COMBINATORIAL MULTI-ARMED BANDIT PROBLEM WITH PROBABILISTICALLY TRIGGERED ARMS: A CASE WITH BOUNDED REGRET

A.Ömer Sarıtaç

Department of
Industrial Engineering
Bilkent University, Ankara, Turkey

Cem Tekin

Department of
Electrical and Electronics Engineering
Bilkent University, Ankara, Turkey

ABSTRACT

In this paper, we study the combinatorial multi-armed bandit problem (CMAB) with probabilistically triggered arms (PTAs). Under the assumption that the arm triggering probabilities (ATPs) are positive for all arms, we prove that a simple greedy policy, named greedy CMAB (G-CMAB), achieves bounded regret. This improves the result in previous work, which shows that the regret is $O(\log T)$ under no such assumption on the ATPs. Then, we numerically show that G-CMAB achieves bounded regret in a real-world movie recommendation problem, where the action corresponds to recommending a set of movies, arms correspond to the edges between movies and users, and the goal is to maximize the total number of users that are attracted by at least one movie. In addition to this problem, our results directly apply to the online influence maximization (OIM) problem studied in numerous prior works.

***Index Terms*—** Combinatorial multi-armed bandit, probabilistically triggered arms, bounded regret, online learning.

1. INTRODUCTION

Multi-armed bandit (MAB) problem is a classical example of sequential decision making under uncertainty that has been extensively studied in the literature [1–7]. Two main approaches are used for learning in MAB problems: Upper confidence bound (UCB) based index policies [8] and Thompson sampling [1, 9–11]. CMAB with PTAs is an extension of the classical MAB problem [12–14], where actions chosen by the learner may trigger arms probabilistically. All of the works in this line of research use either UCB based index policies or Thompson sampling, where the UCBs or samples from the posterior are given as input to an offline combinatorial problem in each epoch. For instance, a UCB based algorithm that achieves $O(\log T)$ regret is proposed for this problem in [12]. Later, CMAB with PTAs is extended in [13], where the authors provided tighter regret bounds by getting rid of a problem parameter p^*, which denotes the minimum positive probability that an arm gets triggered by an action. This is achieved by introducing a new smoothness condition on the expected reward function. Based on this, $O(\log T)$ problem dependent and $\tilde{O}(\sqrt{T})$ problem independent regret bounds are proven.

In this paper, we consider an instance of CMAB with PTAs, where all of the ATPs are positive. For this instance, we prove a $O(1)$ regret bound, which implies that the regret is bounded. Moreover, we prove that a greedy algorithm, which always exploits the best action based on the estimated rewards on each epoch, is able to achieve bounded regret. Although not very common, bounded regret appears in various MAB problems, including some instances of parameterized MAB problems [15, 16]. However, these works do not conflict with the asymptotic $O(\log T)$ lower bound for the classical MAB problem [17], because in these works the reward from an arm provides information on the rewards from the other arms. We argue that bounded regret is also intuitive for our problem, because when the arms are probabilistically triggered, it is possible to observe the rewards from arms that never get selected.

A closely related problem is the influence maximization (IM) problem, which is first formulated in [18] as a combinatorial optimization problem, and has been extensively studied since then. The goal in this problem is to select a seed set of nodes that maximizes the influence spread in a social network. In this problem, the seed set corresponds to the action, edges correspond to arms, the influence spread corresponds to the reward, and the ATPs are determined by an influence spread process. Various works consider the online version of this problem, named the OIM problem [19–21]. In this version, the ATPs are unknown a priori. Works such as [12] and [13] solve this problem by using algorithms developed for CMAB with PTAs. Different than these, [19] considers an objective function, which is given as the expected size of the union of nodes influenced in each epoch over time, and [20] uses the node level feedback in addition to the edge level feedback used in prior works when updating the ATPs. In a related work [21], we introduce the contextual OIM problem, which

is a combination of contextual bandits with OIM, and propose an algorithm that achieves sublinear regret. Importantly, the bounded regret result we prove in this paper also applies to the OIM problem defined over a strongly connected graph where the influence probability on each edge is positive.

