. We consider a scenario where a mobile client can probe a subset of APs from all the neighboring APs before picking one of them for data transmission and study both the non-adaptive setting where the set of APs is chosen independently of the probing results as well as the more powerful yet challenging adaptive setting where the next probing decision can depend on the previous probing results. We first consider the offline setting where the distributions of link rates are pre-known. Deriving an optimal probing and scheduling strategy is challenging, even in this case, due to the need to balance the information gained from probing and the cost of reduced data transmission opportunity. We design an approximation algorithm for the non-adaptive setting and derive its performance bound. For the adaptive setting, we propose a dynamic programming-based solution and prove that the algorithm is optimal for Bernoulli link rates.

In the more challenging online setting where the link rate distributions are unknown a priori, we present a novel extension of the contextual bandit learning framework by incorporating adaptive probing into the classic exploration vs. exploitation tradeoff. In particular, the probing results help with both the exploitation in the current round and the exploration that can benefit future rounds. Through a careful analysis of the reward function and by utilizing the results for the offline setting, we show that the UCB-based stochastic combinatorial bandit algorithm in [7] can be adapted to our problem and derive its regret bound for both non-adaptive probing under general distributions and adaptive-probing with Bernoulli link rates.

Our framework can potentially be applied to a broad class of sequential decision making problems where probing can be used to reduce uncertainty before a decision is made. For example, consider the problem of shortest path routing in a road network with unknown traffic. A path searcher can make a limited number of queries to a travel server to obtain hints of real time travel latency [8], before picking a path. Further, the path searcher may utilize contextual data such as time and weather to assist decision making. This problem can be formulated as a combinatorial bandit problem with probing.

We have made the following contributions in this paper.

- We propose a joint probing and scheduling framework for throughput optimization in wireless networks.
- In the offline setting with known link rate distributions, we develop an approximation solution when the probing decision is non-adaptive and a dynamic programming-based solution for the adaptive setting. We show that both algorithms are optimal for Bernoulli link rates and bound the performance of the former for general distributions.
- In the online setting with unknown link rate distributions, we adapt a stochastic combinatorial bandit algorithm to our problem and derive its regret.
- Our solutions are validated using channel and mobility traces collected from a real testbed.

II. Related Work

Multi-armed bandits: The multi-armed bandit (MAB) problem provides a principled framework for sequential decision making under uncertainty [9]–[11]. The framework has been extensively applied to various domains, including wireless resource allocation and link scheduling, with multiple variants of the vanilla MAB model considered [9], [12], [13]. In [14], the Upper Confidence Bound (UCB) algorithm is integrated with the classic greedy maximal matching algorithm for wireless scheduling. In [15], the problem of beam selection in mmWave based vehicular networks is formulated as a contextual bandit problem with the vehicle's direction of arrival and other traffic information as context. In [16], a constrained Thompson Sampling (TS) algorithm is used to solve the link rate selection problem by exploiting the property that a higher data rate is typically associated with a lower transmission success probability. However, none of the above studies consider joint probing and play. Although the recent work [17] considers joint probing and play, it focuses on non-adaptive probing strategies and derives the regret bound for Bernoulli link rates only.

Decision-making with probes: Searching for best alternatives with uncertainty using probing is a problem initiated in economics [18] and has found applications in database query optimization [19]–[21] and wireless communications [22] [23]. Various objectives have been considered including searching for extreme values with a limited budget, maximizing the largest value found minus the total probing cost spent, or solving a knapsack problem with uncertain weights and costs [24]. Since the problems are typically NP-hard problems under general probability distributions, approximation algorithms have been considered for the non-adaptive probing [24] and adaptive probing settings [25]. Recently, probing policies have been studied for optimizing path search over a road network [8]. However, all the above works consider the offline setting with known distributions. Further, none of them consider the same objective function used in our problem. In [26], an endhost-based shortest path routing problem with unknown and time-varying link qualities is modeled as an online learning problem. The paper considers decoupled routing and probing decisions, where a single probe is allowed before a route is selected at each time round, and link qualities can only be observed through probes (but not the actual routing decisions). Because of the simplifications, the paper adopts a simple probing strategy by always probing the path containing the least measured link. In contrast, we consider multiple (and possibly adaptive) probes in each round, and both probing and play provide feedback on rewards. Our work is also closely related to the sequential channel sensing and access problem in cognitive radio networks studied in [27]. However, the work only considers Bernoulli channels without context. Further, a channel can be accessed only when it is sensed idle. In contrast, our problem allows playing an unprobed arm.

III. System Model and Problem Formulation

In this section, we present our joint probing and scheduling framework. We start with the general online setting, which includes the offline setting as a special case. We further introduce the concepts of non-adaptive probing and adaptive probing.

A. Contextual bandits with probing

Consider a set of APs that are connected to a high-speed backhaul to collectively serve a set of mobile users. AP collaboration helps boost wireless performance in both indoor and outdoor environments and is especially useful for directional mmWave networks that are susceptible to blockage [6], [15]. For simplicity, we assume that the beamforming process determines the best Tx and Rx beam sectors and do not distinguish AP selection from beam selection. Our framework readily applies to the more general setting of joint AP and beam selection.

To simplify the problem, we focus on the single client setting in this work. Let X be a set of contexts that correspond to the location (or a rough estimate of it) of a moving client. Let A be a discrete set of arms that correspond to the set of APs, and $N \triangleq |A|$. We consider a fixed time horizon T that is known to the decision maker. In each time round t, the decision maker first receives a context $x_t \in X$ and then plays an arm $a_t \in A$ and receives a reward $Y_{a_t|x_t} \in [0,1]$, which is $i.i.d.$ sampled from an *unknown* distribution $D_{a_t|x_t}$, that depends on both the context x_t and the arm a_t. We assume that the outcomes from all N arms are mutually independent. In this work, we assume that for all context-arm pairs, $D_{a|x}$ is a discrete distribution with a finite support $\{y_{a|x,1}, ..., y_{a|x,l}\}$ where $0 = y_{a|x,1} < \cdots < y_{a|x,l-1} < y_{a|x,l} = 1$. Let $\mathcal{Y}$ denote the union of possible values across all the arms. We have $|\mathcal{Y}| \leq N \times l$. Let $\mu_{a|x}$ denote the expected value of $D_{a|x}$. The finite support assumption is mainly needed for our offline dynamic programming algorithm (Algorithm 3) and the online algorithm (Algorithm 4). Let $\mathbb{D} = \{D_{a|x}\}$ denote the collection of distributions over all the context-arm pairs. The context sequence $(x_t)_{t\in\mathbb{N}}$ is considered to be outside the decision making process. In the AP selection problem, the reward corresponds to the data rate that a client at a certain location can receive from an AP.

