Learning and Selecting the Right Customers for Reliability: A Multi-armed Bandit Approach

Yingying Li, Qinran Hu, and Na Li

***Abstract*— In this paper, we consider residential demand response (DR) programs where an aggregator calls upon some residential customers to change their demand so that the total load adjustment is as close to a target value as possible. Major challenges lie in the uncertainty and randomness of the customer behaviors in response to DR signals, and the limited knowledge available to the aggregator of the customers. To learn and select the right customers, we formulate the DR problem as a combinatorial multi-armed bandit (CMAB) problem with a reliability goal. We propose a learning algorithm: CUCB-Avg (Combinatorial Upper Confidence Bound-Average), which utilizes both upper confidence bounds and sample averages to balance the tradeoff between exploration (learning) and exploitation (selecting). We prove that CUCB-Avg achieves $O(\log T)$ regret given a time-invariant target, and $o(T)$ regret when the target is time-varying. Simulation results demonstrate that our CUCB-Avg performs significantly better than the classic algorithm CUCB (Combinatorial Upper Confidence Bound) in both time-invariant and time-varying scenarios.**

I. Introduction

Demand response (DR) has been playing an increasing role in reducing the operation cost and improving the sustainability of the power grid [1]–[9]. Most of the existing successful DR programs are for commercial and industrial customers. As residential demand takes up to almost 40% of the U.S. electricity consumption [10], there is a growing effort in designing residential DR in both academia and industry. In a typical residential DR program, there is a DR aggregator such as a utility company requests load changes from users, for example, by changing the temperature set points of the air conditioners. To encourage users' participation, most of the residential DR programs use incentive schemes such as prices, rewards, coupons, raffles, etc, under the assumption that customers are price responsive [7]–[9]. However, because the average monetary reward budget for single household is usually small, it is reported that rewards play a limited role for users to decide whether to participate in or opt out of a DR program [8].

On the other side, there are many factors besides rewards that affect residential user decisions, such as house size and type, household demographics, outdoor humidity and temperature, people's lifestyles, etc. However, the DR aggregator has limited knowledge of these factors. It is also unclear how these factors will affect people's DR action. Moreover, people with similar factors might react to the same DR signal in very different ways. These intrinsic, heterogeneous uncertainties associated with the residential customers call for learning approaches to understand and interact with the customers in a smarter way.

Multi-armed bandit (MAB) emerges as a natural framework to handle such uncertainties [11], [12]. In the simplest setting, MAB considers n independent arms, each providing a random contribution according to its own distribution at time $1 \leq t \leq T$. Without knowing these distributions, a decision maker picks one arm at each time step, and tries to maximize the total expected contribution. The decision maker should decide whether to *explore* arms to learn the unknown distribution, or to *exploit* the current knowledge by selecting the arm that has been providing the highest contribution. When the decision maker can select multiple arms at each time, the problem is referred as CMAB (Combinatorial MAB) in literature [13]–[17]. (C)MAB captures a fundamental tradeoff in most learning problems: *exploration vs. exploitation*. A common metric to evaluate the performance of (C)MAB learning algorithms is the regret, which captures the difference between the optimal value assuming the distributions are known and the achieved value of the online learning algorithm. A sublinear regret implies good performance because it indicates that the learning algorithm eventually learns the optimal solution.

When applying CMAB framework to residential demand response, we can treat each customer as one arm. Then the aggregator follows CMAB methods to explore (learn) and exploit (select) the customers to achieve the goal of its DR program. There exist studies of DR via (C)MAB [9], [18]–[20]. However, most literature sets the goal as maximizing the load reduction for peak hours without considering the load reduction target and reliability issues.

Our Contributions: In this paper, we formulate the DR as a CMAB problem whose objective is to minimize the deviation between the total load adjustment and a target level for the power system reliability. We consider a large number of residential customers, each of whom can commit one unit of load change (either reduction or increase) with an unknown probability. The task of the aggregator is to select a subset of the customers to approximate the target level as close as possible. The size of the subset is not fixed, giving flexibility to the aggregator to achieve different target levels. Compared with the classic CMAB literature [13]–[17], a major difference of our formulation is that the reliability objective leads to a non-monotonic objective function for the CMAB problem, making the existing CMAB approaches and regret analysis inapplicable here.

This work was supported by NSF CAREER 1553407 and ARPA-E NODES. Y. Li, Q. Hu, and N. Li are with the School of Engineering and Applied Sciences, Harvard University, 33 Oxford Street, Cambridge, MA 02138, USA (email: yingyingli@g.harvard.edu, qinranhu@g.harvard.edu, nali@seas.harvard.edu).

In order to design our CMAB online learning algorithm, we first study the corresponding offline combinatorial optimization problem assuming the probabilities of customers are known. Based on the structure of the offline algorithm, we propose an online algorithm CUCB-Avg (Combinatorial Upper Confidence Bound-Average) and provide a rigorous regret analysis. We show that, over T time steps, CUCB-Avg achieves $O(\log T)$ regret given a static target and $o(T)$ regret given a time-varying target. The dependence of regret in both cases on dimension n (the number of customers) is polynomial. By simulation, we show that the performance of CUCB-Avg is much better than the classic algorithm CUCB [13], [16], [17] and similar to Thompson sampling, another popular CMAB method which has good empirical performance but is usually difficult to theoretically analyze [21]–[23].

Related Work in CMAB. Most literature in CMAB studies a classic formulation which aims to maximize the total (weighted) contribution of K arms with a fixed integer K (and known weights) [12], [14]–[17], [24]. As for more general problem formulation, Chen et. al. study the monotone objective function: the objective value function is monotonically nondecreasing with arms' parameters given a fixed selected subset [13]. They propose the Combinatorial Upper Confidence Bound (CUCB) using the principle of *optimism in the face of uncertainty*. Another line of work follows the *Bayesian* approach. [25] studies a Bayesian learning algorithm, Thompson sampling, for general CMAB problems, but its analysis is based on several assumptions including finite prior distribution and the uniqueness of the optimal solution, and the regret bound consists of a large exponential term.

However, our CMAB problem with the reliability objective function does not satisfy the conditions of monotonicity or the uniqueness of the optimal solution, and a properly selected prior distribution for our problem may not satisfy the assumption in [25]. Therefore, either the learning approaches or the analysis in current literature do not suit our CMAB problem, motivating us to design new CMAB algorithms.

Organization of the Paper. Section II introduces the problem formulation. Section III introduces an offline algorithm and an online algorithm CUCB-Avg. Section IV studies the performance guarantee for a time-invariant target. Section V studies the performance guarantee for a time-varying target. Section VI provides the simulation results.

Notations. Given a set E, and a universal set U, the complement of set E is denoted as $\bar{E}$, the cardinality of set E is $|E|$. For any positive integer n, let $[n] = \{1, \ldots, n\}$. Let $I_E(x)$ denote the indicator function on set U such that $I_E(x) = 1$ if $x \in E$ and $I_E(x) = 0$ if $x \notin E$. When $k = 0$, the summation $\sum_{i=1}^{k} a_i = 0$ for any a_i, and the set $\{\sigma(1), \ldots, \sigma(k)\} = \emptyset$ for any $\sigma(i)$. Finally, we define the big-O and small-o notations. For $x = (x_1, \ldots, x_k) \in \mathbb{R}^k$, we write $f(x) = O(g(x))$ as $x \to +\infty$ if there exists a constant M such that $|f(x)| \leq M|g(x)|$ for any x such that $x_i \geq M \ \forall\, i \in [k]$; and we write $f(x) = o(g(x))$ if $\lim_{x\to+\infty} f(x)/g(x) = 0$. We usually omit the phase "as $x \to +\infty$" for simplicity. When studying the asymptotic behavior near zero, we consider the inverse of x.

II. Problem Formulation

Motivated by the discussion in the previous section, we will formulate the DR as a CMAB problem in this section. We focus on the load reduction to illustrate the problem. The load increase can be treated in the same way.

Consider a demand response (DR) program with an aggregator and n residential customers (arms) over T time steps where each time step corresponds to one DR event.[1] Each customer may respond to a DR event by reducing one unit of power consumption with probability $0 \leq p_i \leq 1$, or not respond at all. The demand reduction by customer i at time step t is denoted by $X_{t,i}$, which is assumed to follow Bernoulli distribution: $X_{t,i} \sim \text{Bern}(p_i)$ and is independent across time[2].

At each time $1 \leq t \leq T$, there is a DR event with a demand reduction target $D \geq 0$ determined by the power system. This reduction target might be due to a sudden drop of renewable energy generation or a peak load reduction request, etc. The aggregator aims to select a subset of customers $S_t \subseteq [n]$, such that the total demand reduction is as close to the target as possible. The loss/cost at time t can be captured by the squared deviation of the total reduction from the target D:

$$L_t(S_t) = (\sum_{i \in S_t} X_{t,i} - D)^2$$

Since demand reduction $X_{t,i}$ are random, the goal is to minimize the expected squared deviation,

$$\min_{S_t \subseteq [n]} \mathbb{E}\, L_t(S_t). \tag{1}$$

In this paper, we will first study the scenario where the target D is time-invariant. Then we will extend the result to the scenario where the target D_t is time-varying to incorporate different DR signals resulting from fluctuations of power supply and demand.