Our contributions can be summarized as follows:

- We theoretically show that the regret is bounded for CMAB with PTAs when ATPs are positive. This improves the previous $O(\log T)$ regret bounds that hold under a more general setting.
- We also numerically show that G-CMAB achives bounded regret for a movie recommendation problem with PTAs defined over a bipartate graph, and illustrate how the regret is affected by the size of the action and p^*.

2. PROBLEM FORMULATION

We adopt the notation of [13]. The system operates in discrete epochs indexed by t. There are m arms, whose states at each epoch are drawn from an unknown joint distribution D with support in $[0,1]^m$. The state of arm i at epoch t is denoted by $X_i^{(t)}$, and the state vector at epoch t is denoted by $\boldsymbol{X}^{(t)} := (X_1^{(t)}, \ldots, X_m^{(t)})$.

In each epoch t, the learner selects an action S_t from the finite set of actions $\mathcal{S}$ based on its history of actions and observations. Then, a random subset of arms $\tau_t \subseteq \{1, \ldots, m\}$ is triggered based on S_t and $\boldsymbol{X}^{(t)}$. Here, τ_t is drawn from a multivariate distribution (also called probabilistic triggering function) $D^{\text{trig}}(S_t, \boldsymbol{X}^{(t)})$ with support $[p^*, 1]^m$ for some $p^* > 0$, which is equivalent to saying that all the ATPs are positive. As we will show in the subsequent sections, this key assumption allows the learner to achieve bounded regret, without the need for explicit exploration. Then, at the end of epoch t, the learner obtains a finite, non-negative reward $R(S_t, \boldsymbol{X}^{(t)}, \tau_t)$ that depends deterministically on S_t, $\boldsymbol{X}^{(t)}$ and τ_t, and observes the states of the triggered arms, i.e., $X_i^{(t)}$, $i \in \tau_t$. The goal of the learner is to maximize its total expected reward over all epochs.

For each arm $i \in \{1, \ldots, m\}$, we let $\mu_i := \mathbb{E}_{\boldsymbol{X}^{(t)} \sim D}[X_i^{(t)}]$ denote the expected state of arm i and $\boldsymbol{\mu} := (\mu_1, \ldots, \mu_m)$. The expected reward of action S is $r_{\boldsymbol{\mu}}(S) := \mathbb{E}[R(S, \boldsymbol{X}, \tau)]$ where the expectation is taken over $\boldsymbol{X} \sim D$ and $\tau \sim D^{\text{trig}}(S, \boldsymbol{X})$. $r_{\boldsymbol{\mu}}(\cdot)$ is also called the expected reward function. The optimal action is given by $S^* \in \arg\max_{S \in \mathcal{S}} r_{\boldsymbol{\mu}}(S)$, and the expected reward of the optimal action is given by $r^*_{\boldsymbol{\mu}}$.

Computing the optimal action even when the expected states and the probabilistic triggering function are known is often an NP-hard problem for which (α, β)-approximation algorithms exist [22]. Due to this, we compare the performance of the learner with respect to an (α, β)-approximation oracle $\mathcal{O}$, which takes $\boldsymbol{\mu}$ as input and outputs the action $S^{\mathcal{O}}$ such that $\Pr(r_\mu(S^{\mathcal{O}}) \geq \alpha r^*_{\boldsymbol{\mu}}) \geq \beta$. Here, α denotes the approximation ratio and β denotes the minimum success probability.

The (α, β)-approximation regret (simply referred to as the regret) of the learner by epoch T is defined as follows:

$$\text{Reg}_{\boldsymbol{\mu},\alpha,\beta}(T) := T\alpha\beta r^*_{\boldsymbol{\mu}} - \mathbb{E}\left[\sum_{i=1}^{T} r_{\boldsymbol{\mu}}(S_t)\right].$$

3. A GREEDY ALGORITHM FOR CMAB AND ITS REGRET ANALYSIS

In this section, we propose the greedy CMAB (G-CMAB) algorithm and analyze its regret (the pseudocode is given in Algorithm 1).