In the classical stochastic bandit setting, the instantaneous reward of an arm is not revealed until it is played. In contrast, we consider a more general setting where after receiving the context x_t, the network controller can first probe a subset of arms $S_t \subseteq A$, observe their outcomes $Y_t = \{Y_{a|x_t} : a \in S_t\}$, where $Y_{a|x_t}$ is an *i.i.d.* sample of $D_{a|x_t}$, then pick an arm a_t to play (which may be different from the set of probed arms). If a_t is from the probing set, the reward received is the same as the probed value. That is, we assume that probing is accurate and the channel condition stays the same within a time round. To model the probing overhead, we further assume that up to K arms can be probed in a single time round and the (normalized) throughput obtained for probing S_t with $|S_t| = k$ and playing a_t is

$$R(S_t, Y_t, a_t|x_t) = \frac{K-k}{K} Y_{a_t|x_t}. \tag{1}$$

Without loss of generality, we assume $K \leq N$. The goal of the network controller is to maximize the total expected throughput over the time horizon T. In general, the optimal choice of a_t given S_t depends on Y_t. Let $r(S_t, a_t|x_t) \triangleq \mathbb{E}_{Y_t}[R(S_t, Y_t, a_t|x_t)]$.

B. Non-adaptive probing

We distinguish two types of probing strategies in this work. In the non-adaptive case, the probing set S_t is chosen independently of the probing results. This setting is useful when adaptive probing is infeasible but it also serves as an important baseline. In this case, the decision in each time round becomes a pair (S_t, a_t) where S_t is the set of arms to probe and a_t is the arm to play. Let H_{t-1} denote the history observations that the decision maker has received before time t. Then the probing policy π_1 maps H_{t-1} to a subset of arms S_t to probe while the play policy π_2 maps (H_{t-1}, S_t, Y_t) to the arm a_t to play. Let $\pi = (\pi_1, \pi_2)$.

Let (S_t^*, a_t^*) denote the optimal offline non-adaptive solution under context x_t when the true distributions $D_{a|x}$ are known for all context-arm pairs. The goal of maximizing the expected cumulative reward then converts to minimizing the expected regret $\mathbb{E}(\mathsf{Reg}(T))$ with $\mathsf{Reg}(T)$ defined as

$$\mathsf{Reg}(T) = \sum_{t=1}^{T} (r(S_t^\star, a_t^\star|x_t) - r^\pi(S_t, a_t|x_t)) \tag{2}$$

where r^π denotes the throughput obtained by following policy π and the expectation is with respect to the randomness in both outcomes and policies. The goal is to obtain a policy π with sublinear regret where $\lim_{T\to\infty} \mathbb{E}(\mathsf{Reg}(T))/T = 0$.

For general context-arm distributions $D_{a|x}$, it can be computationally difficult to find the optimal solution even when the link rate distributions are perfectly known (the offline setting). On the other hand, efficient approximation algorithms may exist. In particular, we say that an offline algorithm is an α-approximation ($0 < \alpha \leq 1$) if it obtains an expected throughput that is at least $\alpha\mathbb{E}r(S^\star, a^\star|x)$ for any context $x \in X$. In this case, it is difficult to obtain sublinear regret with respect to (2). Instead, our goal is to obtain sublinear α-approximation regret defined as follows [7]:

$$\mathsf{Reg}_\alpha(T) = \sum_{t=1}^{T} (\alpha r(S_t^\star, a_t^\star|x_t) - r(S_t, a_t|x_t)) \tag{3}$$

C. Adaptive probing

In the adaptive setting, S_t is chosen sequentially where the next arm to probe can depend on the previous probing results in the current time round. Let $S_{i,t}$ denote the subset of arms probed in the first i probing steps in time t, and $Y_{i,t}$ the corresponding probing outcome. Then the probing policy π_1 can generally be defined as a mapping from $(H_{t-1}, S_{i,t}, Y_{i,t})$ to the next probing decision (either the next arm to probe or terminating probing). The play policy π_2 has the same form as the non-adaptive case. We can then define regret and α-approximation regret similarly to the non-adaptive case by using the optimal offline adaptive solution as the baseline.

D. Assumption on the context space

We consider contextual bandits with Lipschtz-continuity to handle large context spaces. To this end, we normalize the context space to the $[0, 1]$ interval so that $X = [0, 1]$ and the expected rewards are assumed to be Lipshitz with respect to the contexts:

$$|r(S, a|x) - r(S, a|x')| \leq L|x - x'|, \forall S \subseteq A, a \in A, x, x' \in X \tag{4}$$

where L is the Lipschitz constant known to the algorithm.

IV. The Offline Case

In this section, we consider the offline case where the reward distributions $D_{a|x}$ are fully known for all context-arm pairs. In this case, the history H_{t-1} is irrelevant to decision making and we can focus on a single time round. To simplify the notation, we omit the time index t and context x_t, and write $S = S_t$, $Y = Y_t$, $Y_a = Y_{a|x_t}$, $D_a = D_{a|x}$, and $\mu_a = \mu_{a|x}$. We first observe that given a probing set S and its outcome Y, the optimal arm to play can be easily identified, which allows us to focus on π_1 in this section.

Observation 1: Given a probing set S, let $Z_a = Y_a$ for $a \in S$ and $Z_a = \mu_a$ for $a \in A\backslash S$. It is optimal to play an arm a that maximizes Z_a.

A. The offline problem with non-adaptive probing

We first consider the offline problem with non-adaptive probing. Given Observation 1, the problem boils down to finding a subset of arms $S \subseteq A$ to probe to maximize $R(S) \triangleq \frac{K-|S|}{K} g(S)$ where $g(S)$ is defined as

$$g(S) = \mathbb{E}_{Y_a \sim D_a}\Big[\max(\max_{a\in S} Y_a, \max_{j\in A\backslash S} \mu_j)\Big] \tag{5}$$

Algorithm 1 Greedy probing with a fixed number of probes

```
Input: $A$: a set of $N$ arms; $\mathbb{D}$: reward distributions of all the arms; $k$: a fixed budget for probing
Output: $S^{\mathrm{pr}}$: the probing set
1: $S \leftarrow \emptyset$
2: for $i = 1$ to $k$ do
3:     $a \leftarrow \operatorname{argmax}_{a\in A\setminus S} (f(S \bigcup\{a\}) - f(S))$
4:     $S \leftarrow S \cup \{a\}$
5: $S^{pr} \leftarrow S$
6: $a \leftarrow \operatorname{argmax}_{a\in A} \mu_a$
7: if $f(S) < \mu_a$ then $S^{pr} \leftarrow \emptyset$
```

