Variance-Adaptive Algorithm for Probabilistic Maximum Coverage Bandits with General Feedback

Xutong Liu[†1], Jinhang Zuo[†2], Hong Xie[*3], Carlee Joe-Wong[2], John C.S. Lui[1]
[1]The Chinese University of Hong Kong, [2]Carnegie Mellon University,
[3]Chongqing Institute of Green and Intelligent Technology, Chinese Academy of Science
Email: {liuxt, cslui}@cse.cuhk.edu.hk, xiehong2018@foxmail.com, {jzuo, cjoewong}@andrew.cmu.edu

***Abstract*—Probabilistic maximum coverage (PMC) is an important problem that can model many network applications, including mobile crowdsensing, network content delivery, and dynamic channel allocation, where an operator chooses nodes in a graph that can probabilistically cover other nodes. In this paper, we study PMC under the online learning context: the PMC bandit. For PMC bandit where network parameters are not known a priori, the decision maker needs to learn the unknown parameters and the goal is to maximize the total rewards from the covered nodes. Though PMC bandit has been studied previously, the existing model and its corresponding algorithm can be significantly improved. First, we propose the PMC-G bandit whose feedback model generalizes existing semi-bandit feedback, allowing PMC bandit to model applications like online content delivery and online dynamic channel allocation. Next, we improve the existing combinatorial upper confidence bound (CUCB) algorithm by introducing the variance-adaptive algorithm, i.e., the VA-CUCB algorithm. We prove that VA-CUCB can achieve strictly better regret bounds, which improves CUCB by a factor of $\tilde{O}(K)$, where K is the number of nodes selected in each round. Finally, experiments show our superior performance compared with benchmark algorithms on synthetic and real-world datasets.**

I. Introduction

The probabilistic maximum coverage (PMC) problem [1] is a simple yet powerful model that covers many practical network applications, such as network content delivery [2], mobile crowdsensing [3], and channel allocation [4]. Typically, the PMC problem takes a bipartite graph $G = (L, V, E)$ as input, where L are nodes to be selected, V are nodes to be covered, each edge $(u, v) \in E$ is associated with a probability $p(u, v)$, and each node $v \in V$ is associated with a weight $w(v)$. A target node $v \in V$ can be covered by a node $u \in L$ with an independent probability $p(u, v)$ and any successfully covered node v would contribute $w(v)$ reward. The decision maker's goal is to select at most K nodes from L so as to maximize the total rewards given by the covered nodes in V.

In a content delivery network (CDN) [2], for example, contents (e.g., pictures, videos) are cached across mirror servers so that end users can access the contents swiftly via the nearby server. How to strategically choose a set of servers (of size K) to improve the user experience can be modeled by PMC (Fig. 1), where L models the set of candidate mirror servers sending contents, V represents the set of users that consume contents. For each edge $(u, v) \in E$, $p(u, v)$ models the probability that the content can be successfully delivered from server u to node v in time (i.e., u covers v), and $w(v)$ is the probability that v ultimately consumes the contents. The goal of PMC is to provide the best possible user experience, i.e., to maximize the total number of users that ultimately consume the content.

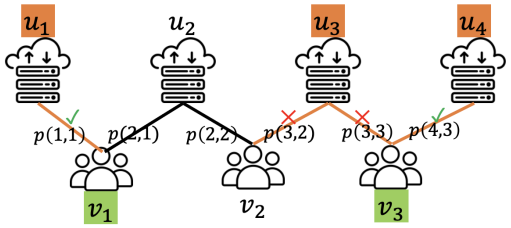


Fig. 1: An example of PMC for content delivery: the decision maker chooses (orange) servers $\{u_1, u_3, u_4\}$ which cover users $\{v_1, v_3\}$ via successful (check mark) edges $\{(1, 1), (4, 3)\}$.

For PMC, setting the correct parametric values (e.g., edge probabilities) is vital to making optimal decisions. Previous works assume that these parameters are known a priori [1], [5], [6]. However, in realistic network applications, the parameters are unknown and may even change on the fly. Take network content delivery for instance [7], users may have random demands and preference $w(v)$ to consume the contents. Due to varying geometric distances and possible network congestion, the service quality of the mirror servers regarding $p(u, v)$ is also unknown. These parameters are not known a priori and must be carefully estimated on the fly by the operator.

A. PMC Bandit with General Feedback

To relax the assumption that parameters are known, one can consider PMC in an online learning context, i.e., the PMC bandit [8]. Specifically, the PMC bandit assumes that each edge (u, v) (or node v) is associated with an arm and each arm has an unknown probability $p(u, v)$ (or $w(v)$) to be learned in T consecutive decision rounds. In each round t, the learning agent (i.e., the decision maker) needs to select a set of arms which we refer as *actions*, and the outcomes of these selected arms will be observed as feedback. Then the agent leverages the arm

[†]Xutong Liu and Jinhang Zuo are co-first authors. [*]Hong Xie is the corresponding author. The work of John C.S. Lui was supported in part by RGC's SRFS2122-4S02. The work of Hong Xie was supported by Chongqing Talents: Exceptional Young Talents Project (cstc2021ycjhbgzxm0195). The work of Jinhang Zuo and Carlee Joe-Wong was supported by the Office of Naval Research (N000142112128) and the US National Science Foundation (CNS-2103024).

feedback to estimate the unknown parameters and to improve future actions. Typically, such feedback is known as the *semi-bandit* feedback [8]–[10]. The agent's goal is to maximize T-round expected rewards, or equivalently, to minimize the expected *regret*, which is defined as the difference for the overall expected rewards between always playing the best action (i.e., the action with highest expected reward) and playing according to the agent's own policy.

For PMC bandit, a good learning algorithm must carefully handle the exploration-exploitation trade-off: whether the agent should explore arms in search of a better action, or should the agent stick to the best action observed so far to gain rewards. To deal with this trade-off, combinatorial upper confidence bound algorithms (CUCB) are proposed [8]. Specifically, CUCB uses the empirical mean as the unbiased estimator for each arm and constructs a Chernoff-type confidence interval. Such an interval serves as the exploration bonus to handle the parameter uncertainty and helps to achieve sublinear regret bounds [8].

Although PMC bandit has been extensively studied, the bandit model and its CUCB algorithm have obvious drawbacks that can be significantly improved. For the model part, semi-bandit feedback can only handle the direct feedback of the deterministic arm selection, but not the feedback that stochastically depends on the arm outcomes. In CDN, for example, semi-bandit feedback cannot model the unknown user consumption probability $w(v)$, since they stochastically depend on the outcome of the content delivery and can only be observed given the content is successfully delivered. To accommodate a broader range of applications, it is essential to consider PMC bandit with a general feedback model. As for the CUCB algorithm, it relies solely on the empirical mean and fails to be variance-adaptive. Ignoring the variance produces a loose confidence interval for each arm, which in turn causes unnecessary exploration. Worse still, a single arm's loose bound will accumulate and amplify due to the selection of K arms in each round. Therefore, one will get an additional K factor multiplied in the regret, where K can be up to hundreds or thousands depending on the applications.

B. Our Contributions

To address the aforementioned drawbacks, this paper makes four contributions as follows.

(1) Model Formulation: We propose a new PMC bandit model (i.e., the PMC-G model) that can handle the general feedback by explicitly introducing the arm observation probability. We show how our new model can cover three network applications, namely online content delivery, mobile crowdsensing, and dynamic channel allocation, which have probabilistic, semi-bandit, and cascading feedback, respectively.

(2) Algorithm Design: We propose a novel variance-adaptive combinatorial upper confidence bound algorithm (VA-CUCB). VA-CUCB leverages the empirical variance to construct a Bernstein-type confidence interval. Such an interval adaptively shrinks CUCB's Chernoff-type confidence interval when the arm has a low (empirical) variance, effectively reducing unnecessary exploration for tighter regret bounds.

(3) Theoretical Analysis: We prove that our proposed algorithm achieves $O(\sum_{i\in[m]} \frac{|V|\log^2 K\log T}{\Delta^i_{\min}})$ regret bound. It significantly improves the regret bound of CUCB by a factor of $\tilde{O}(K)$ (where $\tilde{O}$ hides logarithmic factors of K), and matches the lower bound by logarithmic factors. We overcome several technical challenges to prove the improved regret bounds for PMC-G, such as dealing with the non-deterministic observation, and bounding the variance adaptive over-estimation. Our key strategy is to use a reward sensitivity lemma to distribute the total regret to each arm's over-estimation, which are re-weighted by their variance and observation probability. For each arm's over-estimation, we use the peeling technique to handle the observation probability and carefully design a series of events to derive the final regrets. We believe our proof techniques are tight and novel, which may be of independent interest to improve other works that share the similar feedback model or over-estimation terms.

(4) Experimental Evaluation: We conduct experiments on all three applications mentioned above to validate our theoretical results over synthetic and real-world datasets. Our empirical results demonstrate that our proposed algorithms can achieve more than 30% lower regrets than benchmark algorithms.

II. System Model

The system model of PMC bandit with general feedback (or PMC-G in short) can be described by a tuple $(G, [m], \mathcal{S}, \mathcal{D}, D_{\text{obs}}, R)$ as follows: $G = (L, V, E)$ is the underlying bipartite graph, where L is the set of candidate nodes, V is the target nodes to be covered by the L, and E is the set of edges connecting L and V; $[m] = \{1, 2, ..., m\}$ is the set of base arms and each base arm is associated with an unknown parameter to be learned. Depending on different application scenarios in Section V, the base arms for PMC-G could refer to the edge set E, or the edge and target node sets $E \cup V$, therefore we use $[m]$ to cover both cases; $\mathcal{S}$ is the set of eligible actions and $S \in \mathcal{S}$ is action. Similar to $[m]$, $\mathcal{S}$ is application-dependent and can be either a collection of subsets of $[m]$, or subsets of L; $\mathcal{D}$ is the set of possible *Bernoulli* distributions over the outcomes of base arms with support $\{0, 1\}^m$; D_{obs} is the observation function to model the general feedback and R is the reward function, the definitions of which will be introduced shortly.