For each arm $i \in \{1, \ldots, m\}$, G-CMAB keeps a counter T_i, which counts the number of times arm i is played (observed) and the sample mean estimate of its expected state $\hat{\mu}_i$. Let $\hat{\boldsymbol{\mu}} := \{\hat{\mu}_1, \ldots, \hat{\mu}_m\}$. We will use superscript t when explicitly referring to the counters and estimates that G-CMAB uses at epoch t. For instance, we have $T_i^t = \sum_{j=1}^{t-1} 1_{\{i \in \tau_j\}}$ and $\hat{\mu}_i^t = \frac{1}{T_i^t} \sum_{j=1}^{t-1} X_i^{(j)} 1_{\{i \in \tau_j\}}$. Initially, G-CMAB sets T_i to 0 and $\hat{\mu}_i$ to 1 for all arms. Then, in each epoch $t \geq 1$, it calls an (α, β)-approximation algorithm, which takes as input $\hat{\boldsymbol{\mu}}^t$ and chooses an action S_t. The action S_t depends on the randomness of the algorithm itself in addition to $\hat{\boldsymbol{\mu}}^t$. After selecting the action S_t, the learner observes the states $X_i^{(t)}$ for arms $i \in \tau_t$, and collects a random reward R_t. Then, it updates its estimate $\hat{\boldsymbol{\mu}}$ and the counters to be used in the next epoch.

Algorithm 1 G-CMAB

1: **Input:** Set of actions $\mathcal{S}$
2: **Initialize counters:** For each arm $i \in \{1, \ldots, m\}$, set $T_i = 0$, which is the number of times the arm i is observed, $t = 1$
3: **Initialize estimates:** Set $\hat{\mu}_i = 1$, $\forall i \in \{1, \ldots, m\}$, which is the sample mean estimate for μ_i
4: **while** $t \geq 1$ **do**
5: Call the (α, β)-approximation algorithm with $\hat{\boldsymbol{\mu}}$ as input to get S_t
6: Select action S_t, observe $X_i^{(t)}$ for $i \in \tau_t$ and collect the reward R_t
7: **for** $i \in \tau_t$ **do**
8: $T_i = T_i + 1$
9: $\hat{\mu}_i = \hat{\mu}_i + \frac{X_i^{(t)} - \hat{\mu}_i}{T_i}$
10: **end for**
11: $t = t + 1$
12: **end while**

We analyze the regret of G-CMAB under two mild assumptions on the reward function [12]. The first one states that the reward function is smooth and bounded.

Assumption 1. *$\exists f : \mathbb{R}^+ \cup \{0\} \to \mathbb{R}^+ \cup \{0\}$ such that f is continuous, strictly increasing, and $f(0) = 0$, where f is called the bounded-smoothness function. For any two expectation vectors, $\boldsymbol{\mu}$ and $\boldsymbol{\mu}'$, and for any $\Delta > 0$, we have*

$|r_{\boldsymbol{\mu}}(S) - r_{\boldsymbol{\mu}'}(S)| \leq f(\Delta)$, *if* $\max_{i\in\{1,\ldots,m\}} |\mu_i - \mu_i'| \leq \Delta$, $\forall S \in \mathcal{S}$.

The second assumption states that the expected reward is monotone under $\boldsymbol{\mu}$.

Assumption 2. *If for all arms* $i \in \{1,\ldots,m\}$, $\mu_i \leq \mu_i'$, *then we have* $r_{\boldsymbol{\mu}}(S) \leq r_{\boldsymbol{\mu}'}(S)$, $\forall S \in \mathcal{S}$.

For the regret analysis, we first show that the event that the number of times an arm is played by the end of epoch t is less than a linear function of t for some arm has a very low probability for t sufficiently large.