When the response probability profile $p = (p_1, \ldots, p_n)$ is known, the problem (1) is a combinatorial optimization, and an offline optimization algorithm is provided in Section III. The optimal solution is denoted by S_t^*.

In reality, the probabilities of response are usually unknown. Thus, the aggregator should learn the probabilities from the feedback of previous demand response events, then make online decisions to minimize the difference between the total demand reduction and the target D. The learning performance is measured by Regret(T), which compares the total expected cost of online decisions and the optimal total

[1]The specific definition of DR event and the duration of each event is up to the choice of the system designer. Our methods can accommodate different scenarios.

[2]For simplicity, we only consider that each customer has one unit to reduce. Our method can be easily extended to multi-unit setting and/or the setting where different users have different size of units. But the regret analysis will be more complicated which we leave as future work.

expected costs in T time steps[3]:

$$\text{Regret}(T) \coloneqq \mathbb{E}[\sum_{t=1}^{T} R_t(S_t)] \tag{2}$$

where $R_t(S_t) \coloneqq L_t(S_t) - L_t(S_t^*)$ and the expectation is taken with respect to random $X_{t,i}$ and possibly random S_t.

The feedback of previous demand response events includes the responses of every selected customer, i.e., $\{X_{t,i}\}_{i\in S_t}$. Such feedback structure is called *semi-bandit* in literature [13], and carries more information than bandit feedback which only includes the realized cost $L_t(S_t)$.

Lastly, we note that our problem formulation can be applied to other applications beyond demand response. One example is introduced below.

Example 1. *Consider a crowd-sourcing related problem. Given a fixed budget D, a survey planner sends out surveys and offers one unit of reward for each participant. Each potential participant may participate with probability p_i. Let $X_{t,i} = 1$ if agent i participates; and $X_{t,i} = 0$, if agent i ignores the survey. The survey planner wants to maximize the total number of responses without exceeding the budget too much. One possible formulation is to select subset S_t such that the total number of responses is close to the budget D,*

$$\min_{S_t}(\sum_{i\in S_t} X_{t,i} - D)^2$$

Since the participation probabilities are unknown, the survey planner can learn the participation probabilities from the previous actions of its selected agents and then try to minimize the total costs during the learning process.

III. Algorithm Design

In this section, we first analyze the offline optimization problem and provide an optimization algorithm. Then we introduce the notations for online algorithm analysis, and discuss two simple algorithms: greedy algorithm and CUCB (Combinatorial Upper Confidence Bound). Finally, we present our online algorithm CUCB-Avg, and provide intuitions behind the algorithm design.

A. Offline Optimization

When the probability profile p is known, the problem (1) becomes a combinatorial optimization problem:

$$\min_{S\subseteq[n]} \mathbb{E}\, L(S) \Leftrightarrow \min_{S\subseteq[n]} (\sum_{i\in S} p_i - D)^2 + \sum_{i\in S} p_i(1-p_i) \tag{3}$$

Though combinatorial optimization is NP-hard in general and only has approximate algorithms, the problem (3) admits a simple optimal algorithm, as shown in Algorithm 1. Roughly speaking, Algorithm 1 takes two steps: i) rank the arms according to p_i, ii) determine the number k according to the probability profile p and the target D and select the top k arms. The output of Algorithm 1 is denoted by $\phi(p, D)$ which is a subset of $[n]$. In the following theorem, we show that such algorithm finds an optimal solution to (3).

[3]Strictly speaking, this is the definition of pseudo-regret, because its benchmark is the optimal expected cost: $\min_{S_t\subseteq[n]} \mathbb{E}\, L_t(S_t)$, instead of the optimal cost for each time, i.e. $\min_{S_t\subseteq[n]} L_t(S_t)$.

Algorithm 1: Offline optimization algorithm

1: **Inputs:** n, $p_1, \ldots, p_n$, $D > 0$
2: Rank p_i in a non-increasing order:

$$p_{\sigma(1)} \geq \cdots \geq p_{\sigma(n)}$$

3: Find the smallest $k \geq 0$ such that

$$\sum_{i=1}^{k} p_{\sigma(i)} > D - 1/2$$

or $k = n$ if

$$\sum_{i=1}^{n} p_{\sigma(i)} \leq D - 1/2$$

Ties are broken randomly.
4: **Ouputs**: $\phi(p, D) = \{\sigma(1), \ldots, \sigma(k)\}$

Theorem 1. *For any $D > 0$, the output of Algorithm 1, $\phi(p, D)$, is an optimal solution to* (3)*.*

Proof Sketch. We defer the detailed proof to Appendix A and only introduce the intuition here. To solve (3), we need two things: i) the total expected contribution of S, $\sum_{i\in S} p_i$, is closed to the target D, ii) the total variance of arms in S is minimized. i) is guaranteed by Line 3 of Algorithm 1: it is easy to show that $|\sum_{i\in\phi(p,D)} p_i - D| \leq 1/2$. ii) is guaranteed by only selecting arms with higher response probability, as indicated by Line 2 of Algorithm 1. The intuition is given below. Consider an arm with large parameter p_1 and two arms with smaller parameters p_2, p_3. To make analysis easier, we assume $p_1 = p_2 + p_3$. Thus replacing p_1 with p_2, p_3 will not affect the first term in (3). However,

$$p_1(1-p_1) \leq p_2(1-p_2) + p_3(1-p_3)$$

by $p_1^2 = (p_2+p_3)^2 \geq p_2^2 + p_3^2$. Therefore, replacing one arm with higher response probability by two arms with lower response probabilities will only increase the variance. □

Remark 1. *There might be more than one optimal subset. Algorithm 1 only outputs one of them.*

Corollary 1. *When $D \leq 1/2$, $\phi(p, D) = \emptyset$ is optimal.*

Notice that when $D \leq 1/2$, the optimal subset $\phi(p, D) = \emptyset$ does not depend on p. Therefore, in the online setting, we can always find an optimal subset for $D \leq 1/2$ even without any knowledge of p.

B. Notations for Online Algorithms

Let $\bar{p}_i(t)$ denote the sample average of parameter p_i by time t (including time t), then

$$\bar{p}_i(t) = \frac{1}{T_i(t)} \sum_{\tau\in I_i(t)} X_{\tau,i}$$

where $I_i(t)$ denotes the set of times steps when arm i was selected by time t and $T_i(t) = |I_i(t)|$ denotes the number of times that arm i has been selected by time t. Let $\bar{p}(t) =$

$(\bar{p}_1(t), \ldots, \bar{p}_n(t))$. Notice that before making decisions at time t, only $\bar{p}(t-1)$ is available.

C. Two Simple Online Algorithms: Greedy Algorithm and CUCB

In this subsection, we introduce two simple algorithms: greedy algorithm and CUCB, and explain why they perform poorly in our problem.

Greedy algorithm is initialized by selecting every arm at time $t = 1$. It then uses the sample average of each parameter $\bar{p}_i(t-1)$ as an estimation of unknown probability p_i and chooses a subset based on the offline oracle described in Algorithm 1, i.e. $S_t = \phi(\bar{p}(t-1), D)$. The greedy algorithm is expected to perform poorly because it only exploits the current information, but fails to explore the unknown information, as demonstrated below.

Example 2. *Consider two arms with parameters $p_1 > p_2$. The goal is to select the arm with the higher parameter. Now, suppose after some time steps, we have explored the suboptimal arm 2 for enough times, such that the sample average provides a good estimation $\bar{p}_2 \approx p_2$, but haven't explored the optimal arm 1 enough so that the sample average is under-estimated: $\bar{p}_1 < \bar{p}_2 < p_1$. If we apply greedy algorithm, we will keep selecting the suboptimal arm 2 based on current information: $\bar{p}_1, \bar{p}_2$, but fails to explore arm 1's information. As a result, the regret will be $O(T)$.*

A well-known algorithm in CMAB literature that balances the exploration and exploitation is CUCB [13], [17]. Instead of using sample average $\bar{p}$ directly, CUCB modifies the sample average by adding a confidence interval radius,

$$U_i(t) = \min(\bar{p}_i(t-1) + \sqrt{\frac{\alpha \log t}{2T_i(t-1)}}, 1) \tag{4}$$

where α is a positive parameter of the algorithm. Then CUCB applies the offline oracle $S_t = \phi(U(t), D)$ where $U(t) = (U_1(t), \ldots, U_n(t))$. $U_i(t)$ is restricted to $[0,1]$ in case $U(t)$ is outside the domain of the oracle ϕ. $U_i(t)$ is also known as the upper confidence bound of p_i, and is restricted to $[0,1]$ allows the algorithm to balance exploration and exploitation, because it carries the information of both the sample average, and the number of exploration times $T_i(t-1)$. CUCB performs well in classic CMAB problems, such as maximizing the total contribution of K arms for a fixed number K [13], [17].

However, CUCB performs poorly in our problem, as demonstrated by simulations in Section VI. The major problem of CUCB is the over-estimation of the arm parameter p. By choosing $S_t = \phi(U(t), D)$ based on upper confidence bounds, CUCB selects less arms than needed, which not only results in a large distance between the total load reduction and the target, but also discourages exploration.