In PMC-G, the learning agent interacts with the unknown environment in a sequential manner as follows. First, the environment chooses a distribution $D \in \mathcal{D}$ unknown to the agent. Then, at round $t = 1, 2, ..., T$, the agent selects an action $S_t \in \mathcal{S}$ and the environment draws from the unknown distribution D a Bernoulli outcome $\boldsymbol{X}_t = (X_{t,1}, ...X_{t,m}) \in \{0, 1\}^m$. Intuitively, for $e = (u, v) \in E$, $X_{t,e} = 1$ means the target node $v \in V$ is covered when $u \in L$ is selected and for $v \in V$, $X_{t,v} = 1$ means the target node yields one unit of reward when v is covered. Similar to [8], we assume that the outcome $X_{t,e}$ on edge $e \in E$ is independent with any other outcomes $X_{t,i}$, $i \in [m], i \neq e$, yet the outcomes X_v and $X_{v'}$ of nodes $v', v \in V$ could be dependent. We remark that the independence assumption is only for the convenience of deriving the expected

reward $r(S;\boldsymbol{\mu})$ as in Equation (2), which can be relaxed when the closed form of $r(S;\boldsymbol{\mu})$ is available.

When the action S_t is played, the agent will receive a non-negative reward $R(S_t, \boldsymbol{X}_t)$. For PMC-G, the reward at round t is the total rewards received from the covered nodes,

$$R(S_t, \boldsymbol{X}_t) = \sum_{v\in V} \mathbb{I}\{\exists\, u \in S_t \text{ s.t. } X_{t,(u,v)} = 1\} X_{t,v}. \quad (1)$$

Let $\boldsymbol{\mu} = (\mu_1, ..., \mu_m)$ denote the mean vector of base arms' outcomes, which are unknown initially. Given the independence assumption, the expected reward $r(S;\boldsymbol{\mu}) \triangleq \mathbb{E}[R(S, \boldsymbol{X}_t)]$ is

$$r(S;\boldsymbol{\mu}) = \sum_{v\in V} \mu_v \left(1 - \prod_{u\in S}(1-\mu_{(u,v)})\right). \quad (2)$$

Note that this expected reward function is highly non-linear in $\boldsymbol{\mu}$ and finding the optimal solution S^* is NP-hard in general [1]. Fortunately, using submodular set function maximization technique, one can achieve $(1-1/e)$-approximate solutions [1].

At the end of round t, the agent can observe some of the arm outcomes as feedback, which are critical to improve future decisions. In particular, base arms in a random set $\tau_t \sim D_{\text{obs}}(S_t, \boldsymbol{X}_t)$ are observed, meaning that the outcomes of arms in τ_t, i.e. $(X_t)_{t\in\tau_t}$ are revealed as the feedback to the agent, where function D_{trig} is used to model general feedback and is referred as the *general feedback function*. For notational convenience, we define *observation probability* $p_i^{D,D_{\text{obs}},S}$ as the probability that base arm i is observed when the action is S, the outcome distribution is D, and the feedback function is D_{obs}. Since D_{obs} is always fixed in a given application context, we ignore it in the notation for simplicity, and use $p_i^{D,S}$ henceforth. We remark that the PMC-G model significantly enhances the modeling power of previous PMC bandit [8] as it not only models semi-bandit feedback that are deterministic but can also model the probabilistic feedback when τ_t are randomly determined, or even the partial feedback that depends on certain stopping criteria, which will be discuss in details in Section V.

The goal of PMC-G is to accumulate as much reward as possible over T rounds, by learning the Bernoulli distribution D, or equivalently the unknown mean vector $\boldsymbol{\mu}$. The performance of an online learning algorithm A is measured by its *regret*, defined as the difference of the expected cumulative reward between always playing the best action $S^* \triangleq \text{argmax}_{S\in\mathcal{S}}\, r(S;\boldsymbol{\mu})$ and playing actions chosen by algorithm A. As mentioned before, it could be NP-hard to compute the exact S^* even when $\boldsymbol{\mu}$ is known, so similar to [8], [9], [11], we assume that the algorithm A has access to an offline (α,β)-approximation oracle, which for mean vector $\boldsymbol{\mu}$ outputs an action S such that $\Pr[r(S;\boldsymbol{\mu}) \geq \alpha \cdot r(S^*;\boldsymbol{\mu})] \geq \beta$. Formally, the T-round (α,β)-approximate regret is defined as

$$Reg(T;\alpha,\beta,\boldsymbol{\mu}) = T\cdot\alpha\beta\cdot r(S^*;\boldsymbol{\mu}) - \mathbb{E}\left[\sum_{t=1}^{T} r(S_t;\boldsymbol{\mu})\right], \quad (3)$$

where the expectation is taken over the randomness of outcomes $\boldsymbol{X}_1, ..., \boldsymbol{X}_T$, the observation sets $\tau_1, ..., \tau_T$, as well as the randomness of algorithm A itself.

Algorithm 1 VACUCB: Variance Adaptive Combinatorial Upper Confidence Bound Algorithm for PMC-G

Input: Base arms $[m]$, computation oracle ORACLE.
Initialize: For each arm i, $T_{0,i} \leftarrow 0, \hat{\mu}_{0,i} = 0, \hat{V}_{0,i} = 0$.
for $t = 1, ..., T$ **do**
  For arm i, compute $\rho_{t,i}$ according to Eq. (4) and set UCB value $\bar{\mu}_{t,i} = \min\{\hat{\mu}_{t-1,i} + \rho_{t,i}, 1\}$.
  Select action $S_t = \text{ORACLE}(\bar{\mu}_{t,1}, ..., \bar{\mu}_{t,m})$.
  The agent plays S_t and observe arms $\tau_t \subseteq [m]$ with outcome $X_{t,i}$'s, for $i \in \tau_t$.
  For every $i \in \tau_t$, update $T_{t,i} = T_{t-1,i} + 1$, $\hat{\mu}_{t,i} = \hat{\mu}_{t-1,i} + (X_{t,i} - \hat{\mu}_{t-1,i})/T_{t,i}$, $\hat{V}_{t,i} = \frac{T_{t-1,i}}{T_{t,i}}\left(\hat{V}_{t-1,i} + \frac{1}{T_{t,i}}(\hat{\mu}_{t-1,i} - X_{t,i})^2\right)$.
end for

III. Algorithm Design

In this section, we provide the Variance-Adaptive Combinatorial Upper Confidence Bound algorithm for PMC-G problem in Algorithm 1, or VACUCB algorithm for short. Algorithm 1 maintains the empirical estimate $\hat{\mu}_{t,i}$ and $\hat{V}_{t,i}$ for the true mean and the true variance of the base arm outcomes, respectively. As discussed earlier, we follow the principle of Optimism in the Face of Uncertainty (OFU), and compute the upper confidence bound (UCB) value $\bar{\mu}_i = \hat{\mu}_{t,i} + \rho_{t,i}$ as an optimistic estimate of $\boldsymbol{\mu}_i$. Intuitively, confidence interval $\rho_{t,i}$ serves as a bonus term to explore the unknown mean μ_i: when arm i is not observed often (i.e., $T_{t,i}$ is small), $\rho_{t,i}$ will be large and encourages the algorithm to select arm i.

Compared with the CUCB algorithm [8] which uses confidence interval $\rho_{t,i} = \sqrt{\frac{3\log t}{2T_{t-1,i}}}$ based on Chernoff-type concentration bound [12] for the PMC problem, the key difference is that we leverage on the stronger Bernstein-type concentration bound and use empirical variance $\hat{V}_{t-1,i}$ to construct the following "variance-adaptive" confidence interval:

$$\rho_{t,i} = \sqrt{\frac{6\hat{V}_{t-1,i}\log t}{T_{t-1,i}}} + \frac{9\log t}{T_{t-1,i}} \quad (4)$$

As we will show in Section IV, the variance-adaptive interval is the key to achieve tighter regret bounds, in that the expected reward in Eq. (2) is more sensitive to arms whose mean is close to 0 or 1, whose over-estimation ($\rho_{t,i}$) will cause a large regret. However, these arms happen to have smaller variance, which means that being variance-adaptive helps to reduce the over-estimation and results in significantly smaller regrets.

To select the action S_t, the next step is to insert UCB values into the computational oracle, which is typically an $(1-1/e, 1)$-approximation greedy oracle [1], [13] for PMC-G applications with the monotone submodular reward Eq. (2). After play action S_t, the agent will observe a set of arms τ_t as feedback and update the statistics accordingly.

IV. Theoretical Analysis

In this section, we present our main theoretical result, its analysis, and some discussions.

A. Main Result

To state the regret bound, we first give some definitions followed by our main result.