Theorem 1. *For* $\eta \in (0,1)$ *and for all integers* $t \geq t' := 4c^2/e^2$, *where* $c := 1/(p^*(1-\eta))^2$, *we have*

$$\Pr\left(\bigcup_{i\in\{1,\ldots,m\}} \{T_i^{t+1} \leq \eta p^* t\}\right) \leq \frac{m}{t^2}.$$

Proof. The proof can be found in the extended version of the paper [23]. □

Theorem 1 is the crux of achieving bounded regret since it guarantees that G-CMAB obtains sufficiently many observations from each arm without explicitly exploring any of the arms.

Next, we show that, provided that t is sufficiently large, the probability that $\Delta_t := \max_{i\in\{1,\ldots,m\}} |\mu_i - \hat{\mu}_i^t|$ is lower than a constant is high. This is a performance measure for how well the algorithm learns the state of each arm by the beginning of epoch t, and is related to Theorem 1 as it is related to how many times the learner plays an arm until epoch t.

Theorem 2. *When G-CMAB is run, for any* $\delta > 0$ *and* $\eta \in (0,1)$, *we have*

$$\Pr(\Delta_{t+1} \geq \delta) \leq \frac{2m}{t^2(1-e^{-2\delta^2})} + 2me^{-2\delta^2\eta p^* t}$$

for all integers $t \geq t' := 4c^2/e^2$, *where* $c := 1/(p^*(1-\eta))^2$.

Proof. The proof can be found in the extended version of the paper [23]. □

Let n_B denote the number of actions whose expected rewards are smaller than $\alpha r^*_{\boldsymbol{\mu}}$. These actions are called *bad actions*. We re-index the bad actions in increasing order, such that $S_{B,l}$ denotes the bad action with lth smallest expected reward. The set of bad actions is denoted by $\mathcal{S}_B := \{S_{B,1}, S_{B,2}, \ldots, S_{B,n_B}\}$. let $\nabla_l := \alpha r^*_{\boldsymbol{\mu}} - r_{\boldsymbol{\mu}}(S_{B,l})$ for each $l \in \{1,\ldots,n_B\}$ and $\nabla_{n_B+1} = 0$. Accordingly, we let $\nabla_{\max} := \nabla_1$, $\nabla_{\min} := \nabla_{n_B}$. In the next theorem, we show that the regret of G-CMAB is bounded for any $T > 0$.

Theorem 3. *The regret of G-CMAB is bounded, i.e.,* $\forall T \geq 1$

$$Reg_{\boldsymbol{\mu},\alpha,\beta}(T) \leq \nabla_{\max} \inf_{\eta\in(0,1)} \left(\lceil t' \rceil + \frac{m\pi^2}{3}\left(1 + \frac{1}{2\delta^2}\right) + 2m\left(1 + \frac{1}{2\delta^2\eta p^*}\right)\right)$$

where $\delta := f^{-1}(\nabla_{\min}/2)$, $t' := 4c^2/e^2$ *and* $c := 1/(p^*(1-\eta))^2$.

Proof. The proof can be found in the extended version of the paper [23]. □

This result is different from the prior results [12, 13, 24] where $O(\log T)$ problem dependent regret upper bounds are proven for the CMAB problem. The main difference of our problem from these works is that we assume the minimum ATP to be nonzero ($p^* > 0$). This allows us to prove the result in Theorem 1, by ensuring that each arm is triggered sufficiently many times without exploring rarely triggered arms.

4. ILLUSTRATIVE RESULTS

In this section, we evaluate the performance of G-CMAB on a recommendation problem. This problem has become popular among researchers with the popularization of on-demand media streaming services like Netflix. We use the *MovieLens* dataset for our experiments. The dataset contains 138k people who assigned 20M ratings to 27k movies between January 1995 and March 2015. We use the portion of the dataset that was collected between March 2014 and March 2015, which consists of 750k ratings. For our experiments, we choose 200 movies in total among the movies that were rated more than 200 times: 50 movies with the smallest ratings, 50 movies with the highest ratings, and 100 movies randomly.