D. Our Proposed Online Algorithm: CUCB-Avg

Based on our discussion above, we propose a new algorithm, CUCB-Avg. The novelty of our algorithm is that it utilizes both sample averages and upper confidence bounds by exploiting the structure of the offline optimal algorithm.

Algorithm 2: CUCB-Avg

1: **Notations:** $T_i(t)$ is the number of times selecting arm i by time t, and $\bar{p}_i(t)$ is the sample average of arm i by time t (both including time t).
2: **Inputs:** $\alpha > 2$, D
3: **Initialization:** At $t = 1$, play $S_1 = [n]$, compute $T_i(1)$, $\bar{p}_i(1)$ according to the observation $\{X_{1,i}\}_{i \in [n]}$
4: **for** $t = 2, \ldots, T$ **do**
5: Compute the upper confidence bound for each i:

$$U_i(t) = \min(\bar{p}_i(t-1) + \sqrt{\frac{\alpha \log t}{2T_i(t-1)}}, 1)$$

6: Rank $U_i(t)$ by a non-increasing order: $U_{\sigma(t,1)}(t) \geq \cdots \geq U_{\sigma(t,n)}(t)$.
7: Find the smallest $k_t \geq 0$ such that

$$\sum_{i=1}^{k_t} \bar{p}_{\sigma(t,i)}(t-1) > D - 1/2$$

or $k_t = n$ if $\sum_{i=1}^{n} \bar{p}_{\sigma(t,i)}(t-1) \leq D - 1/2$.
8: Play $S_t = \{\sigma(t,1), \ldots, \sigma(t,k_t)\}$. Update $T_i(t)$ and $\bar{p}_i(t)$ according to the observation $\{X_{t,i}\}_{i \in S_t}$
9: **end for**

We note that the offline algorithm 1 selects the right subset of arms in two steps: i) rank (top) arms, ii) determine the number k of the top k arms to select. In CUCB-Avg, we use the upper confidence bound $U_i(t)$ to rank the arms in a non-increasing order. This is the same as CUCB. However, the difference is that our CUCB-Avg uses the sample average $\bar{p}_i(t-1)$ to decide the number of arms to select at time t. The details of algorithm are given in Algorithm 2.

Now we explain why the ranking rule and the selection rule of CUCB-Ave are expected to work for our problem. The ranking rule is determined by $U_i(t)$ and an arm with larger $U_i(t)$ is given a priority to be selected at time t. We note that $U_i(t)$ is the summation of two terms: the sample average $\bar{p}_i(t-1)$ and the confidence interval radius that is related to how many times the arm has been explored. Therefore, if an arm i) has a small $T_i(t-1)$ indicating that arm i has not been explored enough or ii) has a large $\bar{p}_i(t-1)$ indicating that arm i might have larger parameter p_i, then arm i tends to have a larger $U_i(t)$ and thus is given a priority to be selected. In this way, CUCB-Avg selects both under-explored arms (*exploration*) and arms with large parameters (*exploitation*).

When determining k, CUCB-Avg uses the sample averages and selects enough arms such that the total sample average is close to D. Compared with CUCB which uses upper confidence bounds to determine k, our algorithm selects more arms, which reduces the distance between the total reduction and the target, and also encourages exploration.

IV. Regret analysis

In this section, we will prove that our algorithm CUCB-Avg achieves $O(\log T)$ regret when D is time invariant.

A. The Main Result

The next theorem upper bounds the regret of CUCB-Avg.

Theorem 2. *Consider n arms with parameter $p = (p_1, \ldots, p_n)$ over T time steps. There exists a constant $\epsilon_0 > 0$ determined by p and D, such that for any $\alpha > 2$, the regret of CUCB-Avg is upper bounded by*

$$\text{Regret}(T) \leq M(1 + \frac{2n}{\alpha - 2}) + \frac{\alpha M n \log T}{2\epsilon_0^2} \tag{5}$$

where $M = \max(D^2, (n-D)^2)$.

Before the formal proof, we make a few comments.

Regret Bound. The regret bound in (5) is $O(n^3 \log T)$ because $M \sim O(n^2)$ and ϵ_0 is a constant determined by p and D. The bound is referred as *distribution-dependent bound* in literature as p is invariant with horizon T [12].

Choice of α. α shows up in two terms: $\frac{2Mn}{\alpha-2}$ and $\frac{\alpha M n \log T}{2\epsilon_0^2}$. The first term grows when α decreases, and the second one decreases when α decreases. Since the second term is $O(\log T)$ while the first term is constant with respect to T, α should be chosen to be close to 2 when T is large.

Role of ϵ_0. We defer the explicit expression of ϵ_0 to Appendix C and only explain the intuition behind ϵ_0 here. To start with, we explain why the upper bound in (5) decreases when ϵ_0 increases. Roughly speaking, ϵ_0 is a robustness measure of our offline optimal algorithm, in the sense that if the probability profile p is perturbed to be $\bar{p}$ by ϵ_0 (i.e., $|\bar{p}_i - p_i| < \epsilon_0$ for all i), Algorithm 1's output $\phi(\bar{p}, D)$ would still be optimal for the true profile p. Intuitively, if ϵ_0 is large, the learning task is easy because we are able to find an optimal subset given a poor estimation, leading to a small regret.

To give a rough idea of what factors will affect the robustness measure ϵ_0, we provide an explicit expression of ϵ_0 under two assumptions in the following proposition.

Proposition 1. *If the following two assumptions hold,*
(A1): p_i are positive and distinct: $p_{\sigma(1)} > \cdots > p_{\sigma(n)} > 0$
(A2): There exists $k \geq 1$ such that

$$\sum_{i=1}^{k} p_{\sigma(i)} > D - 1/2$$
$$\sum_{i=1}^{k-1} p_{\sigma(i)} < D - 1/2$$

then the ϵ_0 in Theorem 2 can be determined by:

$$\epsilon_0 = \min(\frac{\delta_1}{k}, \frac{\delta_2}{k}, \frac{\Delta_k}{2}) \tag{6}$$

where

$$\begin{aligned} &k = |\phi(p, D)| \\ &\sum_{i=1}^{k} p_{\sigma(i)} = D - 1/2 + \delta_1, \\ &\sum_{i=1}^{k-1} p_{\sigma(i)} = D - 1/2 - \delta_2, \\ &\Delta_i = p_{\sigma(i)} - p_{\sigma(i+1)}, \forall\ i = 1, \ldots, n-1 \end{aligned} \tag{7}$$

We defer the proof to Appendix B and only make two comments on the proposition here. Firstly, it is easy to verify that Assumptions (A1) and (A2) imply $\epsilon_0 > 0$. Secondly, we verify that ϵ_0 defined in (6) is the robustness measure. Essentially, we need to show that if $\forall\, i,\ |\bar{p}_i - p_i| < \epsilon_0$, we have $\phi(\bar{p}, D) = \phi(p, D) \coloneqq \{\sigma(1), \ldots, \sigma(k)\}$. We prove this in two steps. Step 1: when $\epsilon_0 \leq \frac{\Delta_k}{2}$, the k arms with higher $\bar{p}_i$ are the same k arms with higher p_i because for any $1 \leq i \leq k$ and $k+1 \leq j \leq n$, we have $\bar{p}_{\sigma(i)} > p_{\sigma(k)} - \epsilon_0 \geq p_{\sigma(k+1)} + \epsilon_0 > \bar{p}_{\sigma(j)}$. Step 2: because $\epsilon_0 \leq \frac{\delta_1}{k}, \frac{\delta_2}{k}$, we have i) $\sum_{i=1}^{k} \bar{p}_{\sigma(i)} > \sum_{i=1}^{k}(p_{\sigma(i)} - \epsilon_0) = D - 1/2 + \delta_1 - k\epsilon_0 \geq D - 1/2$ and ii) $\sum_{i=1}^{k-1} \bar{p}_{\sigma(i)} < \sum_{i=1}^{k-1}(p_{\sigma(i)} + \epsilon_0) = D - 1/2 - \delta_2 + (k-1)\epsilon_0 \leq D - 1/2$ Thus, we have shown that $\phi(\bar{p}, D) = \{\sigma(1), \ldots, \sigma(k)\}$.

Finally, we briefly discuss how to generalize the expression of ϵ_0 in (6) to the case without (A1) and (A2). When (A1) does not hold, we only consider the gap between the arms that are not in a tie, i.e. $\{\Delta_i |\ \Delta_i > 0,\ 1 \leq i \leq n-1\}$. When (A2) does not hold and $\sum_{i=1}^{k-1} p_{\sigma(i)} = D - 1/2$, we consider less than $k-1$ arms to make the total expected contribution below $D - 1/2$. For the explicit expression of ϵ_0, we refer the reader to Appendix C.