Definition 1 (Suboptimality Gap)**.** *Fix a distribution $D \in \mathcal{D}$ and its mean vector $\boldsymbol{\mu}$, for each action $S \in \mathcal{S}$, we define the (approximation) gap as $\Delta_S = \max\{0, \alpha r(S^*; \boldsymbol{\mu}) - r(S; \boldsymbol{\mu})\}$. For each arm i, we define $\Delta_i^{\min} = \inf_{S \in \mathcal{S}: p_i^{D,S} > 0,\ \Delta_S > 0} \Delta_S$, $\Delta_i^{\max} = \sup_{S \in \mathcal{S}: p_i^{D,S} > 0, \Delta_S > 0} \Delta_S$. As a convention, if there is no action $S \in \mathcal{S}$ such that $p_i^{D,S} > 0$ and $\Delta_S > 0$, then $\Delta_i^{\min} = +\infty, \Delta_i^{\max} = 0$. We define $\Delta_{\min} = \min_{i \in [m]} \Delta_i^{\min}$ and $\Delta_{\max} = \max_{i \in [m]} \Delta_i^{\min}$.*

Theorem 1. *For a PMC-G problem instance $(G, [m], \mathcal{S}, \mathcal{D}, D_{obs}, R)$, the regret of VACUCB (Algorithm 1) is upper bounded by $O\left(\left(\sum_{i \in [m]} \frac{|V|}{\Delta_i^{\min}} + \log^2\left(\frac{K}{\Delta_i^{\min}}\right)\right) \log^2 K \log T\right)$.*

B. Analysis

Our analysis uses several events to filter the total regret and then bound these event-filtered regrets accordingly. Below we give the definition of the event-filtered regret.

Definition 2 (Event-filtered regret)**.** *For any series of events $(\mathcal{E}_t)_{t \geq 0}$ indexed by round number t, we define the $Reg^A_{\alpha,\boldsymbol{\mu}}(T, (\mathcal{E}_t)_{t \geq 0})$ as the regret filtered by events $(\mathcal{E}_t)_{t \geq 0}$, or the regret is only counted in t if $\mathcal{E}$ happens in t. Formally, $Reg^A_{\alpha,\boldsymbol{\mu}}(T, (\mathcal{E}_t)_{t \geq 0}) = \mathbb{E}\left[\sum_{t \in [T]} \mathbb{I}(\mathcal{E}_t)(\alpha \cdot r(S^*; \boldsymbol{\mu}) - r(S_t; \boldsymbol{\mu}))\right]$. For simplicity, we will omit $A, \alpha, \boldsymbol{\mu}, T$ and rewrite $Reg^A_{\alpha,\boldsymbol{\mu}}(T, (\mathcal{E}_t)_{t \geq 0})$ as $Reg(T, \mathcal{E}_t)$ when contexts are clear.*

To give the concrete events that filters the leading regret, we leverage on the following two lemmas. Intuitively, the first lemma bound the reward difference by ℓ_2 and ℓ_1 norm of each arm's over-estimation $\boldsymbol{x}_v$ and $\boldsymbol{x}_1$ given by our VA-CUCB algorithm, and $\sqrt{V}$ is to bound the non-linearity of $r(S; \boldsymbol{\mu})$. Notice that both $x_{v,i}$ and $x_{1,i}$ are re-weighted by $p_i^{D,S}$ which reduces the regret contribution from unlikely observed arms to handle the general feedback model. The second lemma bounds each arm's actual over-estimation, and the $\sqrt{(1-\mu_i)\mu_i}$ is the key term to cancel the denominator in $x_{v,i}$ to give improved regret bounds.

Lemma 1 (Reward sensitivity)**.** *For PMC-G with semi-bandit, probabilistic, and cascading feedback model, and for any parameter change $\boldsymbol{\zeta}, \boldsymbol{\eta} \in [0,1]^m$ s.t. $\boldsymbol{\mu}' = \boldsymbol{\mu} + \boldsymbol{\zeta} + \boldsymbol{\eta}$, the reward sensitivity $r(S; \boldsymbol{\mu}') - r(S; \boldsymbol{\mu})$ satisfies*

$$r(S; \boldsymbol{\mu}') - r(S; \boldsymbol{\mu}) \leq \sqrt{|V|}\, \|\boldsymbol{x}_v\|_2 + \|\boldsymbol{x}_1\|_1\,, \tag{5}$$

where $\boldsymbol{x}_v \triangleq \left(\frac{p_i^{D,S}\zeta_i}{\sqrt{(1-\mu_i)\mu_i}}\right)_{i \in [m]}$, $\boldsymbol{x}_1 \triangleq \left(p_i^{D,S}\eta_i\right)_{i \in [m]}$.

Proof. See Section VIII-B for details. ■

Lemma 2 (Arm-level over-estimation)**.** *For every base arm $i \in [m]$ and every time $t \in [T]$, it holds with probability at least $1 - 4mt^{-3}$ that $\mu_i \leq \bar{\mu}_{t,i} \leq \min\left\{\mu_i + 4\sqrt{3}\sqrt{\frac{\mu_i(1-\mu_i)\log t}{T_{t-1,i}}} + \frac{28 \log t}{T_{t-1,i}}, 1\right\}$.*

Proof. See Section VIII-C for details. ■

Let $\tilde{S}_t = \{i \in [m] : p_i^{D,S_t} > 0\}$ be the set of arms that could be observed in round t. Now We have the following lemma for the regret decomposition.

Lemma 3 (Regret decomposition)**.** *We define two error terms*

$$e_{t,1}(S_t) = 4\sqrt{3}\sqrt{|V|}\sqrt{\sum_{i \in \tilde{S}_t} \left(\frac{\log t}{T_{t-1,i}} \wedge \frac{1}{28}\right)(p_i^{D,S_t})^2} \tag{6}$$

$$e_{t,2}(S_t) = 28 \sum_{i \in \tilde{S}_t} \left(\frac{\log t}{T_{t-1,i}} \wedge \frac{1}{28}\right)(p_i^{D,S_t}) \tag{7}$$

and two events $E_{t,1} = \{\Delta_{S_t} \leq 2e_{t,1}(S_t)\}$, $E_{t,2} = \{\Delta_{S_t} \leq 2e_{t,2}(S_t)\}$. The regret of Algorithm 1, when used with (α, β) approximation oracle is bounded by

$$Reg(T) \leq Reg(T, E_{t,1}) + Reg(T, E_{t,2}) + \frac{2\pi^2}{3} m \Delta_{\max}. \tag{8}$$

Our final step are bounding $Reg(T, E_{t,1})$ and $Reg(T, E_{t,2})$, which corresponds to the first term and the second term in Theorem 1, respectively. Our idea main is to define a cascade of infinitely-many mutually-exclusive events as [10], [14]. Then we transform event $E_{t,1}, E_{t,2}$ to these events and bound the number of times that these events could happen. To handle the general feedback for each arm i, our key ingredient is to use the peeling technique to divide observation probability $p_i^{D,S}$ into geometrically separated bins $(1/2, 1], (1/4, 1/2]..., (2^{-j}, 2^{-j+1}), ...,$, so that we can use delicate analysis to avoid the exponential factors caused by considering the combination of all possible actions $S \in \mathcal{S}$ that can observe i. We defer the detailed proofs of $Reg(T, E_{t,1})$ and $Reg(T, E_{t,2})$ in Section VIII-E and Section VIII-F.

C. Discussions

Looking at the above regret bound, the leading term is $O(\sum_{i=1}^m \frac{|V| \log^2 K \log T}{\Delta_i^{\min}})$ when gaps are not too large, i.e., $\Delta_{\min}^i \leq |V|^{1-\epsilon}/\log^2 K$, for any $\epsilon > 0$. The dependence over K is $O(\log^2 K)$. For PMC bandit with general feedback, [11] is the closest work to ours, and following their CUCB algorithm can only give $O(\sum_{i=1}^m \frac{|V| K \log T}{\Delta_i^{\min}})$ for PMC-G. Our result is strictly better than theirs by a factor of $O(K/\log^2 K)$. For the classical PMC bandit with semi-bandit feedback, [15] recently gives a regret lower bound $\Omega(\frac{L|V|^2}{\Delta_{\min}})$, which means our regret bound is near-optimal (by setting $m = L|V|, \Delta_{\min} \leq \Delta_i^{\min}$) and matches the lower bound up to $O(\log^2 K)$.

V. Applications for PMC-G

We consider three applications with semi-bandit, probabilistic, and cascading feedback to illustrate the utility of our PMC-G framework: mobile crowdsensing, online content delivery, and dynamic wireless channel allocation. We compare the regret of our VACUCB algorithm to two baselines: CUCB [9],

TABLE I: Summary of feedback and oracles for different applications, where they all achieve $O(\sum_{i\in[m]} \frac{|V|\log^2 K\log)T}{\Delta_i^{\min}})$ regret and improve CUCB [11] by $O(K/\log^2 K)$.

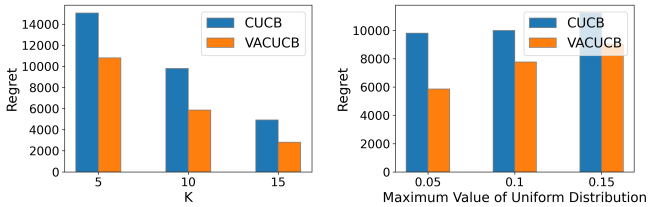
Application	Feedback	(α,β)-Oracle
Mobile Crowdsensing	Semi-bandit	Greedy, $(1-1/e,1)$
Online Content Delivery	Probabilistic	Greedy, $(1-1/e,1)$
Dynamic Channel Allocation	Cascading	Greedy, $(1,1)$

a state-of-the-art combinatorial bandit algorithm that does not use variance-adaptive confidence intervals; and ϵ-greedy, which explores new actions with fixed probability ϵ and otherwise greedily chooses the empirically optimal action.