Algorithm 2 Offline non-adaptive probing

```
Input: $A$: a set of $N$ arms; $K$: probing budget; $\mathbb{D}$: reward distributions for all the arms
Output: $S^{\mathrm{pr}}$: the probing set
1: for $k = 0$ to $K - 1$ do
2:     $S_k \leftarrow \emptyset$
3: for $k = 1$ to $K - 1$ do
4:     $a \leftarrow \operatorname{argmax}_{a\in A\setminus S_{k-1}} (f(S_{k-1} \bigcup\{a\}) - f(S_{k-1}))$
5:     $S_k \leftarrow S_{k-1} \cup \{a\}$
6: $S^{\mathrm{pr}} \leftarrow \operatorname{argmax}_{k} \frac{K-k}{K} f(S_k)$
7: $a \leftarrow \operatorname{argmax}_{a\in A} \mu_a$
8: if $f(S^{pr}) < \mu_a$ then $S^{pr} \leftarrow \emptyset$
```

We first note that a similar problem has been considered in [24], where given a budget k, the goal is to find a subset $S \subseteq A$ with $|S| = k$ such that $g(S)$ is maximized. This problem is NP-hard, yet a simple greedy algorithm gives a constant approximation factor (see Algorithm 1). The key idea is to consider a new objective function

$$f(S) = \mathbb{E}_{Y_a\sim D_a}\Big[\max_{a\in S} Y_a\Big] \tag{6}$$

This is equivalent to always choosing a probed arm with maximum reward to play without considering the unprobed arms. $f(S)$ has the following nice properties:

- Monotonicity: $f(S) \le f(T)$ for any $S \subseteq T \subseteq A$;
- Submodularity: $f(S \cup \{a\}) - f(S) \ge f(T \cup \{a\}) - f(T)$ for $S \subseteq T \subseteq A$ and $a \notin T$.

The two properties together ensure that a simple greedy algorithm obtains an approximation factor of $\alpha = 1 - 1/e$ to the problem $\max_{S\subseteq A,|S|=k} f(S)$ [28] . The greedy algorithm starts with $S = \emptyset$ and in each step, a new arm that maximizes the marginal improvement $f(S \cup \{a\}) - f(S)$ is added to S until $|S| = k$ (see lines 1-4 in Algorithm 1). To obtain a solution to the problem $\max_{S\subseteq A,|S|=k} g(S)$, we simply take the greedy solution S found and compare $f(S)$ with $\max_{a\in A} \mu_a$ and take the maximum (lines 5-7 in Algorithm 1).

We then present Algorithm 2 that adapts Algorithm 1 to our problem of maximizing $R(S)$. The key difference is that Algorithm 2 maintains the greedy solution S_k for each $k \in \{0, ..., K - 1\}$ and identifies the best solution among them (with respect to $\frac{K-|S|}{K} f(S)$), which is then compared with the maximum μ_a among all the arms. We establish the following approximation ratio of our algorithm.

Theorem 1: Let $S^\star$ denote the optimal probing set with respect to the reward distributions $\mathbb{D}$. Algorithm 2 outputs a subset S^{pr} such that $R(S^{\mathrm{pr}}) \ge \alpha R(S^\star)$ where $\alpha = \frac{e-1}{2e-1}$.

Proof: Let S_k be the k-th probing set found by Algorithm 2 line 5 and S_0 be the one that maximizes $\frac{K-k}{K} f(S_k)$ (Algorithm 2 line 6). We have

$$\begin{aligned}
R(S^\star) &= \frac{K-|S^\star|}{K}\mathbb{E}_{Y_a\sim D_a}\Big[\max(\max_{a\in S^\star} Y_a, \max_{j\in A\setminus S^\star} \mu_j)\Big] \\
&\le \frac{K-|S^\star|}{K}\mathbb{E}_{Y_a\sim D_a}\Big[(\max_{a\in S^\star} Y_a + \max_{j\in A} \mu_j)\Big] \\
&\le \frac{K-|S^\star|}{K}\mathbb{E}_{Y_a\sim D_a}\Big[\max_{a\in S^\star} Y_a\Big] + \max_{j\in A} \mu_j \\
&= \frac{K-|S^\star|}{K} f(S^\star) + \max_{j\in A} \mu_j \\
&\overset{a}{\le} \frac{K-|S^\star|}{K} \frac{e}{e-1} f(S_{|S^\star|}) + \max_{j\in A} \mu_j \\
&\overset{b}{\le} \frac{K-|S_0|}{K} \frac{e}{e-1} f(S_0) + \max_{j\in A} \mu_j \\
&\overset{c}{\le} \frac{e}{e-1} R(S^{pr}) + R(S^{pr}) \\
&= \frac{2e-1}{e-1} R(S^{pr})
\end{aligned}$$

where (a) is due to the fact that $f(\cdot)$ is monotone and submodular, (b) follows from the definition of S_0, and (c) is due to the way S^{pr} is defined. ■

Since for each $k \in \{1, ..., K - 1\}$, the algorithm evaluates $f(S_{k-1} \bigcup\{a\})$ for each $a \in A\setminus S_{k-1}$, it involves $O(NK)$ evaluation of $f(\cdot)$ in total, which determines its time complexity.

Remark 1: For Bernoulli arms, Algorithm 1 is optimal ($\alpha = 1$). We omit the proof but note that it can be established using a similar argument as the proof of Theorem 2 in the adaptive setting.

Remark 2: The above result assumes that $f(S)$ can be accurately estimated for any S. For general distributions, $f(S)$ needs to be estimated using samples generated by $\mathbb{D}$. Thus, the performance of the algorithm will be affected by the number of samples used.

B. The offline problem with adaptive probing

We then consider the more challenging setting with adaptive probing. The offline problem of finding the optimal π_1 can be formulated as a finite horizon Markov decision process (MDP) by viewing the set of probed arms and their outcomes (i.e., $\{a_1, ..., a_i, Y_{a_1}, ...Y_{a_i}\}$ by the i-th probing step) as a state and the arm to be probed next or stopping probing as an action. The main challenge, however, is introduced by the combinatorial nature of the problem, leading to large state and action spaces. To derive a more scalable solution, we first observe that once the ordering of the set of arms to be probed is fixed, then an adaptive probing policy simplifies to a binary decision (whether to continue probing or not) at each probing step given the current observations. As we will see for Bernoulli arms (that is, $D_{a|x}$ is a Bernoulli distribution for all context-arm pairs), the natural ordering of probing the arms with larger mean values first is optimal. Although it is difficult to identify the optimal ordering for general reward distributions, the greedy ordering based on marginal improvement used in Algorithms 1 and 2 is a natural choice and achieves good performance as we show in the simulations.