4.1. Definition of the Recommendation Problem

The problem consists of a weighted bipartite graph $G = (L, R, E, p)$ where L denotes the set of movies, R denotes the set of users,[1] E denotes the set of edges between the users, and $p = \{p_{i,j}\}_{(i,j)\in E}$, where $p_{i,j}$ is the weight of edge (i,j), which corresponds to the probability that movie i influences (attracts) user j. The goal of the learner is to find a set $S \subseteq L$ of size k that maximizes the expected number of attracted nodes in R. This problem is an instance of the probabilistic maximum coverage problem [12]. Our problem extends this problem by allowing the nodes in S to trigger any $(i,j) \in E$ probabilistically. For instance, this can happen if the users also interact with each other in a social network, where the recommendation made to a user in the network may influence other users into watching the recommended movie via the word of mouth effect. Moreover, both the triggering and influence probabilities are initially unknown. We let $p_S^{i,j}$ denote the probability that action S triggers edge $(i,j) \in E$. The expected reward is defined as the expected total number of users that are attracted by at least one movie, and is given as $r_G(S) = \sum_{j\in R}(1 - \prod_{(i,j)\in E}(1 - p_S^{i,j} p_{i,j}))$. We assume

[1] Each user corresponds to a pool of individuals with same type of preferences over genres.

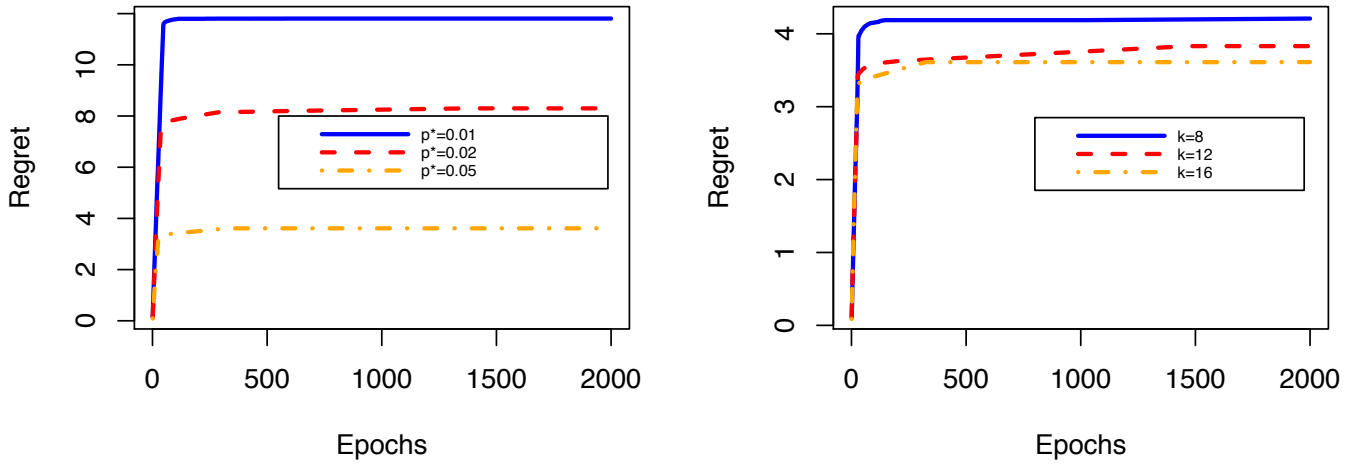

Fig. 1: Regret of G-CMAB for different p^* and k values.

that $p_S^{i,j} = 1$ for the outgoing edges of nodes $i \in S$. This assumption merely says that user j will watch movie i with probability $p_{i,j}$, when the movie is recommended to the user by the system. For the nodes $i \notin S$, $p_S^{i,j} < 1$. For these nodes, $p_S^{i,j}$ denotes the probability that user i gets to know about movie j by the word of mouth affect, without the recommender system showing the movie to the user. For simulations, we set $p_S^{i,j} = p^*$ for $p^* \in (0, 1)$, and evaluate the effect of different values of p^* on the performance of G-CMAB.