A. Mobile Crowdsensing

1) Problem Description: Today's mobile devices (e.g., smartphones, tablets, wearable devices) are often equipped with powerful sensor devices (e.g., GPS, accelerometers and gravity sensors), which can collect and analyze environmental data from users' locations. To collectively utilize these mobile devices, mobile crowdsensing provides a principled way of carrying out a large sensing project, by recruiting a group of individuals to cover (i.e., sense data at) different locations using their own mobile devices, as they move through an area [3]. For example, a task organizer may want to organize a group of participants, and use their cameras, gravity sensors, and GPS as sensors to monitor dust levels [16] or a possible earthquake [17] in a large city. Due to the different movement trajectories of the crowdsensing participants and the varying manufacturing quality of their mobile devices, the quality of the collected data can vary randomly across different participants for different locations [18]. The goal of the mobile crowdsensing task organizer is to select a group of individuals to maximize the amount of high-quality data collected from different locations in the city.

The mobile crowdsensing application can be modeled by our PMC-G problem. Consider a bipartite graph $G(L,V,E)$, where L is the set of candidate participants, V is the set of locations in a city, and E models the data collection process. At each time t, the agent (or the task organizer) wants to choose at most K participants to conduct the sensing task. For example, K may be chosen based on a budget for paying fixed recruitment incentives to each chosen participant. Each selected participant $u \in S_t$ independently uploads their sensor data at location $v \in V$, which is modeled as a Bernoulli random variable $X_{t,(u,v)} \in \{0,1\}$ with probability $\mu_{u,v}$ that the data can be used as valid information to cover location v. In this case, we know the arms are exactly E. The agent can get *semi-bandit feedback*, i.e., observe whether the uploaded data is valid or not for (u,v) s.t. $u \in S_t$. Using the PMC-G formulation, the observation probability $p^{D,S_t}_{(u,v)} = 1$ if $u \in S_t$ or 0 otherwise. The reward is the weighted total number of locations that is covered with valid information: $r(S;\boldsymbol{\mu}) = \sum_{v\in V} \mu_v \left(1 - \prod_{u\in S}(1-\mu_{(u,v)})\right)$, where the known weight μ_v represents the importance of covering location v to the crowdsensing task. Busy areas, for example, may have higher sensing importance as their environmental conditions affect more people.



(a) Varying K (b) Varying x ($\mu_{u,v} \sim U(0,x)$)

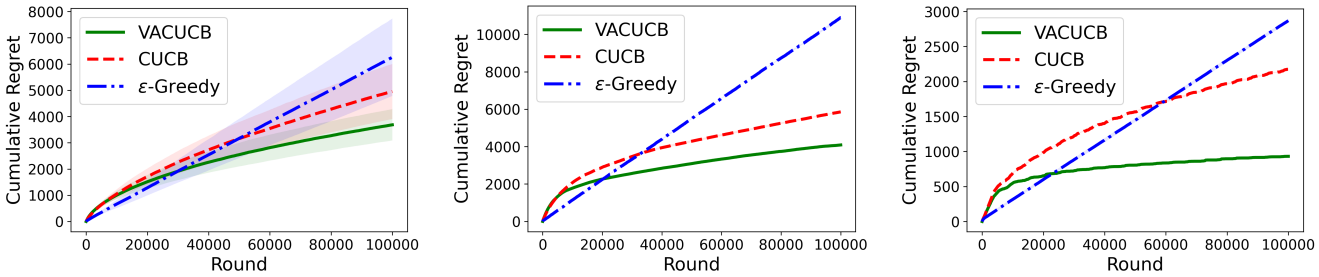
Fig. 2: Total regrets after 100000 rounds in different settings.

2) Performance Evaluation: We simulate the mobile crowdsensing problem using a complete bipartite graph with 20 candidate nodes (participants) and 30 target nodes (locations). The importance weights of locations are sampled from the uniform distribution $U(0,0.5)$ and are known by the task organizer. We first take $K = 15$ (the number of chosen participants) and generate each $\mu_{u,v}$ using the uniform distribution $U(0,0.15)$. Fig. 3a shows the cumulative regrets of different algorithms for $100,000$ rounds (we choose $\epsilon = 0.2$ for the ϵ-Greedy algorithm in all experiments). VACUCB achieves 30% and 42% less regret than the CUCB and ϵ-Greedy algorithms. To verify how K would affect the regrets, we then generate each $\mu_{u,v}$ with $U(0,0.05)$ and show the total regrets for different K after $100,000$ rounds in Fig. 2a. Note that with the change of K, the optimal reward will also change, which explains why the regret of a small K is larger than that of a large K. We find that with the increase of K, VACUCB's improvement over the CUCB baseline also increases (25% for $K = 5$ and 50% for $K = 15$), which is consistent with our theoretical result in Theorem 1. Fig. 2b compares the total regrets of CUCB and VACUCB when varying the value of $\mu_{u,v}$. We set $K = 10$ and generate each $\mu_{u,v}$ with $U(0,x)$, where $x \in \{0.05, 0.1, 0.15\}$. With the increase of x, VACUCB's improvement over the CUCB baseline decreases; one potential reason is that the variance of the Bernoulli variable $X_{t,(u,v)}$, $V_{t,(u,v)} = \mu_{u,v}(1-\mu_{u,v})$, is small when $\mu_{u,v}$ is small, which helps our variance-adaptive algorithm to control the exploration.

B. Online Content Delivery

1) Problem Description: We study the online content delivery problem in content delivery networks (CDNs), which widely appears in web services such as video streaming, web loading, and software downloading [2], [7]. In contrast to the traditional method, which stores contents on just one central server, CDNs replicate and cache contents on multiple mirror servers so that the end users can access the data that are physically closest to them. This way, users can enjoy faster and more reliable delivery services. Our model and algorithm aim to help the content owners (e.g., media companies or e-commerce vendors) to select a set of mirror servers to provide the best possible experience for their end users.

The above application scenario naturally fits into our PMC-G problem with a bipartite graph $G(L,V,E)$, where L models the set of candidate servers, V are the end users, and E models



(a) Mobile Crowdsensing (b) Online Content Delivery (c) Dynamic Channel Allocation

Fig. 3: Cumulative regrets in different applications. We show the average regrets with standard deviations over 20 experiments.

the user-server interactions as follows. At each time slot t, the agent (or the content owner) needs to choose $S_t \subseteq L$ mirror servers that can send contents to users via the CDN network. We assume the number of selected servers at each round is less than K, i.e., $|S_t| \leq K$, since the maintenance of each server usually incurs certain costs, and the content owner has limited budget. The selected servers $u \in S_t$ then independently send contents for each user $v \in V$ with unknown success rates $\mu_{(u,v)}$, depending on varying geometric distances and the network congestion [19]. By "success," we mean the content is delivered in time, which can be modeled by a Bernoulli random variable $X_{t,u,v} \in \{0,1\}$ with mean $\mu_{(u,v)}$. We suppose that each user v attempts to preload content from the selected servers to its device [20], and we use a Bernoulli random variable with unknown mean μ_v to represent whether this preloaded content is ultimately consumed (e.g., video is viewed) by the user. To this end, we can see that arms correspond to the success probability $\mu_{(u,v)}$ for $(u,v) \in E$ and the consuming probability μ_v for users $v \in V$. The question is how to select K mirror servers for content delivery to maximize the total number of users that consume the contents with unknown success rates and consuming probabilities. A good server selection policy should prioritize successful delivery to users more likely to consume the content.

As for the feedback, the agent can observe whether the contents are successfully delivered from the selected servers, i.e., $X_{t,(u,v)}$ for $u \in S_t, v \in V$. We know that the observation probability $p^{D,S_t}_{(u,v)}$ equals to 1 if $u \in S_t$ and 0 otherwise, which is known as the semi-bandit feedback. If user v successfully receives the content, the agent (i.e., content owner or CDN provider) can observe whether the user consumes the content, i.e., $X_{t,v}$ is observed when $\exists v$ s.t. $X_{u,v} = 1$. Such feedback is called the probabilistic feedback since it depends on other random outcomes, and the observation probability $p^{D,S_t}_v = 1 - \prod_{u \in S_t}(1-\mu_{u,v})$. The expected reward is essentially Eq. (2) and the agent's goal is to minimize the total regrets in Eq. (3).

2) Performance Evaluation: For the online content delivery experiments, we consider 10 mirror servers located at some of the point-of-presence (POP) locations of Microsoft Azure CDN in North America[1]. We assume the users are distributed in 20 POP locations (including the servers' locations). We extract the average latency data between these locations[2], and assume the realized latency at each round is the average latency plus a random delay ranging from 0ms to 30ms, which is 76% of the average observed delay. We simulate the random delivery deadlines of the contents with the range from 10ms to 20ms. The users will successfully receive the content if their latencies to the mirror servers are less than the delivery deadline. The probability that user v will consume the content, μ_v, is sampled from $U(0, 0.5)$ and is unknown to the server selector. Figure 3b shows the cumulative regrets of different algorithms for $100,000$ rounds. VACUCB achieves 32% and 65% less regret than CUCB and ϵ-Greedy.