Below we assume that the set of arms A has been sorted according to a certain ordering scheme and let a_i denote the i-th arm on the list. We can then formulate the adaptive probing problem as an MDP with action space $= \{0, 1\}$ with 1 indicating probing the next arm on the list and 0 indicating stopping probing. Further, the state at the i-th probing step can be defined as $(Y_1, ...Y_i)$ as the set of arms probed is determined by i. To further simplify the problem, a key observation is that instead of keeping the complete outcomes, it suffices to maintain $s_i \triangleq \max\{Y_1, ..., Y_i\}$ and define (i, s_i) as the state of the i-th probing step. This can be proved by showing that the MDP using the compressed state space has the same optimal state-action values as the full MDP. We skip the formal proof due to the space constraint.

With the above discussion, the adaptive probing problem can be reduced to a finite horizon MDP with a state space of size $K \times |\mathcal{Y}|$ (the number of possible probing outcomes for each arm) and an action space of size 2. Further, an adaptive probing policy π_1 is a mapping from each state (k, s_k) to the action space. Let $V_i^*(s_i)$ denote the optimal state value at the i-th probing step under state s_i (the maximum value observed so far). We have the following Bellman optimality equation:

$$V_i^\star(s_i) = \max\left\{R_i(s_i), \sum_{s_{i+1}\in\mathcal{Y}} \Pr(s_{i+1}|s_i)V_{i+1}^\star(s_{i+1})\right\} \quad (7)$$

where $R_i(s_i) \triangleq \frac{K-i}{K}\max(s_i, \max_{j>i}\mu_j)$ is the value obtained if we stop probing after step i, and $\Pr(s_{i+1}|s_i)$ is the probability that the next state becomes s_{i+1} when the current state is s_i and probing continues, which can be derived from $\mathbb{D}$. It is clear that $V_K^\star(s_K) = 0$ for any $s_K \in \mathcal{Y}$. The optimal value (under the fixed ordering) is then given by $V_0^\star(0)$ and the optimal probing action under each state can be derived by comparing the two terms in (7).

Based on Equation (7), the optimal probing policy (under a fixed ordering of arms) can be computed using a dynamic programming based solution (see Algorithm 3). The algorithm first sorts all the arms according to a certain ordering scheme. In this work, we assume that the arms are sorted greedily in terms of the marginal improvement with respect to $f(\cdot)$ as in the non-adaptive setting (i.e., lines 2-4 in Algorithm 1 with $k = K$). After initializing the values at the K-th probing step, the algorithm computes $V_k^\star(s_k)$ and the policy $\pi_1((k, s_k)) \in \{0, 1\}$ for $k = K - 1, ..., 1$ recursively and for $s_k \in \mathcal{Y}$, using Equation (7).

Algorithm 3 Offline adaptive probing via dynamic programming

```
Input: $A$: a set of $N$ arms; $K$: the probing budget; $\mathbb{D}$: distribution for all the arms.
Output: $v$: state value function, $\pi_1$: an adaptive probing policy
1: Sort the set of arms greedily in terms of marginal improvement with respect to $f(\cdot)$
2: $v_K(s_K) \leftarrow 0$ for all $s_K \in \mathcal{Y}$.
3: for $k = K - 1$ to 1 do
4:     for each outcome $s_k \in \mathcal{Y}$ do
5:         $R_k(s_k) \leftarrow \frac{K-k}{K}\max(s_k, \max_{j>k}\mu_j)$
6:         $v_k(s_k) \leftarrow \max\{R_k(s_k), \mathbb{E}_{s_{k+1}\in\mathcal{Y}}[v_{k+1}(s_{k+1})]\}$
7:         $\pi_1(k, s_k) \leftarrow 1$
8:         if $v_k(s_k) = R_k(s_k)$ then
9:             $\pi_1(k, s_k) \leftarrow 0$
10: $v_0(0) = \max\{\max_{a\in A}\mu_a, \mathbb{E}_{s_1\in\mathcal{Y}}[v_1(s_1)]\}$
```

Since for each $k \in \{1, ..., K-1\}$ and each possible outcome in $\mathcal{Y}$, the computation of $V_k(s_k)$ involves taking the maximum over unprobed arms (line 5) and taking the expectation over all possible outcomes in the next step (line 6), the time complexity of the algorithm is $O(|\mathcal{Y}|K(N + |\mathcal{Y}|))$.

C. Adaptive probing for Bernoulli arms

Although it is difficult to derive the performance bound of Algorithm 3 for general $\mathbb{D}$, we now show that the algorithm is optimal for Bernoulli arms where D_a is a Bernoulli distribution for all $a \in A$. To this end, we first show that it is optimal to probe the set of arms according to their mean reward μ_a non-increasingly (note that for Bernoulli arms, this ordering is equivalent to the default ordering we use in Algorithm 3). To see this, we first note that for Bernoulli arms, whenever $Y_{a_i} = 1$ is observed, the decision maker can stop probing and play a_i. With this observation, any adaptive probing policy can be defined as follows. First pick a sequence $\vec{S} = (a_1, ..., a_k)$ of k arms; (2) probe arms following the sequence until either a value of one is observed or k continuous zeros are observed. Let a_{k+1} denote the arm played in the latter case (i.e., the one with maximum μ in the remaining arms). Thus, any adaptive policy can be fully characterized by an arbitrary sequence $\vec{S}$ of any length $k \in \{0, K-1\}$. Let $R(\vec{S})$ denote the expected value by following $\vec{S}$.

Theorem 2: For Bernoulli arms, it is optimal to probe the set of arms in their mean reward μ non-increasingly.