The above problem can be viewed as an instance of CMAB with PTAs. Each edge $(i, j) \in E$ is an arm and the state of each arm is a Bernoulli random variable with success probability $p_{i,j}$. For this problem, Assumption 1 is satisfied with the bounded-smoothness function $f(x) = |E|x$. The monotonicity assumption is also satisfied for this problem since increasing the $p_{i,j}$'s will definitely increase the expected reward. In addition, the reward function is submodular, and hence, it can be shown that using the greedy algorithm in [25], we can achieve $(1-1/e)$-approximation to the optimal reward. Hence, the greedy algorithm can be used as a $(1 - 1/e, 1)$-approximation algorithm.

4.2. Calculation of the Influence Probabilities

The MovieLens dataset contains the following attributes for each user: UserId, MovieId, Rating, TimeStamp, Title, and the Genre. Hence, we have the rating each user assigned to a movie with a particular genre and title. The dataset contains 20 genres. For each user $j \in R$ we first calculate the user preference vector $\mathbf{u_j}$, which is a unit vector, where each element of the vector corresponds to a coefficient representing how much the user likes a particular genre. We assume that the genre distribution of the movies that the users rated represents their genre preferences. Note that a movie can have multiple genres. We also create a 20 dimensional vector $\mathbf{g_i}$ for each movie i, and let $\mathbf{g_i}_k = 1$ if a movie belongs to genre k and 0 otherwise. Using this vector, we calculate the genre preference vector $\mathbf{u_j} = \frac{\sum_{i \in L} \mathbf{g_i} + \epsilon_{i,j}}{||\sum_{i \in L} \mathbf{g_i} + \epsilon_{i,j}||}$ for each user $j \in R$, where $\epsilon_{i,j} \sim$ Half-Normal$(\sigma = 0.05)$. The role of $\epsilon_{i,j}$ here is to account for the fact that the user may possibly explore new genres. Similarly, for each movie $i \in L$, we calculate the unit movie genre vector $\mathbf{m_i}$ as $\mathbf{g_i}/||\mathbf{g_i}||$. Using these, the influence probabilities are calculated as $p_{i,j} = sc \times \frac{<\mathbf{m_i}, \mathbf{u_j}> r_i}{\max r_i}$, $(i, j) \in E$, where r_i is the average rating given by all users to the movie i and sc is a scale factor in $(0, 1]$. This way, we took into account the quality in addition to the type (genre) of the movies in determining the influence probabilities.

4.3. Results

All of the presented results are for $p^* = 0.05$, $k = 16$, $sc = 0.2$ unless otherwise stated. In addition, to be able to make plausible comparisons between settings with different parameters, we consider a scaled version of the regret, where the regret is divided by the $\alpha\beta$ fraction of the optimal reward. $\alpha\beta$ fraction of the optimal reward is calculated by running the (α, β)-approximation algorithm, which is the greedy algorithm from [25] by giving the true influence probabilities as input. We observe from Fig.1 that the regret is bounded for different values of k and p^*. It is observed that G-CMAB incurs almost no regret after the first 300 epochs. Moreover, as p^* or k increases, the scaled version of the regret becomes smaller, which shows that learning becomes faster.

5. CONCLUSION

We consider a CMAB problem with positive ATPs and show that a greedy algorithm (G-CMAB) which does not perform explicit exploration, achieves bounded regret for any number of epochs T. We also show numerically that G-CMAB achieves bounded regret in a real-world movie recommendation scenario. These results show that exploration strategies are not necessary for learning algorithms that work in CMAB with PTAs with positive ATPs.

6. REFERENCES

[1] Suleman Alnatheer and Hong Man, "Multi-policy posterior sampling for restless Markov bandits," in *Proc. IEEE Global Conf. Signal and Information Processing (GlobalSIP)*, 2014, pp. 1271–1275.

[2] Cem Tekin and Mihaela van der Schaar, "Distributed online learning via cooperative contextual bandits," *IEEE Trans. Signal Process.*, vol. 63, no. 14, pp. 3700–3714, 2015.