C. Dynamic Channel Allocation

1) Problem Description: We consider a centralized dynamic channel allocation problem where a central controller chooses K channels from the candidate channel set L and allocates them to a group of users V. Each channel $i \in L$ can be viewed as a base arm with unknown Bernoulli availability. Similar to the centralized online channel allocation setting in [4], we let the controller allocate disjoint lists of channels to users to avoid collisions, and each user will get a reward only if at least one of the allocated channel is available in a given round. The overall expected reward of all users is then $\sum_{j \in V}\left(1 - \prod_{i \in S_{j,t}}(1-\mu_i)\right)$, where $S_{j,t}$ is the set of channels allocated to user j in round t and μ_i is the expected availability of channel i. Different from the NP-hard offline problem in [4], the offline optimization problem with such a reward function can be exactly solved by a greedy algorithm that sequentially allocates channels with the maximum marginal returns to the users. Each user j will have an ordered list of allocated channels, $o^t_j = (o^t_{j1}, o^t_{j2}, \cdots)$ with length $|S_{j,t}|$. We consider cascading feedback in this application, as each user will sequentially check the availability of allocated channels and stop when finding the first available one to send data. More specific, only the outcomes of o^t_{jl} for all $l \leq L_t$ are observed, where L_t is the index of the first available channel in the list ($L_t = |S_{j,t}|$ if all channels in the list are unavailable).

2) Performance Evaluation: As in [4], we use a real wireless data trace [21] that contains the availability of 16 channels. We choose the most competitive 4 channels among them with

[1]https://docs.microsoft.com/en-us/azure/cdn/cdn-pop-locations

[2]https://wondernetwork.com/pings

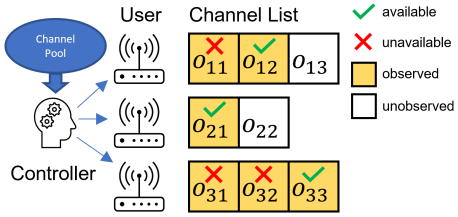


Fig. 4: Illustration of centralized dynamic channel allocation.

average available probabilities μ_i less than 0.1, and consider 4 copies of each to build the candidate channel set with $|L| = 16$. Notice that the real availability trace is no longer i.i.d. Bernoulli distributed, so it is more challenging than the ideal setting. Also, there is no real optimal online policy, so we adjust our regret to compare with the optimal policy when assuming the channel availability is uniformly sampled from the whole data trace, i.e., the expected availability is equal to the average availability. We consider a central controller that chooses $K = 8$ channels from the 16 candidate channels and allocate them to $|V| = 4$ users. Figure 3c shows the cumulative regrets of different algorithms for $100,000$ rounds. VACUCB achieves 57% and 68% less regret than the CUCB and ϵ-Greedy baseline algorithms.

VI. Related Work

There have been vast literature focusing on online learning problems under the multi-armed bandit (MAB) model, which is first studied by [22] and then extended by many other works (cf. [23]–[25]). The principle of Optimism in the Face of Uncertainty (OFU) [26] is one of the most fundamental concepts in MAB, and has been widely used in MAB algorithms [25]. While most algorithms rely on Hoeffding-type concentration bounds to build upper confidence bound (UCB) of an arm, a few works [27]–[29], including ours, apply Bernstein-type bounds and successfully show superior performance, both in theory and in experiments.

Probabilistic maximum coverage (PMC) problem [1] is a widely studied topic with many applications in computer science, and especially in the area of network optimization. Besides the three applications mentioned in this paper, PMC also covers many other applications, including wireless sensor placement [30] and social network advertising [31], [32]. The online learning version of the PMC problem (or PMC bandit) is first proposed by [8], and are then followed by [9], [28]. Different from these works that only considers the semi-bandit feedback, we proposes a new PMC-G model that generalizes the semi-bandit feedback and can model broader applications with the general probabilistic feedback and the cascading feedback.

The stochastic Combinatorial MAB (CMAB) has received much attention recently [8], [9], [11], [15], [28], [33], [34], and PMC bandit fits into CMAB framework. For CMAB with semi-bandit feedback, [33] is the first study on stochastic CMAB, and its regret bound has been improved by [10], [35]. Later on, [9], [11] considers probabilistic feedback to generalize the semi-bandit feedback model and . However, all above CMAB frameworks suffers an additional $O(K)$ factor in their regret bound and the best of them only achieve $O(\sum_{i\in[m]} \frac{K \log T}{\Delta^i_{\min}})$, since they use combinatorial upper confidence bound (CUCB) algorithms that ignores the variance of the arm.

Recently, [28] focuses on the PMC bandit and proposes the BC-UCB algorithm with the Gini-smoothness condition to achieve a similar improvement as our work. But their work only works for the semi-bandit feedback and is essentially a special case of our PMC-G. As a result, their technique is different and much simpler than ours.

VII. Conclusion

In this paper, we propose the first PMC bandit with the general feedback model that accommodates a broader network applications. We provide the variance-adaptive online learning algorithm, and conduct rigorous analysis to achieve strictly better regrets. To validate our theoretical results, we conduct experiments for mobile crowdsensing, content delivery and channel allocation applications, showing superior performance compared with benchmarks algorithms. For future directions, it will be interesting to reduce the $O(\log^2 K)$ dependency via finer analysis and to explore the variance-adaptive algorithms for applications beyond PMC problems.

VIII. Appendix

A. Facts and Definitions

We use the following tail bounds for our analysis.

Lemma 4 (Empirical Bernstein Inequality [27]). *Let $(X_i)_{i\in[n]}$ be n i.i.d random variables with bounded support $[0,1]$ and mean $\mathbb{E}[X_i] = \mu$. Let $\hat{X}_n$ and $\hat{V}_n$ be the empirical mean and empirical variance of $(X_i)_{i\in[n]}$. Then for any $n \in \mathbb{N}$ and $y > 0$, it holds that $\Pr\left[|\hat{X}_n - \mu| \geq \sqrt{\frac{2\hat{V}_n y}{n}} + \frac{3y}{n}\right] \leq 3e^{-y}$.*

Lemma 5 (Bernstein Inequality [12]). *Let $(X_i)_{i\in[n]}$ be n independent random variables in $[0,1]$ with mean $\mathbb{E}[X_i] = \mu$ and variance $Var[X_i] = V$. Then with probability $1-\delta$: $\frac{1}{n}\sum_{i\in[n]} X_i \leq \mu + \frac{2\log 1/\delta}{3n} + \sqrt{\frac{2V \log 1/\delta}{n}}$.*

We define the following events for arm-level concentration.

Definition 3. *We say that the sampling is nice at the beginning of round t if: (1) for every base arm $i \in [m]$, $|\hat{\mu}_{t-1,i} - \mu_i| \leq \rho_{t,i}$, where $\rho_{t,i} = \sqrt{\frac{6\hat{V}_{t-1,i}\log t}{T_{t-1,i}}} + \frac{9\log t}{T_{t-1,i}}$; (2) for every base arm $i \in [m]$, $\hat{V}_{t-1,i} \leq 2\mu_i(1-\mu_i) + \frac{3.5\log t}{T_{t-1,i}}$. We denote such event as $\mathcal{N}^s_t$.*

The following lemma bounds the probability that $\neg\mathcal{N}^s_t$.

Lemma 6. *For each round t, $\Pr[\neg\mathcal{N}^s_t] \leq 4mt^{-2}$.*

Proof. Let $\mathcal{N}^{s,1}_t, \mathcal{N}^{s,2}_t$ be the event (1) and event (2) in Definition 3. For $\Pr[\neg\mathcal{N}^{s,1}_t]$, we can bound it using Lemma 4 by setting $y = 3\log t$ We then bound the probability that second event $\neg\mathcal{N}^{s,2}_t$ using the similar proof of Eq. (7) in [28]. Fix $T_{t-1,i} = \tau$ and consider $(Y^1_i, ..., Y^\tau_i)$, where $Y^k_i = (X^k_i - \mu_i)^2 \in [0,1]$ and X^k_i is the random outcome of the k-th i.i.d trial. In this case, one can verify that $\hat{V}_{t-1,i} \leq \frac{1}{\tau}\sum_{k=1}^{\tau} Y^k_i$;

$\mathbb{E}[Y_i^k] \le (1-\mu_i)\mu_i$; and $\mathrm{Var}[Y_i] \le \mathbb{E}[Y_i^k] \le (1-\mu_i)\mu_i$. By Lemma 5, it holds with probability at least $1-t^{-3}$ that $\hat{V}_{t-1,i} \le \mu_i(1-\mu_i) + \frac{2\log t}{\tau} + \sqrt{\frac{6(1-\mu_i)\mu_i \log t}{\tau}} \le \mu_i(1-\mu_i) + \frac{2\log t}{\tau} + \mu_i(1-\mu_i) + \frac{3\log t}{2\tau} = 2\mu_i(1-\mu_i) + \frac{3.5\log t}{\tau}$ Now by applying union bound over $i \in [m]$ and $\tau \in [t]$, $\mathcal{N}_t^{s,1}$ and $\mathcal{N}_t^{s,2}$, we have $\Pr[\neg\mathcal{N}_t^s] \le 4mt^{-2}$. ■

To deal with the general feedback, we use the following definitions and lemmas to peel the observation probability.