Proof: Consider a general adaptive policy defined by $\vec{S}$ with k arms. We can derive its expected value as follows.

$$\begin{aligned} R(\vec{S}) \overset{a}{=} &\mu(a_1)\frac{K-1}{K} + (1-\mu(a_1))\mu(a_2)\frac{K-2}{K} \\ &+ (1-\mu(a_1))(1-\mu(a_2))\mu(a_3)\frac{K-3}{K} + ... \\ &+ (1-\mu(a_1))...(1-\mu(a_{k-1}))\mu(a_k)\frac{K-k}{K} \end{aligned}$$

$$
\begin{aligned}
&+ (1-\mu(a_1))...(1-\mu(a_k))\mu(a_{k+1})\frac{K-k}{K}\\
&\overset{b}{=}\frac{K-k}{K}[1-\prod_{i=1}^{k+1}(1-\mu(a_i))]\\
&+\frac{1}{K}[1-\prod_{i=1}^{k-1}(1-\mu(a_i)]\\
&+\frac{1}{K}[1-\prod_{i=1}^{k-2}(1-\mu(a_i))]+...\\
&+\frac{1}{K}[1-(1-\mu(a_1))(1-\mu(a_2))]\\
&+\frac{1}{K}\mu(a_1)
\end{aligned}
$$

where (a) follows from the definition of the policy and (b) is based on the following observation. Note that the first term in the second expression corresponds to the expected reward obtained when at least one of the $k+1$ arms (including the played one) gives a one, assuming that the unit reward is $(K-k)/K$. However, when an arm is sampled to one before step k, their rewards are higher, which are compensated by the following terms. The second expression makes it clear that is always better to choose arms with larger mean values for probing and play. To further show that it is optimal to probe arms in a non-increasingly order, assume that $\mu_{a_i} < \mu_{a_{i+1}}$ for some i in the sequence. Let $\vec{S}'$ denote the sequence constructed from $\vec{S}$ by switching a_i and a_{i+1} while keeping the ordering of other arms. From the first expression above, we have

$$
\begin{aligned}
R(\vec{S}')-R(\vec{S}) &= \prod_{j=1}^{i-1}(1-\mu_{a_j})\frac{K-i}{K}[\mu_{a_{i+1}}-\mu_{a_i}]\\
&+\prod_{j=1}^{i-1}(1-\mu_{a_j})\frac{K-(i+1)}{K}\Big[(1-\mu_{a_{i+1}})\mu_{a_i}\\
&\qquad\qquad -(1-\mu_{a_i})\mu_{a_{i+1}}\Big]\\
&=\prod_{j=1}^{i-1}(1-\mu_{a_j})\frac{K-i}{K}[\mu_{a_{i+1}}-\mu_{a_i}]\\
&+\prod_{j=1}^{i-1}(1-\mu_{a_j})\frac{K-(i+1)}{K}\Big[\mu_{a_i}-\mu_{a_{i+1}}\Big]\\
&\geq 0
\end{aligned}
$$

■

From Theorem 2, it is optimal to probe arms with non-decreasing μ. Thus, Algorithm 3 is optimal for Bernoulli arms.

V. The Online Setting

In this section, we study the online setting where $\mathbb{D}$ is unknown a priori. We show that the offline algorithms developed in the previous section can be integrated with the combinatorial bandit framework in [7] to solve the joint probing and play problem in the online setting. We further establish the regret of our solution under general arm reward distributions when probing is non-adaptive and for Bernoulli arms when probing is adaptive.

A. A combinatorial contextual bandit based algorithm

We first present our algorithm for the online setting (see Algorithm 4). Our solution is adapted from the combinatorial bandit algorithm in [7], which was originally developed for general reward functions without considering context. In contrast, we consider the contextual setting and further distinguish the different reward functions introduced by non-adaptive and adaptive probing.

The algorithm (Algorithm 4) first normalizes the context space to X' (see the discussion below). For each context-arm pair $a \in A$ and $x \in X'$, the algorithm maintains the number of times that arm a has been probed or played under context x as $n_{a|x}$, and an approximation $\tilde{F}_{a|x}$ of the true cumulative distribution function (CDF) $F_{a|x}$ of $D_{a|x}$. Note that $F_{a|x}$ with support $\{y_{a|x,1}, y_{a|x,2}, ..., y_{a|x,l}\}$ is defined as $F_{a|x}(y_{a,j}) = \Pr_{Y_{a|x}\sim D_{a|x}}[Y_{a|x} \leq y_{a|x,j}]$. Since $F_{a|x}$ is unknown, we approximate it by $\tilde{F}_{a|x}(y)$, which is computed as the fraction of the observed outcomes from arm a under context x that are no larger than y.

Algorithm 4 plays each arm once under each context, and updates $n_{a|x}$ and $\tilde{F}_{a|x}$. After that, it maintains a lower bound $\underline{F}_{a|x}$ of $\tilde{F}_{a|x}$ using a UCB type of technique (lines 11-13). Let $\underline{D} = \underline{D}_{1|x} \times \underline{D}_{2|x} \times \cdots \times \underline{D}_{N|x}$ denote the corresponding distributions across all arms. The algorithm then distinguishes two cases. When non-adaptive probing is used, it invokes Algorithm 2 with $\underline{D}$ as the input. Otherwise, it invokes Algorithm 3. In each case, a probing set S^{pr} can be derived by following the policy obtained. We then collect samples for arms in S^{pr}. For arms not in S^{pr}, we estimate their mean reward using $\underline{D}_{a|x}$. We then identify the arm to play according to the play strategy π_2 given in Observation 1, and update $n_{a|x}$ and $\tilde{F}_{a|x}$ based on the probing and play results (lines 23-28). It is crucial to note that the play decision is based on $\underline{D}$ for unprobed arms (line 24) rather than the true distributions $\mathbb{D}$, which introduces another level of suboptimality.

B. Dealing with large context space

When the context space is large or continuous, we group similar context together. Given $X = [0, 1]$, let X' denote the ϵ-uniform mesh of X. When a new context $x_t \in X$ is received, it is mapped to the closet point in X' according to

$$\bar{x}_t = \operatorname*{argmin}_{x\in X'} |x - x_t|$$

where we choose the smaller value when there is a tie.

To understand the loss introduced by the compression of the context space, let $(S^\star(x), a^\star(x))$ denote the optimal offline solution under context x and $r(S^\star(x), a^\star(x)|x)$ its expected value when the solution is applied under context x. We define the discretization error as

$$DE(T) = \sum_{t=1}^{T}[r(S^\star(x_t), a^\star(x_t)|x) - r(S^\star(\bar{x}_t), a^\star(\bar{x}_t)|\bar{x})]$$

We have the following lemma:

Algorithm 4 Online contextual bandit for joint probing and play

```
Input: $A$: a set of $N$ arms; $X$: a set of context; $K$: probing budget; $T$: time horizon
$X' \leftarrow$ the normalization of context space $X$.
$n_{a|x} \leftarrow 0$, $\tilde{F}_{a|x} \leftarrow 0$ for all $a \in A, x \in X'$
for $t = 1, 2, ..., T$ do
    Receive context $x_t \in X$
    $x \leftarrow \bar{x}_t$
    if there is an arm $a$ never played under $x$ then
        Play arm $a$, get $Y_{a|x}$
        $n_{a|x} = n_{a|x} + 1$
        Update $\tilde{F}_{a|x}$ based on $Y_{a|x}$
        continue
    For each arm $a \in A$, let $\underline{D}_{a|x}$ be a distribution whose CDF $\underline{F}_{a|x}$ is given by
    $\underline{F}_{a|x} = \begin{cases} \max\{\tilde{F}_{a|x}(y) - \sqrt{\frac{3\ln t}{2n_{a|x}}}, 0\}, & 0 \le y < 1 \\ 1, & y = 1 \end{cases}$
    $\underline{D} = \underline{D}_{1|x} \times \underline{D}_{2|x} \times \cdots \times \underline{D}_{N|x}$
    if probing is non-adaptive then
        $S_t^{\text{pr}} \leftarrow$ Algorithm2$(A, K, \underline{D})$
        Probe each arm $a \in S_t^{\text{pr}}$ and get $Y_{a|x}$
        PLAY AND UPDATE$(S_t^{\text{pr}}, \{Y_{a|x} : a \in S_t^{\text{pr}}\}, x)$
    if probing is adaptive then
        $\pi_1 \leftarrow$ Algorithm3$(A, K, \underline{D})$
        Probe arms according to $\pi_1$ and get their outcomes
        $S_t^{\text{pr}} \leftarrow$ the set of arms probed
        PLAY AND UPDATE$(S_t^{\text{pr}}, \{Y_{a|x} : a \in S_t^{\text{pr}}\}, x)$
function PLAY AND UPDATE$(S_t^{\text{pr}}, Y_t, x)$:
    Play $a_t^{\text{pl}} \leftarrow \underset{a}{\text{argmax}}\{\{Y_{a|x} : a \in S_t^{\text{pr}}\}, \{\mathbb{E}(\underline{D}_{a|x}) : a \in A \backslash S_t^{\text{pr}}\}\}$
    if $a_t^{\text{pl}} \notin S_t^{\text{pr}}$ then get $Y_{a_t^{\text{pl}}|x}$
    for each $j \in \{S_t^{\text{pr}}, a_t^{\text{pl}}\}$ do
        $n_{j|x} = n_{j|x} + 1$
        Update $\tilde{F}_{j|x}$ based on $\{y_{j|x,1}, y_{j|x,2}, ..., y_{j|x,l}\}$ and $Y_{j|x}$
```

Lemma 1:

$$DE(T) \le \epsilon LT$$

Proof: For each round t and the respective context x_t, we have

$$\begin{aligned} r(S^\star(\bar{x}_t), a^\star(\bar{x}_t)|\bar{x}_t) &\ge r(S^\star(x_t), a^\star(x_t)|\bar{x}_t) \\ &\ge r(S^\star(x_t), a^\star(x_t)|x_t) - \epsilon L \end{aligned}$$

where the first inequality is because that $(S^\star(\bar{x}_t), a^\star(\bar{x}_t))$ is the optimal solution under context $\bar{x}_t$ and the second inequality is due to the definition of $\bar{x}_t$ and the Lipschtz-continuity assumption (4). Summing this up over all rounds t, we obtain

$$DE(T) \le \epsilon LT$$

■

With the above lemma, we can bound the total regret as

$$\mathsf{Reg}_\alpha(T) \le \alpha\epsilon LT + R_\alpha(T)$$

where $R_\alpha(T)$ is defined as

$$R_\alpha(T) = \sum_{t=1}^{T}(\alpha r(S_t^\star, a^\star|\bar{x}_t) - r(S_t, a_t|\bar{x}_t))$$

C. Theoretical analysis of the online algorithm

In this section, we analyze the regret of Algorithm 4 for the non-adaptive probing setting under a general $\mathbb{D}$ and as well as the adaptive probing setting for Bernoulli arms. To simplify the notation, we let $R(S, Y, a|x)$ denote the value obtained given the probing and play decision (S, a), the probing result Y, and context x, and $r(S, a|x)$ its expectation over Y in both settings. It is understood that in the adaptive setting, S is interpreted as a sequence with a pre-defined ordering (which is without loss of generality for Bernoulli bandits according to Theorem 2). Note that although in the offline setting, the optimal a is completely determined by S and Y, the online algorithm picks a according to $\underline{D}_{a|x}$. We include a in the value functions to emphasize this fact. In both cases, we can show that $R(S, Y, a|x)$ satisfies the following properties required by [7].

- There exists $M > 0$ such that for any S, Y, and x, we have $0 \le R(S, Y, a|x) \le M$.
- $R(S, Y|x) \le R(S, Y'|x)$ for any S, x, Y and Y' such that $Y_i \le Y'_i$ for every $a_i \in S$.

The following definitions are adapted from [7]. For any $S \subseteq A$ and $a \in A$, we define

$$\Delta_{S,a|x} = \max\{\alpha r(S^\star, a^\star|x) - r(S, a|x), 0\}$$

Let $E_B(x)$ denote the set of arms that are contained in at least one suboptimal probing decision S under context x. For each $i \in E_B(x)$, we define

$$\Delta(i, \min|x) = \min\{\Delta_{S,a|x}|S \subseteq A, a \in A, i \in S\}$$

$$L_e = \max\{|E_B(\bar{x})||\bar{x} \in X'\}$$

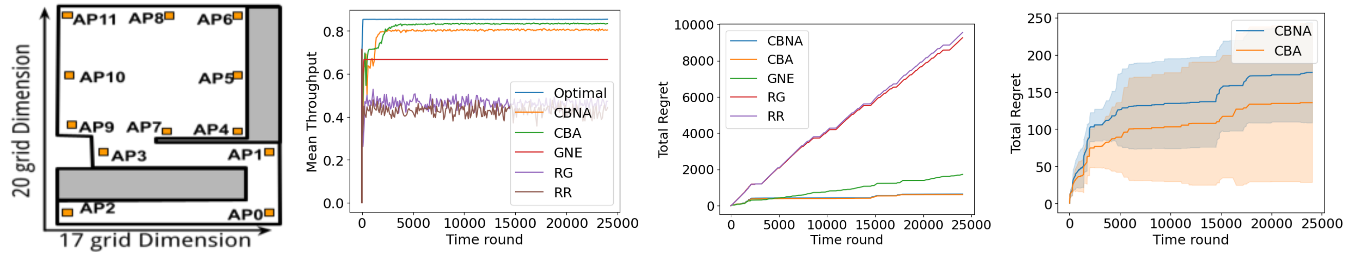
and

$$\Delta(\min) = \min\{\Delta(i, \min|\bar{x})|i \in E_B(\bar{x}), \bar{x} \in X'\}$$

With the above definitions, we can prove the following regret bounds for both non-adaptive probing under general distributions and adaptive probing under Bernoulli distributions.