Comparsion with the regret bound of classic CMAB. In classic CMAB literature whose goal is to select K arms with highest parameters for a fixed integer K, the regret bound depends on $\frac{\Delta_K}{2}$ [17]. We note that $\frac{\Delta_K}{2}$ plays the same role as the ϵ_0 in our problem, as it is the robustness measure of the classic problem above. That is, given any estimation $\bar{p}$ with estimation error at most $\Delta_K/2$: $\forall\, i,\ |\bar{p}_i - p_i| < \Delta_K/2$, the highest K arms with the profile $\bar{p}$ are the same highest K arms with the profile p.

In addition, the regret bound in literature is $O(\frac{\log T}{\Delta_K/2})$ as Δ_K goes to zero [17], while our regret bound in (5) is $O(\frac{\log T}{\epsilon_0^2})$. This difference may be due to technical reasons.

B. Proof of Theorem 2

Proof outline: We will divide the T time steps into four parts, and bound the regret in each part separately. Then Theorem 2 is proved by summing up the four regret bounds.

The time steps are partitioned based on event E_t, and $B_t(\epsilon_0)$, which are defined below. Let E_t denote the event when the sample average is outside the confidence interval considered in Algorithm 2:

$$E_t \coloneqq \{\exists\, i \in [n],\ |\bar{p}_i(t-1) - p_i| \geq \sqrt{\frac{\alpha \log t}{2T_i(t-1)}}\}$$

For any $\epsilon > 0$, let $B_t(\epsilon)$ denote the event when Algorithm 2 selects an arm who has been explored for no more than $\frac{\alpha \log T}{2\epsilon^2}$ times:

$$B_t(\epsilon) \coloneqq \{\exists\, i \in S_t,\ s.t.\ T_i(t-1) \leq \frac{\alpha \log T}{2\epsilon^2}\} \tag{8}$$

Let $\epsilon_0 > 0$ be a small number such that Lemma 3 holds.

Now, we will define the four parts of the T time steps, and briefly introduce the regret bound of each part.

1) *Initialization.* The regret of initialization at $t = 1$ is bounded by a constant (Inequality (9)).

2) *Time steps when E_t happens.* The regret is bounded by a constant because E_t happens rarely due to concentration properties in statistics (Lemma 1).
3) *Time steps when $\bar{E}_t$ and $B_t(\epsilon_0)$ happen.* The regret is at most $O(\log T)$ because $B_t(\epsilon_0)$ occurs for at most $O(\log T)$ times (Lemma 2).
4) *Time steps when $\bar{E}_t$ and $\bar{B}_t(\epsilon_0)$ happen.* No regret due to enough exploration of the selected arms (Lemma 3).

Notice that time steps are not divided sequentially here. For example, time steps $t = 1$ and $t = 3$ may belong to Part 2 while time step $t = 2$ belongs to Part 3.

Detailed proof: Firstly, we note that for all time steps $1 \leq t \leq T$, the regret is upper bounded by

$$R_t(S_t) \leq L_t(S_t) \leq \max(D^2, (n-D)^2) =: M \tag{9}$$

Thus, the regret of initialization at $t = 1$ is bounded by M.

The number of time steps in Part 2 is small because the sample average $\bar{p}_i(t)$ concentrates around the true value p_i with a high probability. In the following, we first state a classic concentration bound: Chernoff-Hoeffding's inequality and then use it to bound the regret when E_t happens.

Theorem 3 (Chernoff-Hoeffding's inequality)**.** *$X_1, \ldots, X_m$ i.i.d in $[0, 1]$ with mean μ, then*

$$\mathbb{P}(|\sum_{i=1}^{m} X_i - m\mu| \geq m\epsilon) \leq 2e^{-2m\epsilon^2} \tag{10}$$

Lemma 1. *When $\alpha > 2$,*

$$\mathbb{E}\sum_{t=2}^{T} I_{E_t} R_t(S_t) \leq \frac{2Mn}{\alpha - 2}$$

Proof. By Inequality (9) and the fact $I_{E_t} \geq 0$, we have,

$$\mathbb{E}\sum_{t=2}^{T} I_{E_t} R_t(S_t) \leq M\,\mathbb{E}\sum_{t=2}^{T} I_{E_t}.$$

So we only need to bound $\mathbb{E}\sum_{t=2}^{T} I_{E_t}$.

$$\begin{aligned}
&\mathbb{E}\sum_{t=2}^{T} I_{E_t} = \sum_{t=2}^{T} \mathbb{P}(E_t) \\
&\leq \sum_{t=2}^{T}\sum_{i=1}^{n} \mathbb{P}\left(|\bar{p}_i(t-1) - p_i| \geq \sqrt{\frac{\alpha \log t}{2T_i(t-1)}}\right) \\
&\leq \sum_{t=2}^{T}\sum_{i=1}^{n}\sum_{s=1}^{t-1} \mathbb{P}(|\bar{p}_i(t-1) - p_i| \geq \sqrt{\frac{\alpha \log t}{2s}}, T_i(t-1) = s) \\
&\leq \sum_{t=2}^{T}\sum_{i=1}^{n}\sum_{s=1}^{t-1} \frac{2}{t^\alpha} \leq \sum_{t=2}^{T} n\frac{2}{t^{\alpha-1}} \leq \frac{2n}{\alpha - 2}
\end{aligned}$$

where the first inequality is by enumerating possible $i \in [n]$, the second inequality is by enumerating possible values of $T_i(t-1)$: $\{1, \ldots, t-1\}$, the third inequality is by Chernoff-Hoeffding's inequality, and the last inequality is by $\sum_{t=2}^{T} \frac{1}{t^{\alpha-1}} \leq \int_1^{+\infty} \frac{1}{t^{\alpha-1}} \leq \frac{1}{\alpha-2}$. □

Next, we show Part 3 contributes at most $O(\log T)$ regret.

Lemma 2. *For any $\epsilon_0 > 0$,*

$$\mathbb{E}\sum_{t=2}^{T} R_t(S_t) I_{\{\bar{E}_t, B_t(\epsilon_0)\}} \leq \frac{\alpha M n \log T}{2\epsilon_0^2}$$

Proof. By Inequality (9) and the fact $I_{\{\bar{E}_t, B_t(\epsilon_0)\}} \geq 0$,

$$\mathbb{E}\sum_{t=2}^{T} R_t(S_t) I_{\{\bar{E}_t, B_t(\epsilon_0)\}} \leq M\,\mathbb{E}\sum_{t=2}^{T} I_{\{B_t(\epsilon_0)\}}$$

Now, by the definition of $B_t(\epsilon_0)$ in (8), whenever $B_t(\epsilon_0)$ happens, the algorithm selects an arm i that has not been selected for $\frac{\alpha \log T}{2\epsilon_0^2}$ times, increasing the selection time counter $T_i(t)$ by one. Thus, such event can happen for at most $\frac{\alpha n \log T}{2\epsilon_0^2}$ times. □

When $\bar{E}_t$ and $\bar{B}_t(\epsilon_0)$ happen (Part 4), every selected arm is fully explored and every arm's sample average is within the confidence interval. As a result, CUCB-Avg selects the right subset and hence contributes zero regret. This is formally stated in the following lemma.

Lemma 3. *There exists $\epsilon_0 > 0$, such that for each $1 \leq t \leq T$, if $\bar{E}_t$ and $\bar{B}_t(\epsilon_0)$ happen, CUCB-Avg selects an optimal subset, i.e., S_t is optimal. Accordingly, $\mathbb{E}\, R_t(S_t) I_{\{\bar{E}_t, \bar{B}_t(\epsilon_0)\}} = 0$, where the expectation is taken with respect to the algorithm history by time $t-1$: $\mathcal{F}_{t-1} := \{S_\tau, \{X_{\tau,i}\}_{i \in S_\tau}\}_{\tau=1}^{t-1}$.*

Proof Sketch: We defer the proof to Appendix B and C and sketch the proof ideas here, which is based on two facts:
Fact 1: when $\bar{E}_t$ and $\bar{B}(\epsilon_0)$ happen, the upper confidence bounds can be bounded by

$$U_i(t) > p_i, \ \forall\, i \in [n]$$

and the confidence bounds of the selected arm j satisfy

$$|\bar{p}_j(t-1) - p_j| < \epsilon_0, \ U_j(t) < p_j + 2\epsilon_0, \ \forall\, j \in S_t.$$

Fact 2: when ϵ_0 is small enough, CUCB-Avg selects an optimal subset.

To get the intuition for Fact 2, we assume Assumption (A1) and (A2) in Proposition 1 hold and consider the expression of ϵ_0 by (6). We also denote $\phi(p, D) = \{\sigma(1), \ldots, \sigma(k)\}$. In the following, we roughly explain why the selected subset S_t is optimal given ϵ_0 defined in (6):

i) By $\epsilon_0 \leq \frac{\Delta_k}{2}$, we can show that the selected subset S_t is either a superset or a subset of the optimal subset $\{\sigma(1), \ldots, \sigma(k)\}$
ii) By $\epsilon_0 \leq \delta_1/k$, we can show that we will not select more than k arms, because, informally, even if we underestimate p_i, the sum of arms in $\{\sigma(1), \ldots, \sigma(k)\}$ is still larger than $D - 1/2$.
iii) By $\epsilon_0 \leq \delta_2/k$, we can show that we will not select less than k arms, because, informally, even if we overestimate p_i, the sum of $k-1$ arms in $\{\sigma(1), \ldots, \sigma(k)\}$ is still smaller than $D - 1/2$.