[3] Hyun-Suk Lee, Cem Tekin, Mihaela van der Schaar, and Jang-Won Lee, "Contextual learning for unit commitment with renewable energy sources," in *Proc. IEEE Global Conf. Signal and Information Processing (GlobalSIP)*, 2016, pp. 866–870.

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

[5] Donald A. Berry and Bert Fristedt, *Bandit problems: Sequential allocation of experiments*, Springer, 1985.

[6] Richard S. Sutton and Andrew G. Barto, *Reinforcement learning : An introduction*, MIT Press, 1998.

[7] Sébastien Bubeck and Nicolò 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.

[8] Peter Auer, Nicolò Cesa-Bianchi, and Paul Fischer, "Finite-time analysis of the multiarmed bandit problem," *Machine Learning*, vol. 47, no. 2-3, pp. 235–256, 2002.

[9] William R. Thompson, "On the likelihood that one unknown probability exceeds another in view of the evidence of two samples," *Biometrika*, vol. 25, no. 3/4, pp. 285–294, 1933.

[10] Shipra Agrawal and Navin Goyal, "Analysis of Thompson sampling for the multi-armed bandit problem," in *Proc. COLT*, 2012, pp. 39.1–39.26.

[11] Daniel Russo and Benjamin van Roy, "Learning to optimize via posterior sampling," *Mathematics of Operations Research*, vol. 39, no. 4, pp. 1221–1243, 2014.

[12] Wei Chen, Yajun Wang, Yang Yuan, and Qinshi Wang, "Combinatorial multi-armed bandit and its extension to probabilistically triggered arms," *J. Mach. Learn. Res.*, vol. 17, no. 1, pp. 1746–1778, 2016.

[13] Qinshi Wang and Wei Chen, "Tighter regret bounds for influence maximization and other combinatorial semi-bandits with probabilistically triggered arms," *arXiv preprint arXiv:1703.01610*, 2017.

[14] Zheng Wen, Branislav Kveton, and Michal Valko, "Online influence maximization under independent cascade model with semi-bandit feedback," *arXiv preprint arXiv:1605.06593*, 2017.

[15] Adam J. Mersereau, Paat Rusmevichientong, and John N. Tsitsiklis, "A structured multiarmed bandit problem and the greedy policy," *IEEE Trans. Autom. Control*, vol. 54, no. 12, pp. 2787–2802, 2009.

[16] Onur Atan, Cem Tekin, and Mihaela van der Schaar, "Global multi-armed bandits with Hölder continuity," in *Proc. AISTATS*, 2015, pp. 28–36.

[17] Tze L. Lai and Herbert Robbins, "Asymptotically efficient adaptive allocation rules," *Advances in Applied Mathematics*, vol. 6, no. 1, pp. 4–22, 1985.

[18] David Kempe, Jon Kleinberg, and Éva Tardos, "Maximizing the spread of influence through a social network," in *Proc. 9th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, 2003, pp. 137–146.

[19] Siyu Lei, Silviu Maniu, Luyi Mo, Reynold Cheng, and Pierre Senellart, "Online influence maximization," in *Proc. 21th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, 2015, pp. 645–654.

[20] Sharan Vaswani, Laks Lakshmanan, Mark Schmidt, et al., "Influence maximization with bandits," *arXiv preprint arXiv:1503.00024*, 2015.

[21] A.Ömer Sarıtaç, Altuğ Karakurt, and Cem Tekin, "Online contextual influence maximization," in *Proc. 54th Annual Allerton Conference on Communication, Control, and Computing*, 2016, pp. 1204–1211.

[22] Vijay V. Vazirani, *Approximation algorithms.*, Springer, 2001.

[23] A.Ömer Sarıtaç and Cem Tekin, "Combinatorial multi-armed bandit with probabilistically triggered arms: A case with bounded regret," *arXiv preprint arXiv:1707.07443*, 2017.

[24] Branislav Kveton, Zheng Wen, Azin Ashkan, and Csaba Szepesvari, "Combinatorial cascading bandits," in *Advances in Neural Information Processing Systems*, 2015, pp. 1450–1458.

[25] 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.