Theorem 3: Algorithm 4 has the following distribution-dependent upper bound on the α-approximation regret in T rounds

$$2[M^2(K-1)L_e\frac{2136}{\Delta(\min)}\ln T + (\frac{\pi^2}{3} + 1)\alpha MN]^{\frac{1}{2}}(\alpha LT)^{\frac{1}{2}} \tag{8}$$



(a) (b) (c) (d)

Fig. 1: (a) measurement setup (in a student hall); (b) mean throughput for Bernoulli arms: $K = 10, N = 10$, single context; (c) accumulative regret for Bernoulli arms (w.r.t. the optimal offline adaptive algorithm): $K = 10, N = 10, \frac{1}{\epsilon} = 8$; (d) average accumulative regret for arms with general reward distributions (w.r.t. the optimal offline adaptive algorithm): $K = 4, N = 4, \frac{1}{\epsilon} = 8$.

and the following distribution-independent upper bound on the α-approximation regret in T rounds

$$2[93M\sqrt{N(K-1)T\ln T}+(\frac{\pi^2}{3}+1)\alpha MN]^{\frac{1}{2}}(\alpha LT)^{\frac{1}{2}}$$

For Bernoulli arms, the above bounds can be simplified by taking $\alpha = 1$ since the offline algorithms are optimal for both non-adaptive and adaptive settings for Bernoulli arms.

The detailed proof of Theorem 3 can be found in our online technical report [29].

VI. Evaluation

In this section, we evaluate the performance of our algorithms using simulations with channel and mobility traces collected from a real testbed.

A. Baselines

We consider the following algorithms in the evaluation. For non-adaptive probing, we apply Algorithm 4 together with Algorithm 2 and call it the contextual bandit with non-adaptive probing algorithm (CBNA). For adaptive-probing, we apply Algorithm 4 together with Algorithm 3, which is referred as the contextual bandit with adaptive probing algorithm (CBA). We use 50 samples to evaluate $f(\cdot)$ in Algorithm 2.

We compare our algorithms with four baselines: (a) Optimal, which is the optimal offline solution under adaptive probing and is obtained through an exhaustive search over all possible ordering of arms in Algorithm 3. (b) GNE (greedy probing and greedy play without exploration in the play), which follows CBNA except that at the play stage, $\underline{F}_{a|x_t}$ is replaced with $\tilde{F}_{a|x_t}$ so that no exploration is considered in play decisions. (c) RG (randomly probing and greedy play) where the decision maker randomly selects a k and probes k arms randomly selected. The play policy is the same as CBNA. (d) RR (random probing and random play) where the decision maker randomly selects a k and probes k arms randomly selected. It then plays a randomly selected arm.

B. Evaluation Settings

We collect channel traces from a real testbed using an 802.11ad router and laptop, as well as a commercially available mmWave channel simulator (Remcom Insite [30]) to evaluate the system. We collect SNR traces with actual testbeds in 250 different locations in the student hall (Scenario Fig. 1a) with a particle size of 0.8m. In the student hall, 12 Airfide [31] 802.11ad APs are installed, each with a 64-sector 8-phased array antenna. Clients are Acer TravelMate-648 [32] laptops with a single phased array and 36 sectors. We modified the open-source driver [33] on both APs and the client to extract SNR and beamforming information. At each grid, the client conducted beamforming with all APs and recorded the per-sector SNR. We repeated this measurement ten times at each grid and stored the average SNR for different Tx and Rx sectors in a 64×36 matrix.

We use the 802.11ad [34] MCS-SNR table to map the SNR and get the average throughput of the connection (as in [35]) after obtaining the signal strength channel trace from the testbed and Remcom channel simulator. It is based on the best Tx/Rx sector pair of links discovered through the beamforming process. In each context x (client location), we normalize the average throughput of each AP a as $\mu(a|x)$ and consider two settings for link rate distributions. In the Bernoulli setting, the probability of getting a reward of 1 from arm a under context x is $\mu(a|x)$, and the probability of getting zero reward is $1 - \mu(a|x)$. We also consider a more general setting where the link rates are supported on $\mathcal{Y} = [0, \frac{1}{3}, \frac{2}{3}, 1]$. For each AP a and location x, we randomly assign the probabilities to the four points so that the mean link rate is equal to $\mu(a|x)$.

For mobility traces, we choose 15-80 customers randomly located and configure their walking patterns by observing the typical walking behavior of a room. We assume that all clients have a walking speed of $1m/s$. At any time, a client is in one of the grids where the channel is measured by the testbed or channel simulator as described above. We choose 10 clients' traces in the simulation, where each trace has 80 steps and each step includes 30 time rounds. We then implemented all the algorithms of AP selection with python based on the collected traces and showed the results below.

C. Results

We evaluate all the algorithms with different probing budget K and number of APs N. We randomly select N APs and compare the algorithms in three different settings.

In Fig. 1b, we consider the Bernoulli setting under a single context, where we randomly pick a location and keep it the same all the time (without mobility). We plot the mean throughput averaged over 100 time rounds. We observe that although GNE and CBNA obtain best performance in the first ~ 2000 time rounds, CBA surpasses them after that and obtains higher average throughput. Additionally, we observe that CBNA and CBA consistently outperform the others. Also, the performance of GNE is most stable due to the lack of exploration.

In Fig. 1c, we consider the Bernoulli setting with multiple contexts. We evenly divide the room into a grid of 8 cells and treat each of them as a context. For each location x visited by a client, we map x to the center of its closest cell to get $\bar{x}$. Fig. 1c plots the expected accumulative expected regret of each baseline with respect to Optimal. We find that the total expected regret of CBNA and CBA is less than other baselines. We further notice that in this figure, there are two jumps around time rounds 15,000 and 17,000, respectively, which correspond to the two starting points that have not been explored enough before, such as the bottom part where most APs are blocked. In addition, we find that for Bernoulli arms, the regret of CBNA and CBA is relatively close, and CBA is slightly better than CBNA.

In order to further distinguish the two algorithms, we plot the results for the general distribution setting described above in Fig. 1d. Here we again group locations into 8 context cells as in Fig. 1c. We run both algorithms five times with different random seeds. The area between the upper blue curve and lower gray curve indicates the standard deviation of CBNA, and the area between the upper grey curve and the lower yellow curve indicates the standard deviation of CBA. We observe that CBA always performs better than CBNA in the simulated scenarios, indicating the importance of considering adaptive probing.

VII. Conclusion

In this paper, we propose an online learning based joint beamforming and scheduling framework for throughput optimization for mmWave WLANs. We model the problem as a contextual bandit problem with joint probing and play. In the offline setting with known link rate distributions, we develop an approximation algorithm when the probing decision is non-adaptive and a dynamic programming based algorithm for the adaptive setting. Both algorithms are optimal for Bernoulli link rates and we establish an approximation factor of the former under general link rate distributions. We further develop a combinatorial contextual bandit algorithm for the online setting and prove its regret. Our algorithms are validated through simulations with channel and mobile traces collected from a real testbed.