As a result, $S_t = \{\sigma(1), \ldots, \sigma(k)\}$ according to Line 7 of Algorithm 2. □

Finally, we prove Theorem 2 by summing up the bounds:

$$\begin{aligned}\text{Regret}(T) &= \mathbb{E}\, R_1(S_1) + \mathbb{E}\sum_{t=2}^{T} R_t(S_t) I_{\{E_t\}} \\ &+ \mathbb{E}\sum_{t=2}^{T} R_t(S_t) I_{\{\bar{E}_t, B_t(\epsilon_0)\}} + \mathbb{E}\sum_{t=2}^{T} R_t(S_t) I_{\{\bar{E}_t, \bar{B}_t(\epsilon_0)\}} \\ &\le M + \frac{2Mn}{\alpha - 2} + \frac{\alpha M n \log T}{2\epsilon_0^2}\end{aligned}$$

□

V. Time-varying target

In practice, for each DR event t, the reduction target $D_t \ge 0$ might be different. Thus, it is worth studying the performance of CUCB-Avg given the time-varying target. In the following, we will show how to extend CUCB-Avg to the time-varying case, then provide a sublinear regret bound.

The adaptation of Algorithm 2 to time-varying case is straightforward. At each time t, the decision maker uses the current target D_t to run the algorithm by letting $D = D_t$.

Next, we will analyze the regret of CUCB-Avg in time-varying case. Notice that we impose no assumption on the dynamics of D_t except that it is bounded, which is almost always the case in practice.

Assumption 1. *There exists a finite $\bar{D} > 0$ such that*

$$0 \le D_t \le \bar{D}, \qquad \forall\ 1 \le t \le T$$

The next Theorem provides an upper bound of the regret of CUCB-Avg, the proof of which is deferred to Appendix D.

Theorem 4. *Suppose Assumption 1 holds. For any $\alpha > 2$, the regret of CUCB-Avg is upper bounded by*

$$\begin{aligned}\text{Regret}(T) &\le \bar{M} + \frac{2\bar{M}n}{\alpha - 2} + 9n^2 \left(\bar{M}\log T\right)^{\frac{2}{3}} T^{\frac{1}{3}} \\ &+ 6n \left(\bar{M}\log T\right)^{\frac{1}{3}} T^{\frac{2}{3}} \\ &+ \max\left(\frac{\alpha \bar{M} n \log T}{2\beta^2}, \frac{\alpha n}{2}\left(\bar{M}\log T\right)^{\frac{1}{3}} T^{\frac{2}{3}}\right)\end{aligned}$$

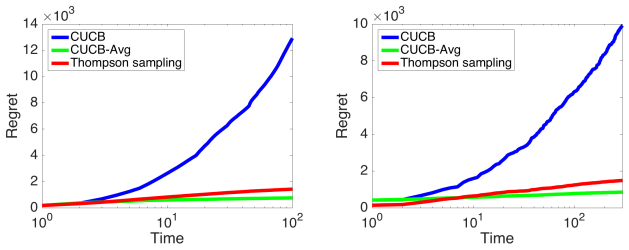
where $\beta = \min\{\frac{\Delta_i}{2} |\ \Delta_i > 0\}$ and $\bar{M} = \max(\bar{D}^2, n^2)$.

We defer the proof to Appendix D and make some comments here.

Discussion of $\bar{M}$. We note that $\bar{M}$ is the upper bound on the single-step regret $R_t(S_t)$ given any possible target D_t and realization of $X_{t,i}$.

Regret bound. The dominating term of the regret with respect to T is $(6n + \frac{\alpha n}{2})\left(\bar{M}\log T\right)^{\frac{1}{3}} T^{\frac{2}{3}}$, which is $O(n^{5/3}\log(T)^{1/3}T^{2/3})$ by $\bar{M} \sim O(n^2)$. The dominating term with respect to n is $9n^2\left(\bar{M}\log T\right)^{\frac{2}{3}} T^{\frac{1}{3}}$, which is $O(n^{10/3}\log(T)^{2/3}T^{1/3})$ by $\bar{M} \sim O(n^2)$. In total, the regret bound is $O(n^{5/3}\log(T)^{1/3}T^{2/3} + n^{10/3}\log(T)^{2/3}T^{1/3})$.

The regret bound is worse than the bound given a static target because D_t can change in an adversarial way. When D_t is randomly generated, the regret is usually much better than the worst case regret bound, as shown in Section VI.



(a) Static case (b) Time-varying case

Fig. 1: The figures compare the regret of CUCB, CUCB-Avg and Thompson sampling. In both cases, CUCB-Avg has $O(\log T)$ regret, greatly outperforms CUCB, and performs similarly to Thompson sampling.

Role of β. β captures the gap between any two arms with similar mean values. Roughly speaking, it plays a similar role to ϵ_0 in the static case. Intuitively, when β is small, the learning task is difficult because our algorithm cannot rank the arms correctly without a good estimation. Thus a smaller β results in a larger regret, which aligns with Theorem 4.

VI. Numerical Experiments

In this section, we numerically study the performance of CUCB-Avg for residential DR, and compare it with classic bandit algorithms such as CUCB and Thompson sampling [13], [21]. We will show that CUCB-Avg performs much better than CUCB and similarly to Thompson sampling.

A. Thompson sampling

Thompson sampling is a Bayesian algorithm that views the unknown probability vector p as a random vector with a prior distribution. It is fundamentally different from all the algorithms mentioned above, which all view p as an unknown but deterministic vector. Thompson sampling is well-known for its good empirical performance in classical CMAB problems [22], [23], [25], thus it is worth comparing our algorithm with Thompson sampling by simulation for our problem. The theoretical analysis of Thompson sampling is both limited and complicated, thus we leave for future work the regret analysis of Thompson sampling for our problem.

For the reader's convenience, we briefly explain the algorithm procedures here. Thompson sampling first selects the subset S_t based on sample $\hat{p}_t$ from the prior distribution at $t = 1$ (or the posterior distribution at $t \ge 2$) and the offline oracle ϕ: $S_t = \phi(\hat{p}_t, D)$, then updates the posterior distribution by the feedback from the selected subset, $\{X_{t,i}\}_{i \in S_t}$. For more details, we refer the reader to [21].

B. Time-invariant target D

This section studies a residential DR program with a time-invariant load-reduction target. Specifically, we consider $n = 100$ customers, whose response probability p_i is uniformly randomly drawn from $[0, 1]$ for all i. Let the load-reduction target be $D = 35$ units, and the time horizon be $T = 100$. Set the algorithm parameter as $\alpha = 2.1$.

Figure 1a plots the regret of CUCB, CUCB-Avg and Thompson sampling with a logarithmic scale for the x-axis

based on 200 independent simulations. The prior distribution of p is chosen to be the uniform distribution on $[0,1]^n$. Firstly, the figure shows that the regret of CUCB-Avg is linear with respect to $\log(T)$, which matches our theoretical result in Theorem 2. In addition, the classic algorithm CUCB performs poorly in our problem, generating regret almost linear in T. This is aligned with our intuition in Section III. Finally, the figure shows that CUCB-Avg and Thompson sampling have similar performance. In this scenario, CUCB-Avg performs slightly better, but we note that there exist other scenarios where Thompson sampling is slightly better.

C. Time-varying target D_t

This section studies a DR program with time-varying load-reduction targets. Specifically, we consider $n = 100$ customers, whose mean value p_i is uniformly randomly drawn from $[0,1]$ for all $i \in [n]$. Let time horizon be $T = 300$. Consider target D_t to be independently uniformly drawn from $[10, 30]$. Set the algorithm parameter as $\alpha = 2.05$.

Figure 1b plots the regret of CUCB, CUCB-Avg and Thompson sampling with a logarithmic scale for the x-axis based on 200 independent simulations. The prior distribution of p is chosen to be the uniform distribution on $[0,1]^n$. Interestingly, the figure shows that CUCB-Avg still guarantees $O(\log(T))$ regret in the time-varying case, better than the regret upper bound $O(T^{\frac{2}{3}}(\log T)^{\frac{1}{3}})$ in Theorem 4. This is because the regret bound in Theorem 4 considers the worst case scenario of D_t. The regret can be much better when D_t does not behave adversarially, as in our experiment. In addition, we see that CUCB-Avg generates larger regret than Thompson sampling at the first 6 time steps. This is because CUCB-Avg conducts more exploration at the beginning. Finally, similar to the static case, Figure 1b shows the poor performance of CUCB, and the similar performance of CUCB-Avg and Thompson sampling. In this scenario, CUCB-Avg performs slightly better, but there exist other scenarios where Thompson sampling is slightly better.

VII. Conclusion

In this paper, we study a combinatorial multi-armed bandit problem motivated by residential demand response with the goal of minimizing the difference between the total load adjustment and a target value. We propose a new algorithm CUCB-Avg, and apply it to both static and time-varying cases. We show that when the target is time-invariant, CUCB-Avg achieves $O(\log T)$ regret. When the target is time-varying case, our algorithm achieves $o(T)$ regret. The numerical results also confirm the performance of the algorithm. Future work includes 1) studying the performance guarantee of Thompson sampling, 2) deriving the lower bound of the regret for our problem, 3) generalizing the model to handle dynamic population and other load reduction models of customers, e.g. continuous distribution, Markov processes.