Definition 4 (Observation Probability (OP) group)**.** *For any arm i and index j, define the observation probability (OP) group (of actions) as $\mathcal{S}_{i,j}^D = \{S \in \mathcal{S} : 2^{-j} < p_i^{D,S} \le 2^{-j+1}\}$. Notice $\{\mathcal{S}_{i,j}^D\}$ forms a partition of $\{S \in \mathcal{S} : p_i^{D,S}\}$.*

Definition 5 (Counter)**.** *For each OP group $S_{i,j}$, we define a counter $N_{i,j}$ which is initialized to 0. In each round t, we have the following recursive equation to define $N_{t,i,j}$ as follows: $N_{t,i,j} = 0$, if $t = 0$; $N_{t,i,j} = N_{t-1,i,j} + 1$, if $t > 0$ and $S_t \in S_{i,j}^D$; $N_{t,i,j} = N_{t-1,i,j}$, otherwise.*

Definition 6 (Nice observation event $\mathcal{N}_t^t$)**.** *Given a series integers $\{j_i^{\max}\}_{i\in[m]}$, we say that the observation is nice at the beginning of round t, if for every observation group identified by (i,j), as long as $\frac{6\ln t}{\frac{1}{3}N_{t-1,i,j}2^{-j}} \le 1$, there is $T_{t-1,i} \ge \frac{1}{3}N_{t-1,i,j}\cdot 2^{-j}$. We denote this event as $\mathcal{N}_t^t$.*

Lemma 7 (Appendix B.1, Lemma 4, [11])**.** *For a series of integers $(j_i^{\max})_{i\in[m]}$, we have $\Pr[\neg\mathcal{N}_t^t] \le \sum_{i\in[m]} j_i^{\max} t^{-2}$ for every round $t \in [T]$.*

Proof. We refer the readers to Lemma 4 in Appendix B.1 from [11] for detailed proofs. ■

B. Proof of Lemma 1

For cascading feedback, without loss of generality, let the action in group $i \in V$ be $\{\mu_{i,1}, ..., \mu_{i,K}\}$, then the reward function is $r(S;\boldsymbol{\mu}) = \sum_{j\in V} 1 - \prod_{i=1}^K (1-\mu_{i,j})$ and the observation probability is $p_{i,j}^{D,S} = \prod_{\ell=1}^{i-1}(1-\mu_{\ell,j})$. Let $\bar{\boldsymbol{\mu}} = (\bar{\mu}_{i,j})_{i\in[K],j\in V}$ and $\boldsymbol{\mu} = (\mu_{i,j})_{i\in[K],j\in V}$, where $\bar{\boldsymbol{\mu}} = \boldsymbol{\mu} + \boldsymbol{\zeta} + \boldsymbol{\eta}$ with $\bar{\boldsymbol{\mu}}, \boldsymbol{\mu} \in (0,1)^{[K]\times V}, \boldsymbol{\zeta}, \boldsymbol{\eta} \in [0,1]^{[K]\times V}$. Now we can derive $r(S;\bar{\boldsymbol{\mu}}) - r(S;\boldsymbol{\mu})$ equals to

$$\sum_{j\in V, i\in[K]} (\bar{\mu}_i - \mu_i) \prod_{\ell=1}^{i-1}(1-\mu_{\ell,j}) \prod_{\ell=i+1}^{K}(1-\bar{\mu}_{\ell,j}). \tag{9}$$

$$\le \sum_{j\in V}\sum_{i\in[K]} (\zeta_{i,j})(\prod_{\ell=1}^{i-1}(1-\mu_{\ell,j}) \prod_{\ell=i+1}^{K}(1-\mu_{\ell,j}))$$

$$+ \sum_{j\in V}\sum_{i\in[K]} (\eta_{i,j}) \prod_{\ell=1}^{i-1}((1-\mu_{\ell,j})). \tag{10}$$

$$\le \sqrt{\sum_{j\in V, i\in[K]} \frac{\zeta_{i,j}^2 (p_{i,j}^{D,S})^2}{(1-\mu_{i,j})\mu_{i,j}}} \cdot \sqrt{\sum_{j\in V, i\in[K]} \prod_{\ell=i+1}^{K}(1-\mu_{\ell,j})\mu_{i,j}}$$

$$+ \sum_{j\in V}\sum_{i\in[K]} \eta_{i,j} p_{i,j}^{D,S}, \tag{11}$$

where the first inequality is by definition of $\zeta_{i,j}, \eta_{i,j}$ and $\bar{\mu}_{i,j} \ge \mu_{i,j}$, the second inequality is by Cauchy-Schwarz inequality and definition of $p_{i,j}^{D,S}$, concluding the lemma by $\sqrt{\sum_{j\in V}(1-\prod_{\ell=1}^K(1-\mu_{\ell,j}))} \le \sqrt{|V|}$.

For probabilistic feedback, let effective base arms $\boldsymbol{\mu} = (\boldsymbol{x}, \boldsymbol{y}) \in (0,1)^{(K|V|+|V|)}, \bar{\boldsymbol{\mu}} = (\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}}) \in (0,1)^{(K|V|+|V|)}$, where $\bar{\boldsymbol{x}} = \boldsymbol{\zeta}_x + \boldsymbol{\eta}_x + \boldsymbol{x}, \bar{\boldsymbol{y}} = \boldsymbol{\zeta}_y + \boldsymbol{\eta}_y + \boldsymbol{y}$, for $\boldsymbol{\zeta}, \boldsymbol{\eta} \in [-1,1]^{(n|V|+|V|)}$. For the target node $j \in V$, the per-target reward function $r_j(S;\boldsymbol{x},\boldsymbol{y}) = y_j(1-\prod_{i\in[n]}(1-x_{i,j}))$. Denote $\bar{p}_j^{D,S} = 1 - \prod_{i\in[n]}(1-\bar{x}_{i,j})$. Now we can derive $r(S;\bar{\boldsymbol{\mu}}) - r(S;\boldsymbol{\mu}) = \sum_{j\in V} r_j(S;\bar{\boldsymbol{x}},\bar{\boldsymbol{y}}) - r_j(S;\boldsymbol{x},\boldsymbol{y}) = \sum_{j\in V}\bar{y}_j\left(\prod_{i\in[n]}(1-x_{i,j}) - \prod_{i\in[n]}(1-\bar{x}_{i,j}))\right) + \sum_{j\in V}(\bar{y}_j - y_j)p_j^{D,S}$. For the first summation, we follow exactly the derivation of the cascading feedback, we have RHS $\le \sqrt{\sum_{j\in V, i\in[n]}(\frac{\zeta_{x,i,j}^2}{(1-x_{i,j})x_{i,j}}) + \sum_{j\in V}\frac{\zeta_{y,j}^2 (p_j^{D,S})^2}{(1-y_j)y_j}} \cdot \sqrt{\sum_{j\in V}\bar{y}_j^2 + (1-y_j)y_j} + (\sum_{j\in V, i\in[n]}|\eta_{x,i,j}| + \sum_{j\in V}|\eta_{y,j}|\, p_j^{D,S})$ and replacing $\sqrt{\sum_{j\in V}\bar{y}_j^2 + (1-y_j)y_j} \le \sqrt{|V|}$ concludes the proof.

For semi-bandit feedback, it is easy to follow the cascading feedback but set $p_{i,j}^{D,S} = 1$ if $i \in S$ and 0, otherwise.

C. Proof of Lemma 2

Proof. Under event $N_t^{s,1}$, we have $|\mu_i - \hat{\mu}_{t,i}| \le \rho_{t,i}$ by Lemma 6, hence the first and the second inequality in Lemma 2 holds. For the last inequality, it holds under event $N_t^{s,2}$ to replace $\hat{V}_{t-1,i}$ with $2\mu_i(1-\mu_i) + \frac{3.5\log t}{T_{t-1,i}})$. Since $\mathcal{N}_t^s = \mathcal{N}_t^{s,1} \bigcap \mathcal{N}_t^{s,2}$ and by Lemma 6, Lemma 2 holds with probability at least $1-4mt^{-2}$. ■

D. Proof of Lemma 3

Proof. Under event $\mathcal{N}_t^s$, by Lemma 2, it is easy to check that $\bar{\mu}_{t,i} \le \min\{\mu_{t-1,i} + 4\sqrt{3}\sqrt{\frac{\mu_i(1-\mu_i)\log t}{T_{t-1,i}}} + \frac{28\log t}{T_{t-1,i}}, 1\} \le \mu_{t-1,i} + 4\sqrt{3}\sqrt{\mu_i(1-\mu_i)(\frac{\log t}{T_{t-1,i}} \wedge \frac{1}{28})} + 28(\frac{\log t}{T_{t-1,i}} \wedge \frac{1}{28})$. Therefore, it holds that

$$\alpha r(S^*;\boldsymbol{\mu}) \le \alpha r(S^*;\bar{\boldsymbol{\mu}}_t) \le r(S_t;\bar{\boldsymbol{\mu}}_t) \tag{12}$$

$$\le r(S_t;\boldsymbol{\mu}) + e_{t,1}(S_t) + e_{t,2}(S_t), \tag{13}$$

where the first inequality is because the reward function is monotone and second inequality is due to the computation oracle, the third inequality is because of the inequality above and Lemma 1 by plugging in $\zeta_i = 4\sqrt{3}\sqrt{\mu_i(1-\mu_i)(\frac{\log t}{T_{t-1,i}} \wedge \frac{1}{28})}$ and $\eta_i = 28(\frac{\log t}{T_{t-1,i}} \wedge \frac{1}{28})$. So $Reg(T,\mathcal{N}_t^s) \le Reg(T,E_{t,1}) + Reg(T,E_{t,2})$. Now for $Reg(T,\neg\mathcal{N}_t^s)$, by Lemma 6 it holds that $Reg(T,\neg\mathcal{N}_t^s) \le \sum_{t=1}^T \Pr[\neg\mathcal{N}_t^s] \le \sum_{t=1}^T 4mt^{-2} \le \frac{2\pi^2}{3}m\Delta_{\max}$, which concludes the lemma. ■

E. Upper bound of $Reg(T,E_{t,1})$

Let $c_1 = 4\sqrt{3}$ be a constant and $O_t = \{i \in \tilde{S}_t : j_i^{S_t} \le j_i^{\max}\}$ where the threshold $j_i^{\max} = \frac{1}{2}(\lceil\log_2 \frac{c_1^2|V|K}{(\Delta_i^{\min})^2}\rceil + 1)$. Let $\alpha_1 > \alpha_2 > ... > \alpha_k > ... > \alpha_\infty$ and $1 = \beta_0 >$