Acknowledgment

This work was supported in part by NSF grant CNS-1816943.

References

[1] M. J. Neely, *Stochastic Network Optimization with Application to Communication and Queueing Systems*. Springer, 2010.

[2] T. Stahlbuhk, B. Shrader, and E. Modiano, "Learning algorithms for scheduling in wireless networks with unknown channel statistics," *Ad Hoc Networks*, vol. 85, no. 2019, pp. 131–144, 2019.

[3] F. Li, J. Liu, , and B. Ji, "Combinatorial sleeping bandits with fairness constraints," in *IEEE INFOCOM*, 2019.

[4] H. Gupta, A. Eryilmaz, and R. Srikant, "Link rate selection using constrained thompson sampling," in *IEEE INFOCOM*, 2019.

[5] A. Slivkins, "Contextual bandits with similarity information," in *COLT*, 2011.

[6] D. Zhang, P. Selvam, P. H. Pathak, and Z. Zheng, "Networked beamforming in dense mmwave wlans," in *HotMobile*, 2022.

[7] W. Chen, W. Hu, F. Li, J. Li, Y. Liu, and P. Lu, "Combinatorial multi-armed bandit with general reward functions," in *Advances in Neural Information Processing Systems*, 2016.

[8] A. Bhaskara, S. Gollapudi, K. Kollias, and K. Munagala, "Adaptive probing policies for shortest path routing," in *NeurIPS*, 2020.

[9] P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite-time analysis of the multiarmed bandit problem," *Machine learning*, vol. 47, no. 2, pp. 235–256, 2002.

[10] J. Zhu, R. Sandhu, and J. Liu, "A distributed algorithm for sequential decision making in multi-armed bandit with homogeneous rewards," in *2020 59th IEEE Conference on Decision and Control (CDC)*, pp. 3078–3083, IEEE, 2020.

[11] P. Landgren, V. Srivastava, and N. E. Leonard, "Distributed cooperative decision making in multi-agent multi-armed bandits," *Automatica*, vol. 125, p. 109445, 2021.

[12] S. Bubeck and N. Cesa-Bianchi, "Regret analysis of stochastic and nonstochastic multi-armed bandit problems," *Foundations and Trends® in Machine Learning*, vol. 5, no. 1, pp. 1–122, 2012.

[13] J. Gittins, K. Glazebrook, and R. Weber, *Multi-armed bandit allocation indices*. John Wiley & Sons, 2011.

[14] T. Stahlbuhk, B. Shrader, and E. Modiano, "Learning algorithms for scheduling in wireless networks with unknown channel statistics," *Ad Hoc Networks*, vol. 85, pp. 131–144, 2019.

[15] A. Asadi, S. Müller, G. H. Sim, A. Klein, and M. Hollick, "FML: Fast Machine Learning for 5G mmWave Vehicular Communications," in *IEEE INFOCOM*, 2018.

[16] H. Gupta, A. Eryilmaz, and R. Srikant, "Link rate selection using constrained thompson sampling," in *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*, pp. 739–747, IEEE, 2019.

[17] T. Xu, D. Zhang, P. H. Pathak, and Z. Zheng, "Joint ap probing and scheduling: A contextual bandit approach," in *IEEE MILCOM*, 2021.

[18] M. L. Weitzman, "Optimal search for the best alternative," *Econometrica*, vol. 47, no. 3, pp. 641–654, 1979.

[19] K. Munagala, S. Babu, R. Motwani, and J. Widom, "The pipelined set cover problem," in *International Conference on Database Theory*, pp. 83–98, Springer, 2005.

[20] A. Deshpande, L. Hellerstein, and D. Kletenik, "Approximation algorithms for stochastic submodular set cover with applications to boolean function evaluation and min-knapsack," *ACM Transactions on Algorithms (TALG)*, vol. 12, no. 3, pp. 1–28, 2016.

[21] Z. Liu, S. Parthasarathy, A. Ranganathan, and H. Yang, "Near-optimal algorithms for shared filter evaluation in data stream systems," in *ACM SIGMOD*, 2008.

[22] S. Guha, K. Munagala, and S. Sarkar, "Optimizing transmission rate in wireless channels using adaptive probes," in *ACM Sigmetrics*, 2006.

[23] T. Shu and M. Krunz, "Throughput-efficient sequential channel sensing and probing in cognitive radio networks under sensing errors," in *Proceedings of the 15th annual international conference on Mobile computing and networking*, pp. 37–48, 2009.

[24] A. Goel, S. Guha, and K. Munagala, "Asking the right questions: Model-driven optimization using probes," in *PODS*, 2006.

[25] S. Guha and K. Munagala, "Model-driven optimization using adaptive probes," in *Soda*, vol. 7, pp. 308–317, Citeseer, 2007.

[26] T. He, D. Goeckel, R. Raghavendra, and D. Towsley, "Endhost-based shortest path routing in dynamic networks: An online learning approach," in *IEEE INFOCOM*, 2013.

[27] B. Li, P. Yang, J. Wang, Q. Wu, S. Tang, X.-Y. Li, and Y. Liu, "Almost optimal dynamically-ordered channel sensing and accessing for cognitive networks," *IEEE Transactions on Mobile Computing*, vol. 13, no. 10, pp. 2215–2228, 2013.

[28] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher, "An analysis of approximations for maximizing submodular set functions—i," *Mathematical programming*, vol. 14, no. 1, pp. 265–294, 1978.

[29] T. Xu, D. Zhang, and Z. Zheng, "Online Learning for Adaptive Probing and Scheduling in Dense WLANs." Technical Report, available online at https://arxiv.org/abs/2212.13585.

[30] "Remcom Wireless InSite 3D Wireless Propagation Software." https://www.remcom.com/wireless-insite-em-propagation-software.

[31] "Airfide afn2200." https://airfidenet.com/.

[32] Acer TM-P648, "https://www.acer.com/ac/en/US/press/2016/175243."

[33] wil6210, "https://wireless.wiki.kernel.org/en/users/drivers/wil6210."

[34] IEEE P802.11adTM/D4.0, "Part 11: Wlan mac/phy enhancements for very high throughput in the 60 ghz band," 2012.

[35] D. Zhang, M. Garude, and P. H. Pathak, "Mmchoir: Exploiting joint transmissions for reliable 60ghz mmwave wlans," in *ACM Mobihoc*, 2018.