References

[1] P. Siano, "Demand response and smart grid – a survey," *Renewable and Sustainable Energy Reviews*, vol. 30, no. C, pp. 461–478, 2014.

[2] M. H. Albadi and E. F. El-Saadany, "Demand response in electricity markets: An overview," in *Power Engineering Society General Meeting, 2007. IEEE*, June 2007, pp. 1–5.

[3] N. Li, L. Chen, and S. H. Low, "Optimal demand response based on utility maximization in power networks," in *2011 IEEE power and energy society general meeting*. IEEE, 2011, pp. 1–8.

[4] "PJM: Demand response," http://www.pjm.com/markets-and-operations/demand-response.aspx, 2018.

[5] "NYISO demand response program," http://www.nyiso.com/public /markets_operations/market_data/demand_response/index.jsp.

[6] F. Rahimi and A. Ipakchi, "Demand response as a market resource under the smart grid paradigm," *IEEE Transactions on Smart Grid*, vol. 1, no. 1, pp. 82–88, 2010.

[7] Y. Li and N. Li, "Mechanism design for reliability in demand response with uncertainty," in *American Control Conference (ACC), 2017*. IEEE, 2017, pp. 3400–3405.

[8] "Reports on Demand Response and Advanced Metering ," Federal Energy Regulatory Commission, Tech. Rep., 12 2017.

[9] D. O'Neill, M. Levorato, A. Goldsmith, and U. Mitra, "Residential demand response using reinforcement learning," in *Smart Grid Communications (SmartGridComm), 2010 First IEEE International Conference on*. IEEE, 2010, pp. 409–414.

[10] "Electric power monthly," https://www.eia.gov/electricity/ monthly/epm_table_grapher.php?t=epmt_5_01, 2018.

[11] 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.

[12] 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.

[13] 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.

[14] R. Combes, M. S. T. M. Shahi, A. Proutiere *et al.*, "Combinatorial bandits revisited," in *Advances in Neural Information Processing Systems*, 2015, pp. 2116–2124.

[15] 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.

[16] B. Kveton, Z. Wen, A. Ashkan, H. Eydgahi, and B. Eriksson, "Matroid bandits: Fast combinatorial optimization with learning," *arXiv preprint arXiv:1403.5045*, 2014.

[17] B. Kveton, Z. Wen, A. Ashkan, and C. Szepesvari, "Tight regret bounds for stochastic combinatorial semi-bandits," in *Artificial Intelligence and Statistics*, 2015, pp. 535–543.

[18] Q. Wang, M. Liu, and J. L. Mathieu, "Adaptive demand response: Online learning of restless and controlled bandits," in *Smart Grid Communications (SmartGridComm), 2014 IEEE International Conference on*. IEEE, 2014, pp. 752–757.

[19] A. Lesage-Landry and J. A. Taylor, "The multi-armed bandit with stochastic plays," *IEEE Transactions on Automatic Control*, 2017.

[20] S. Jain, B. Narayanaswamy, and Y. Narahari, "A multiarmed bandit incentive mechanism for crowdsourcing demand response in smart grids." in *AAAI*, 2014, pp. 721–727.

[21] D. Russo, B. Van Roy, A. Kazerouni, and I. Osband, "A tutorial on thompson sampling," *arXiv preprint arXiv:1707.02038*, 2017.

[22] S. Agrawal and N. Goyal, "Analysis of thompson sampling for the multi-armed bandit problem," in *Conference on Learning Theory*, 2012, pp. 39–1.

[23] O. Chapelle and L. Li, "An empirical evaluation of thompson sampling," in *Advances in neural information processing systems*, 2011, pp. 2249–2257.

[24] J.-Y. Audibert, S. Bubeck, and G. Lugosi, "Regret in online combinatorial optimization," *Mathematics of Operations Research*, vol. 39, no. 1, pp. 31–45, 2013.

[25] A. Gopalan, S. Mannor, and Y. Mansour, "Thompson sampling for complex online problems," in *International Conference on Machine Learning*, 2014, pp. 100–108.

[26] Y. Li, Q. Hu, and N. Li. (2018) Learning and targeting the right customers for reliability: A multi-armed bandit approach (extended version). [Online]. Available: https://nali.seas.harvard.edu/files/nali/files/2018cdcmab.pdf

APPENDIX

A. Proof of Theorem 1

Only in this subsection, we assume $p_1 \geq \cdots \geq p_n$ to simplify the notation. This will not cause any loss of generality in the offline analysis. In other parts of the document, the order of $p_1, \ldots, p_n$ is unknown, and we will use $p_{\sigma(1)} \geq \ldots p_{\sigma(n)}$ to denote the non-increasing order of parameters.

Since adding or removing an arm i with $p_i = 0$ will not affect the regret, in the following we will assume $p_i > 0$ without loss of generality [4].

The proof of Theorem 1 takes two steps.

1) In Lemma 4, we establish a local optimality condition, which is a necessary condition to the global optimality.
2) In Lemma 5, we show that there exists an optimal selection S^* which includes first k arms with an unknown k. Then we can easily show that the Algorithm 1 selects the optimal k.

We first state and prove Lemma 4.

Lemma 4. *Suppose S^* is optimal and $p_i > 0$ for $i \in S^*$. Then we must have*

$$\sum_{i \in S^* - \{j\}} p_i \leq D - 1/2, \quad \forall\ j \in S^*$$

If $S^* \neq [n]$, then we will also have

$$\sum_{i \in S^*} p_i \geq D - 1/2$$

Proof. Since S^* is optimal, removing an element will not reduce the expected loss, i.e.

$$\mathbb{E}\, L(S^*) \leq \mathbb{E}\, L(S^* - \{j\})$$

which is equivalent with

$$\begin{aligned}
&\mathbb{E}\, L(S^*) - \mathbb{E}\, L(S^* - \{j\}) \\
&= (2 \sum_{i \in S^*} p_i - 2D - p_j) p_j + p_j(1 - p_j) \\
&= (2 \sum_{i \in S^*} p_i - 2D + 1 - 2p_j) p_j \leq 0
\end{aligned}$$

Since $p_j > 0$, we must have

$$\sum_{i \in S^* - \{j\}} p_i \leq D - 1/2$$

If $S^* \neq [n]$, then adding an element will not reduce the cost. So we must have

$$\mathbb{E}\, L(S^*) \leq \mathbb{E}\, L(S^* \cup \{j\}), \quad \forall j \notin S^*$$

which is equivalent with

$$\begin{aligned}
&\mathbb{E}\, L(S^*) - \mathbb{E}\, L(S^* \cup \{j\}) \\
&= -(2 \sum_{i \in S^*} p_i - 2D + p_j) p_j - p_j(1 - p_j) \\
&= -(2 \sum_{i \in S^*} p_i - 2D + 1) p_j \leq 0
\end{aligned}$$

Since $p_j > 0$, we must have

$$\sum_{i \in S^*} p_i \geq D - 1/2$$

□

Corollary 2. *When $D < 1/2$, the empty set is optimal.*

Proof. Suppose there exists a non-empty optimal subset $S \neq \emptyset$. Then

$$\sum_{i \in S^* - \{j\}} p_i \geq 0 > D - 1/2, \quad \forall\ j \in S^*$$

which results in a contradiction by Lemma 4. □

Corollary 3. *When $\sum_{i=1}^n p_i \leq D - 1/2$, the optimal subset is $[n]$.*

Proof. Suppose there exists an optimal subset $S \neq [n]$. Then

$$\sum_{i \in S^*} p_i < \sum_{i=1}^n p_i \leq D - 1/2$$

which results in a contradiction by Lemma 4. □

Next, we are going to show that there must exist an optimal subset containing all elements with highest mean values. This is done by contradiction.

Lemma 5. *When $D \geq 1/2$ and $\sum_{i=1}^n p_i > D - 1/2$, there must exist an optimal subset S^* whose elements' mean values are $q_1 \geq \cdots \geq q_m$, such that for any $p_i > q_m$, we have $i \in S^*$.*

Proof. Let's prove by construction and contradiction. Consider S^*, assume there exists $p_i > q_m$ but $i \notin S^*$. In the following, we will ignore other random variables outside $S^* \cup \{i\}$ because they are irrelevant. Now, we rank the mean value $\{q_1, \ldots, p_i, \ldots, q_m, \}$ and assume that p_i is the jth largest element here. To simplify the notation, we will call the newly ranked mean value set as

$$p_1 \geq \cdots \geq p_{j-1} \geq p_j \geq p_{j+1} \geq \cdots \geq p_{m+1}$$

The mean values of random variables in S^* are $\{p_1, \ldots, p_{j-1}, p_{j+1}, \ldots, p_{m+1}\}$ (used to be called as $\{q_1, \ldots, q_m\}$) and the injected element (used to be denoted as p_i) now is called p_j. Under this simpler notation, we proceed to construct a subset of top k arms with some k whose expected loss is no more than the optimal expected loss.