$\beta_1 > ... > \beta_k > ... > \beta_\infty$ be two infinite sequences of positive numbers that are decreasing and converge to 0. Recall $\tilde{S}_t = \{i \in [m] : p_i^{D,S_t} > 0\}$. For positive integers k and t, we define $A_{t,k} = \{i \in \tilde{S}_t \cap O_{S_t} : N_{t-1,i,j_i^{S_t}} \leq \alpha_k \frac{g(K,\Delta_{S_t})f(t)}{\Delta_{S_t}^2}\}$, where $g(K,\Delta_{S_t})$ and $f(t)$ are going to be tuned for later use. Moreover, we define the complementary set $\bar{A}_{t,k} = \{i \in \tilde{S}_t \cap O_t : N_{t-1,i,j_i^{S_t}} > \alpha_k \frac{g(K,\Delta_{S_t})f(t)}{\Delta_{S_t}^2}\}$. Now we are ready to define the events $\mathbb{G}_{t,k} = \{|\bar{A}_{t,k}| \geq \beta_k K; \forall h < k, |A_{t,h}| < \beta_h K\}$. Let $\mathbb{G}_t = \bigcup_{k=1}^{\infty} \mathbb{G}_{t,k}$ and by definition its complementary $\overline{\mathbb{G}}_t = \{|A_{t,k}| < \beta_k K, \forall k \geq 1\}$. We first introduce a lemma that uses finite many events to conclude infinitely many events.

Lemma 8 (Lemma 7, [14])**.** *If there exists k_0 such that $\beta_{k_0} \leq 1/K$, then $\mathbb{G}_t = \bigcup_{k=1}^{k_0} \mathbb{G}_{t,k}$ and $\overline{\mathbb{G}}_t = \{|A_{t,k}| < \beta_k K, \forall 1 \leq k \leq k_0\}$.*

Now we bound $e_{t,2}(S_t)$ under events $\overline{\mathbb{G}}_t$ and N_t^t.

Lemma 9. *Under the event $\overline{\mathbb{G}}_t$ and N_t^t and if $\exists\, k_0$ such that $\beta_{k_0} \leq 1/K$, then $(e_{t,1}(S_t))^2 < \frac{6c_1^2|V|2^{(-j_i^{S_t}+1)}\log t \Delta_{S_t}^2 K}{g(K,\Delta_{S_t})f(t)}(\sum_{k=1}^{k_0} \frac{\beta_{k-1}-\beta_k}{\alpha_k} + \frac{\beta_{k_0}}{\alpha_{k_0}}) + \frac{\Delta_{S_t}^2}{8}$.*

Proof. By definition of $e_{t,1}(S_t)$ and event $\mathcal{N}_t^t$,

$$(e_{t,1}(S_t))^2 = \sum_{i\in\tilde{S}_t} c_1^2|V|(p_i^{D,S_t})^2 \min\{\frac{\log t}{T_{i,t-1}}, \frac{1}{28}\} \quad (14)$$

$$\leq \sum_{i\in\tilde{S}_t} c_1^2|V|(p_i^{D,S_t})^2 \min\{\frac{\log t}{\frac{1}{3}N_{t-1,i,j_i^{S_t}}2^{-j_i^{S_t}}}, \frac{1}{28}\}. \quad (15)$$

$$\leq \sum_{i\in\tilde{S}_t\cap O_{S_t}} c_1^2|V|(2^{-j_i^{S_t}+1})^2 \frac{\log t}{\frac{1}{3}N_{t-1,i,j_i^{S_t}}2^{-j_i^{S_t}}} + \frac{1}{28}\sum_{i\in\tilde{S}_t\cap\bar{O}_{S_t}} c_1^2|V|(2^{-j_i^{\max}+1})^2. \quad (16)$$

By setting the k_0 be the largest number that $\beta_{k_0} \leq 1/K$,

$$Eq.\ (16) \leq \sum_{k=1}^{k_0}\sum_{i\in\bar{A}_{t,k}\setminus\bar{A}_{t,k-1}} \frac{6c_1^2|V|2^{(-j_i^{S_t}+1)}\log t}{N_{t-1,i,j_i^{S_t}}} + \frac{\Delta_{S_t}^2}{8}$$

$$< \sum_{k=1}^{k_0} \frac{6c_1^2|V|2^{(-j_i^{S_t}+1)}\log t \Delta_{S_t}^2 |\bar{A}_{t,k}\setminus\bar{A}_{t,k-1}|}{\alpha_k g(K,\Delta_{S_t})f(t)} + \frac{\Delta_{S_t}^2}{8} \quad (17)$$

where the second inequality uses the definition of $\bar{A}_{t,k}$. The lemma is then concluded by Eq. (17)$< \frac{6c_1^2|V|2^{(-j_i^{S_t}+1)}\log t \Delta_{S_t}^2 K}{g(K,\Delta_{S_t})f(t)}(\sum_{k=1}^{k_0} \frac{\beta_{k-1}-\beta_k}{\alpha_k} + \frac{\beta_{k_0}}{\alpha_{k_0}}) + \frac{\Delta_{S_t}^2}{8}$, using the similar reason of Lemma 8 from [14]. ■

Now we set $g(K,\Delta_{S_t}) = 2^{(-j_i^{S_t}+1)}Kl$, where $l = \sum_{k=1}^{k_0} \frac{\beta_{k-1}-\beta_k}{\alpha_k} + \frac{\beta_{k_0}}{\alpha_{k_0}}$ and $f(t) = 48c_1^2|V|\log t$. By Lemma 9, under event $\mathcal{N}_t^t$, if $E_{t,1}$ holds, then $\mathbb{G}_t$ must hold.

For any arm i, let arm related event $\mathbb{G}_{t,k,i} = \mathbb{G}_{t,k} \bigcap \{i \in \tilde{S}_t, N_{t-1,i,j_i^{S_t}} \leq \alpha_k \frac{g(K,\Delta_{S_t})f(t)}{\Delta_{S_t}^2}, j_i^{S_t} \leq j_i^{\max}\}$. When $\mathbb{G}_{t,k}$ happens, we have $\mathbb{I}\{\mathbb{G}_{t,k}\} \leq \frac{1}{\beta_k K}\sum_{i\in[m]}\{\mathbb{G}_{t,k,i}\}$. We have

$$Reg(T, E_{t,1}\bigcap\mathcal{N}_t^t) \leq \sum_{t=1}^{T}\sum_{k=1}^{k_0}\Delta_{S_t}\mathbb{I}\{\mathbb{G}_{t,k}\} \quad (18)$$

$$\leq \sum_{t=1}^{T}\sum_{k=1}^{k_0}\sum_{i=1}^{m}\frac{\Delta_{S_t}}{K\beta_k}\mathbb{I}\{\mathbb{G}_{t,k,i}\}. \quad (19)$$

Let $\theta_k = \alpha_k Klf(t)$, and $(\Delta_{i,\ell})_{\ell\in[D_i]}$ be all possible gaps that are decreasing, RHS $\leq \sum_{i=1}^{m}\sum_{j=1}^{\infty}\sum_{k=1}^{k_0}\frac{1}{K\beta_k}\sum_{t=1}^{T}\sum_{n=1}^{D_i}\Delta_{i,n}\mathbb{I}\{i \in \tilde{S}_t, N_{i,j_i^{S_t},t-1} \leq \frac{\theta_k 2^{(-j+1)}}{\Delta_{i,n}^2}, \Delta_{S_t} = \Delta_{i,n}, j_i^{S_t} = j\} \leq \sum_{i=1}^{m}\sum_{j=1}^{\infty}\sum_{k=1}^{k_0}\frac{1}{K\beta_k}\sum_{t=1}^{T}\sum_{p=1}^{D_i}\Delta_{i,p}\mathbb{I}\{i \in \tilde{S}_t, N_{i,j_i^{S_t},t-1} \in (\frac{\theta_k 2^{(-j+1)}}{\Delta_{i,p-1}^2}, \frac{\theta_k 2^{(-j+1)}}{\Delta_{i,p}^2}], \Delta_{S_t} = \Delta_{i,n}, \Delta_{S_t} > 0, j_i^{S_t} = j\}$. Now if we bound the number of times the event happen to the length of interval Eq. (19) can be bounded by

$$\sum_{i=1}^{m}\sum_{j=1}^{\infty}\sum_{k=1}^{k_0}\frac{2^{(-j+1)}}{K\beta_k}(\frac{\theta_k}{\Delta_{i,1}} + \theta_k\sum_{p=2}^{D_i}\Delta_{i,p}(\frac{1}{\Delta_{i,p}^2} - \frac{1}{\Delta_{i,p-1}^2}))$$

$$\leq \sum_{i=1}^{m}\sum_{j=1}^{\infty}\sum_{k=1}^{k_0}\frac{2^{(-j+1)}}{K\beta_k}(\frac{\theta_k}{\Delta_{i,D_i}} + \theta_k\int_{\Delta_{i,D_i}}^{\Delta_{i,1}} x^{-2}dx) \quad (20)$$

$$\leq \sum_{i=1}^{m}\sum_{j=1}^{\infty}\sum_{k=1}^{k_0}\frac{2\theta_k 2^{(-j+1)}}{K\beta_k\Delta_{i,D_i}} \quad (21)$$

$$\leq \sum_{i=1}^{m}\left(1920c_1^2|V|\right)\lceil\frac{\log K}{1.61}\rceil^2\frac{\log T}{\Delta_i^{\min}}, \quad (22)$$

where the last inequality uses Lemma 11, Appendix C of [14] by setting $\alpha_k = \beta_k = 0.2^k$ and $\sum_{k=1}^{k0}\frac{\alpha_k}{\beta_k}l \leq 5\lceil\frac{\log K}{1.61}\rceil^2$, which concludes the lemma by adding $Reg(T, \neg\mathcal{N}_t^t) \leq \sum_{t=1}^{T}\sum_{i\in[m]} j_i^{\max}t^{-2}\Delta_{\max}$.