Construct a subset A in the following way. Pick the smallest k such that

$$\sum_{i=1}^{k} p_i > D - 1/2$$
$$\sum_{i=1}^{k-1} p_i \leq D - 1/2$$

[4] One way to think about this is that we only consider subset S such that $p_i > 0$ for $i \in S$.

Then let $A = \{1, \ldots, k\}$. It is easy to see that $k \geq j$. (Since S^* is optimal, by Lemma 4, excluding any element will go below $D - 1/2$, so k must include the newcomer j to be beyond $D - 1/2$.)

We claim that $\mathbb{E}\, L(A) \leq \mathbb{E}\, L(S^*)$. Since $\mathbb{E}\, L(S^*)$ is optimal, we must have $\mathbb{E}\, L(A) = \mathbb{E}\, L(S^*)$ and A is also optimal. Then, we can construct a new subset A_1 with the same rule above. Since there are only finite elements, we can always end up with an optimal set $\hat{A}$ which includes variables with the highest mean values. Then the proof is done.

The only remaining part is to prove $\mathbb{E}\, L(A) \leq \mathbb{E}\, L(S^*)$. Denote

$$\sum_{i=1}^{k} p_i = D - 1/2 + \delta_x$$
$$\sum_{i \neq j,\, i=1}^{m+1} p_i = D - 1/2 + \delta_y$$

By construction of k, $\delta_x \in (0, p_k]$. By Lemma 4, $\delta_y \in [0, p_{m+1}]$. So we have

$$\sum_{i=1}^{k} p_i - \sum_{i \neq j,\, i=1}^{m+1} p_i = p_j - \sum_{i=k+1}^{m+1} p_i = \delta_x - \delta_y$$

Now let's do some simple algebra and try to prove $\mathbb{E}\, L(A) - \mathbb{E}\, L(S^*) \leq 0$. Basically, we are trying to write $\mathbb{E}\, L(A) - \mathbb{E}\, L(S^*)$ with δ_x δ_y defined above and then bound it using bounds of δ_x δ_y .

$$\mathbb{E}\, L(A) - \mathbb{E}\, L(S^*) = (\sum_{i=1}^{k} p_i - D)^2 + \tag{11}$$
$$\sum_{i=1}^{k} p_i(1-p_i) - (\sum_{i \neq j,\, i=1}^{m+1} p_i - D)^2 - \sum_{i \neq j,\, i=1}^{m+1} p_i(1-p_i) \tag{12}$$
$$= (\sum_{i=1}^{k} p_i - D + \sum_{i \neq j,\, i=1}^{m+1} p_i - D) \tag{13}$$
$$(\sum_{i=1}^{k} p_i - D - \sum_{i \neq j,\, i=1}^{m+1} p_i + D) \tag{14}$$
$$+ p_j(1-p_j) - \sum_{i=k+1}^{m+1} p_i(1-p_i) \tag{15}$$
$$= (\delta_x + \delta_y - 1)(\delta_x - \delta_y) + p_j - \sum_{i=k+1}^{m+1} p_i + \sum_{i=k+1}^{m+1} p_i^2 - p_j^2 \tag{16}$$
$$= (\delta_x + \delta_y - 1)(\delta_x - \delta_y) + \delta_x - \delta_y + \sum_{i=k+1}^{m+1} p_i^2 - p_j^2 \tag{17}$$
$$= (\delta_x + \delta_y)(\delta_x - \delta_y) + \sum_{i=k+1}^{m+1} p_i^2 - p_j^2 \tag{18}$$
$$= (\delta_x + \delta_y)(\delta_x - \delta_y) + \sum_{i=k+1}^{m+1} p_i^2 - (\sum_{i=k+1}^{m+1} p_i + \delta_x - \delta_y)^2 \tag{19}$$
$$= (\delta_x + \delta_y)(\delta_x - \delta_y) + \sum_{i=k+1}^{m+1} p_i^2 \tag{20}$$
$$- \left[(\sum_{i=k+1}^{m+1} p_i)^2 + (\delta_x - \delta_y)^2 + 2(\delta_x - \delta_y) \sum_{i=k+1}^{m+1} p_i \right] \tag{21}$$
$$= (\delta_x - \delta_y)(\delta_x + \delta_y - \delta_x + \delta_y - 2 \sum_{i=k+1}^{m+1} p_i) \tag{22}$$
$$+ \sum_{i=k+1}^{m+1} p_i^2 - (\sum_{i=k+1}^{m+1} p_i)^2 \tag{23}$$
$$= (\delta_x - \delta_y)(2\delta_y - 2 \sum_{i=k+1}^{m+1} p_i) + \sum_{i=k+1}^{m+1} p_i^2 - (\sum_{i=k+1}^{m+1} p_i)^2 \tag{24}$$

Now, we first notice that $\delta_y \leq p_{m+1} \leq \sum_{i=k+1}^{m+1} p_i$, so $(2\delta_y - 2\sum_{i=k+1}^{m+1} p_i) \leq 0$.

Also notice that

$$\sum_{i=k+1}^{m+1} p_i^2 - (\sum_{i=k+1}^{m+1} p_i)^2 \leq 0$$

since $p_i \geq 0$.

Case 1: $\delta_x \geq \delta_y$. In this case, (24)≤ 0 is straightforward.

Case 2: $\delta_x < \delta_y$. In this case, $p_{m+1} < p_j < \sum_{i=k+1}^{m+1} p_i$. So we must have $m - k \geq 1$. Since $(2\delta_y - 2\sum_{i=k+1}^{m+1} p_i) \leq 0$, we can decrease δ_x to 0

$$(24) \leq -\delta_y(2\delta_y - 2 \sum_{i=k+1}^{m+1} p_i) + \sum_{i=k+1}^{m+1} p_i^2 - (\sum_{i=k+1}^{m+1} p_i)^2$$
$$= RHS$$

RHS is a quadratic function with respect to δ_y and it is increasing in the region $[0, \frac{\sum_{i=k+1}^{m+1} p_i}{2}]$. Since $m - k \geq 1$, we have

$$\frac{\sum_{i=k+1}^{m+1} p_i}{2} \geq (p_{m+1} + p_m)/2 \geq p_{m+1} \geq \delta_y$$

So the highest possible value is reached when $\delta_y = p_{m+1}$. Plugging this in RHS, we have

$$RHS$$
$$\leq -p_{m+1}(2p_{m+1} - 2 \sum_{i=k+1}^{m+1} p_i) + \sum_{i=k+1}^{m+1} p_i^2 - (\sum_{i=k+1}^{m+1} p_i)^2$$
$$= -2p_{m+1}^2 + 2(\sum_{i=k+1}^{m+1} p_i)p_{m+1} + \sum_{i=k+1}^{m+1} p_i^2 - (\sum_{i=k+1}^{m+1} p_i)^2$$
$$= (\sum_{i=k+1}^{m+1} p_i)(p_{m+1} - \sum_{i=k+1}^{m} p_i) - p_{m+1}^2 + \sum_{i=k}^{m} p_i^2$$
$$= (p_{m+1} + \sum_{i=k}^{m} p_i)(p_{m+1} - \sum_{i=k+1}^{m} p_i) - p_{m+1}^2 + \sum_{i=k}^{m} p_i^2$$

$$= p_{m+1}^2 - (\sum_{i=k}^{m} p_i)^2 - p_{m+1}^2 + \sum_{i=k}^{m} p_i^2$$
$$= \sum_{i=k}^{m} p_i^2 - (\sum_{i=k}^{m} p_i)^2 \leq 0 \qquad \text{Since } p_i \geq 0$$

Thus we have shown that

$$\mathbb{E}\, L(A) - \mathbb{E}\, L(S^*) \leq 0$$

□

Now, given Lemma 4 and Lemma 5, we can prove Theorem 1.

Proof of theorem 1: When $D < 1/2$ or $\sum_{i=1}^{n} p_i \leq D - 1/2$, see Corollary 2 and Corollary 3.

When $D \geq 1/2$ and $\sum_{i=1}^{n} p_i > D - 1/2$, by Lemma 5, there exists an optimal subset $S^* = \{1, \ldots, k\}$ for some k, i.e. containing the first several arms with largest mean values. Since S^* is optimal, we must have, by Lemma 4, that

$$\sum_{i=1}^{k} p_i \geq D - 1/2$$
$$\sum_{i=1}^{k-1} p_i \leq D - 1/2$$

- If $\sum_{i=1}^{k} p_i > D - 1/2$, then S^* is the output of Algorithm 1, so the output of Algorithm 1 is optimal.
- If $\sum_{i=1}^{k} p_i = D - 1/2$, then it is easy to show that $\mathbb{E}\, L(\{1, \ldots, k\}) = \mathbb{E}\, L(\{1, \ldots, k+1\})$. So $\{1, \ldots, k+1\}$ is also optimal. The output of Algorithm 1 is $\{1, \ldots, k+1\}$. So the output of Algorithm 1 is still optimal.