F. Upper bound of $Reg(T, E_{t,2})$

Bounding $Reg(T, E_{t,2})$ is very similar to that of $Reg(T, E_{t,1})$, we only state the key differences here and leave the proof to full technical reports. Let $c_2 = 28$ be a constant and $O_t = \{i \in \tilde{S}_t : j_i^{S_t} \leq j_i^{\max}\}$ with $j_i^{\max} = \lceil\log_2\frac{4c_2K}{\Delta_i^{\min}}\rceil + 1$. We also have $A_{t,k}, \bar{A}_{t,k}, \mathbb{G}_t, \mathbb{G}_{t,k}, \mathbb{G}_{t,k,i}$, but with different $g(K,\Delta_{S_t})$ and $f(t)$ which are settled by the following lemma.

Lemma 10. *Under the event $\overline{\mathbb{G}}_t$ and N_t^t and if $\exists\, k_0$ such that $\beta_{k_0} \leq 1/K$, then $e_{t,2}(S_t) < \frac{6c_2\log t\Delta_{S_t}^2 K}{g(K,\Delta_{S_t})f(t)}(\sum_{k=1}^{k_0}\frac{\beta_{k-1}-\beta_k}{\alpha_k} + \frac{\beta_{k_0}}{\alpha_{k_0}}) + \frac{\Delta_{S_t}}{4}$.*

Let $g(K,\Delta_{S_t}) = K\Delta_{S_t}l$, where $l = \sum_{k=1}^{k_0}\frac{\beta_{k-1}-\beta_k}{\alpha_k} + \frac{\beta_{k_0}}{\alpha_{k_0}}$ and $f(t) = 24c_2\log T$. We have $Reg(T, E_{t,2}\bigcap\mathcal{N}_t^t) \leq \sum_{t=1}^{T}\sum_{k=1}^{k_0}\Delta_{S_t}\mathbb{I}\{\mathbb{G}_{t,k}\} \leq \sum_{i=1}^{m}\sum_{j=1}^{j_i^{\max}}\sum_{k=1}^{k_0}\frac{1}{K\beta_k}(\theta_k + \theta_k\int_{\Delta_{i,D_i}}^{\Delta_{i,1}} x^{-1}dx) \leq 120c_2\sum_{i=1}^{m}\left(\log_2\frac{c_2K}{\Delta_i^{\min}}\right)\left(1 + \log\frac{\Delta_i^{\max}}{\Delta_i^{\min}}\right)\lceil\frac{\log K}{1.61}\rceil^2\log T$, similar to the proof after Lemma 9

REFERENCES

[1] D. S. Hochba, "Approximation algorithms for np-hard problems," *ACM Sigact News*, vol. 28, no. 2, pp. 40–52, 1997.

[2] M. Pathan, R. Buyya, and A. Vakali, "Content delivery networks: State of the art, insights, and imperatives," *Content Delivery Networks*, pp. 3–32, 2008.

[3] R. K. Ganti, F. Ye, and H. Lei, "Mobile crowdsensing: current state and future challenges," *IEEE communications Magazine*, vol. 49, no. 11, pp. 32–39, 2011.

[4] J. Zuo, X. Zhang, and C. Joe-Wong, "Observe before play: Multi-armed bandit with pre-observations," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 04, 2020, pp. 7023–7030.

[5] D. Kempe, J. Kleinberg, and É. Tardos, "Maximizing the spread of influence through a social network," in *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2003, pp. 137–146.

[6] A. Krause and C. Guestrin, "Near-optimal observation selection using submodular functions," in *AAAI*, vol. 7, 2007, pp. 1650–1654.

[7] L. Chen, J. Xu, S. Ren, and P. Zhou, "Spatio–temporal edge service placement: A bandit learning approach," *IEEE Transactions on Wireless Communications*, vol. 17, no. 12, pp. 8388–8401, 2018.

[8] W. Chen, Y. Wang, and Y. Yuan, "Combinatorial multi-armed bandit: General framework and applications," in *International Conference on Machine Learning*. PMLR, 2013, pp. 151–159.

[9] W. Chen, Y. Wang, Y. Yuan, and Q. Wang, "Combinatorial multi-armed bandit and its extension to probabilistically triggered arms," *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 1746–1778, 2016.

[10] B. Kveton, Z. Wen, A. Ashkan, and C. Szepesvari, "Tight regret bounds for stochastic combinatorial semi-bandits." in *AISTATS*, 2015.

[11] Q. Wang and W. Chen, "Improving regret bounds for combinatorial semi-bandits with probabilistically triggered arms and its applications," in *Advances in Neural Information Processing Systems*, 2017, pp. 1161–1171.

[12] D. P. Dubhashi and A. Panconesi, *Concentration of measure for the analysis of randomized algorithms*. Cambridge University Press, 2009.

[13] 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.

[14] R. Degenne and V. Perchet, "Combinatorial semi-bandit with known covariance," in *Advances in Neural Information Processing Systems*, 2016, pp. 2972–2980.

[15] N. Merlis and S. Mannor, "Tight lower bounds for combinatorial multi-armed bandits," in *Conference on Learning Theory*. PMLR, 2020, pp. 2830–2857.

[16] K. Han, C. Zhang, and J. Luo, "Taming the uncertainty: Budget limited robust crowdsensing through online learning," *Ieee/acm transactions on networking*, vol. 24, no. 3, pp. 1462–1475, 2015.

[17] M. Faulkner, M. Olson, R. Chandy, J. Krause, K. M. Chandy, and A. Krause, "The next big one: Detecting earthquakes and other rare events from community-based sensors," in *Proceedings of the 10th ACM/IEEE International Conference on Information Processing in Sensor Networks*. IEEE, 2011, pp. 13–24.

[18] Y. Chon, N. D. Lane, F. Li, H. Cha, and F. Zhao, "Automatically characterizing places with opportunistic crowdsensing using smartphones," in *Proceedings of the 2012 ACM conference on ubiquitous computing*, 2012, pp. 481–490.

[19] S. A. Bitaghsir, A. Dadlani, M. Borhani, and A. Khonsari, "Multi-armed bandit learning for cache content placement in vehicular social networks," *IEEE Communications Letters*, vol. 23, no. 12, pp. 2321–2324, 2019.

[20] D. Karamshuk, N. Sastry, M. Al-Bassam, A. Secker, and J. Chandaria, "Take-away tv: Recharging work commutes with predictive preloading of catch-up tv content," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 8, pp. 2091–2101, 2016.

[21] S. Wang, "Multichannel dqn channel model," https://github.com/ANRGUSC/MultichannelDQN-channelModel, 2018.

[22] H. Robbins, "Some aspects of the sequential design of experiments," *Bulletin of the American Mathematical Society*, vol. 58, no. 5, pp. 527–535, 1952.

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

[24] A. Slivkins *et al.*, "Introduction to multi-armed bandits," *Foundations and Trends® in Machine Learning*, vol. 12, no. 1-2, pp. 1–286, 2019.

[25] T. Lattimore and C. Szepesvári, *Bandit algorithms*. Cambridge University Press, 2020.

[26] P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite-time analysis of the multiarmed bandit problem," *Machine learning*, vol. 47, no. 2-3, pp. 235–256, 2002.

[27] J.-Y. Audibert, R. Munos, and C. Szepesvári, "Exploration–exploitation tradeoff using variance estimates in multi-armed bandits," *Theoretical Computer Science*, vol. 410, no. 19, pp. 1876–1902, 2009.

[28] N. Merlis and S. Mannor, "Batch-size independent regret bounds for the combinatorial multi-armed bandit problem," in *Conference on Learning Theory*. PMLR, 2019, pp. 2465–2489.

[29] X. Liu, J. Zuo, S. Wang, C. Joe-Wong, J. Lui, and W. Chen, "Batch-size independent regret bounds for combinatorial semi-bandits with probabilistically triggered arms or independent arms," *arXiv preprint arXiv:2208.14837*, 2022.

[30] M. Hefeeda and H. Ahmadi, "A probabilistic coverage protocol for wireless sensor networks," in *2007 IEEE International Conference on Network Protocols*. IEEE, 2007, pp. 41–50.

[31] J. Zuo, X. Liu, C. Joe-Wong, J. C. Lui, and W. Chen, "Online competitive influence maximization," in *International Conference on Artificial Intelligence and Statistics*. PMLR, 2022, pp. 11 472–11 502.

[32] X. Liu, J. Zuo, X. Chen, W. Chen, and J. C. Lui, "Multi-layered network exploration via random walks: From offline optimization to online learning," in *International Conference on Machine Learning*. PMLR, 2021, pp. 7057–7066.

[33] 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 (TON)*, vol. 20, no. 5, pp. 1466–1478, 2012.

[34] B. Kveton, Z. Wen, A. Ashkan, and C. Szepesvári, "Combinatorial cascading bandits," in *Proceedings of the 28th International Conference on Neural Information Processing Systems-Volume 1*, 2015, pp. 1450–1458.

[35] R. Combes, M. S. Talebi Mazraeh Shahi, A. Proutiere *et al.*, "Combinatorial bandits revisited," *Advances in neural information processing systems*, vol. 28, 2015.