□

Corollary 4. *If $\sum_{i=1}^{k-1} p_i = D - 1/2$, then the subset $\{1, \ldots, k-1\}$ and $\{1, \ldots, k\}$ are both optimal.*

Corollary 5. *If $D \leq 1/2$, the empty set is optimal.*

B. Proof of Lemma 3 with Assumption (A1) and (A2) and proof of Proposition 1

In order to prove Lemma 3, we first introduce some properties of the algorithm-selected subsets according to the algorithm rules in Lemma 6, then provide the formal proof.

Lemma 6 (Properties of CUCB-Avg's Selection)**.** *Given any realization of $\mathcal{F}_{t-1}$ (now every thing is deterministic), if $S_t = s$ for some deterministic subset $s \subseteq [n]$, we have the following:*

- *$\sum_{i \in s} \bar{p}_i > D - 1/2$ or $s = [n]$*
- *Define $i_{min} = \arg\min_{i \in s} U_i(t)$, then $\sum_{i \in s - \{i_{min}\}} \bar{p}_i \leq D - 1/2$*
- *For any $j \in s$ and any $i \in [n]$ such that $U_i(t) > U_j(t)$, $i \in s$.*

Proof. The proof becomes trivial when we fix $\mathcal{F}_{t-1}$. □

Proof of Lemma 3 given Assumption (A1) and (A2):

We will prove Lemma 3 using the explicit expression given in Proposition 1. The proof of Proposition 1 follows naturally.

Step 1: Given $\bar{E}_t, \bar{B}_t(\epsilon_0)$, either $S_t \subseteq S^*$ or $S^* \subseteq S_t$: We prove this by contradiction. Suppose there is a realization of $\mathcal{F}_{t-1}$ such that $\bar{E}_t$ and $\bar{B}_t(\epsilon_0)$ hold, but $S_t \not\subseteq S^*$ and $S^* \not\subseteq S_t$. Fix this realization of $\mathcal{F}_{t-1}$, then there exists $j \in S_t - S^*$, $i \in S^* - S_t$. We will show that $U_i(t) > U_j(t)$, then by Lemma 6, we have $i \in S_t$ which leads to contradiction.

$$\begin{aligned} U_i(t) &= \bar{p}_i(t-1) + \sqrt{\frac{\alpha \log t}{2T_i(t-1)}} \\ &> p_i \\ &\geq p_{\sigma(k)} \qquad (\text{by } i \in S^*,\ \sigma(k) \text{ is the smallest one}) \end{aligned}$$

$$\begin{aligned} U_j(t) &= \bar{p}_j(t-1) + \sqrt{\frac{\alpha \log t}{2T_j(t-1)}} \qquad (\text{by definition}) \\ &< p_j + 2\sqrt{\frac{\alpha \log t}{2T_j(t-1)}} \qquad (\text{by definition of } \bar{E}_t) \\ &< p_j + 2\epsilon_0 \qquad (\text{by } j \in S_t, \text{ and definition of } \bar{B}_t(\epsilon_0)) \\ &\leq p_{\sigma(k+1)} + 2\epsilon_0 \qquad (\text{by } j \notin S^*,\ \sigma(k+1) \text{ is the largest}) \\ &\leq p_{\sigma(k+1)} + 2\frac{p_{\sigma(k)} - p_{\sigma(k+1)}}{2} \qquad (\text{by definition } \epsilon_0 \leq \frac{\Delta_k}{2}) \\ &\leq p_{\sigma(k)} < U_i(t) \end{aligned}$$

Step 2: Given $\bar{E}_t, \bar{B}_t(\epsilon_0)$, $S_t \subsetneq S^*$ is impossible: We prove this by contradiction. Suppose $S_t \subsetneq S^*$, then $S_t \neq [n]$. We can show that

$$\begin{aligned} &\sum_{i \in S_t} \bar{p}_i(t-1) \\ &< \sum_{i \in S_t} \left(p_i + \sqrt{\frac{\alpha \log t}{2T_i(t-1)}} \right) \qquad \text{by } \bar{E}_t \\ &< \sum_{i \in S_t} (p_i + \epsilon_0) \qquad \text{by } \bar{B}_t(\epsilon_0) \\ &\leq \sum_{i=1}^{k-1} \left(p_{\sigma(i)} + \epsilon_0 \right) \qquad S_t \subsetneq S^* \text{ but } S_t \neq S^* \\ &\leq D - 1/2 - \delta_2 + (k-1)\epsilon_0 \qquad \text{by definition of } \delta_2 \\ &\leq D - 1/2 \qquad \text{by definition of } \epsilon_0,\ \epsilon_0 \leq \frac{\delta_2}{k} \end{aligned}$$

However, by Lemma 6, $\sum_{i \in S_t} \bar{p}_i(t-1) > D - 1/2$, which leads to a contradiction.

Step 3: Given $\bar{E}_t, \bar{B}_t(\epsilon_0)$, $S^* \subsetneq S_t$ is impossible: We prove this by contradiction. Suppose $S^* \subsetneq S_t$, so $S^* \neq [n]$, so $\sum_{i=1}^{n} p_i > D - 1/2$. We will show that $i_{min} = \arg\min_{i \in S_t} \{U_i(t)\} \in S_t - S^*$, and $\sum_{i \in S_t - \{i_{min}\}} \bar{p}_i(t-1) > D - 1/2$. Then by Lemma 6, we have a contradiction.

Now first we show that $i_{min} = \arg\min_{i \in S_t} \{U_i(t)\} \in S_t - S^*$. We only need to show that for any $i \in S^*$, and $j \in S_t - S^*$, $U_i(t) > U_j(t)$. The reason is the following.

$$\begin{aligned} U_i(t) &> p_i \geq p_{\sigma(k)} \qquad (\text{by } \bar{E}_t) \\ U_j(t) &< p_j + 2\epsilon_0 \qquad (\text{by } \bar{E}_t \text{ and } \bar{B}_t(\epsilon_0)) \\ &\leq p_{\sigma(k+1)} + 2\frac{\Delta_k}{2} \quad (\text{by } j \notin S^* \text{ and def of } \epsilon_0) \end{aligned}$$

$$\leq p_{\sigma(k)} < U_i$$

Then we show $\sum_{i\in S_t-\{i_{min}\}} \bar{p}_i(t-1) > D - 1/2$ by

$$\begin{aligned}
&\sum_{i\in S_t-\{i_{min}\}} \bar{p}_i(t-1) \\
&\geq \sum_{i\in S^*} \bar{p}_i(t-1) && (i_{min} \in S_t - S^*,\ S_t \supsetneq S^*) \\
&> \sum_{i\in S^*} (p_i - \epsilon_0) && (\text{by } \bar{E}_t) \\
&= D - 1/2 + \delta_1 - k\epsilon_0 \\
&\geq D - 1/2 && (\text{def of } \epsilon_0)
\end{aligned}$$

Step 4: $\mathbb{E}\, R_t(S_t) I_{\{\bar{E}_t, \bar{B}_t(\epsilon_0)\}} = 0$**.** By the three steps above, we have $S_t = S^*$, then it is straightforward that

$$\begin{aligned}
&\mathbb{E}\, R_t(S_t) I_{\{\bar{E}_t, \bar{B}_t(\epsilon_0)\}} = \mathbb{E}\, R_t(S_t) I_{\{\bar{E}_t, \bar{B}_t(\epsilon_0), S_t = S^*\}} \\
&= \mathbb{E}\, R_t(S^*) I_{\{\bar{E}_t, \bar{B}_t(\epsilon_0), S_t = S^*\}} = 0
\end{aligned}$$

□

C. Proof of Lemma 3 without Assumption (A1) and (A2)

We will first give an explicit expression of ϵ_0, then we will show $\mathbb{E}\, R_t(S_t) I_{\{\bar{E}_t, \bar{B}_t(\epsilon_0)\}} = 0$ given the new ϵ_0.

Without loss of generality, we will consider $D > 1/2$ due to Corollary 5.

Now we present the expression of ϵ_0 in the general case.

Definition 1. *The ϵ_0 in Lemma 3 can be determined by*

$$\epsilon_0 = \min(\frac{\delta_1}{l_1}, \frac{\delta_2}{l_2}, \frac{\Delta_{k_1}}{2}, \frac{\Delta_{k_2}}{2})$$

In the following, we define each parameter respectively.

***Definition of** l_1**:** *Let l_1 be the smallest number of the top arms such that the total weight is higher than $D - 1/2$, or n if the total weight of all arms is no more than $D - 1/2$*

$$l_1 = \begin{cases} \min G_1 & \text{if } \sum_{i=1}^n p_i > D - 1/2 \\ n & \text{if } \sum_{i=1}^n p_i \leq D - 1/2 \end{cases}$$

where $G_1 = \{1 \leq k \leq n |\ \sum_{i=1}^k p_{\sigma(i)} > D - 1/2\}$. Note that $1 \leq l_1 \leq n$ when $D > 1/2$. The offline algorithm output is $\phi(p, D) = \{\sigma(1), \ldots, \sigma(l_1)\}$.

By algorithm design, $\{\sigma(1), \ldots, \sigma(l_1)\}$ is one of the possible output of the offline optimization algorithm (there might be other outputs due to the random tie-breaking rule).

***Definition of** δ_1**:** *Let δ_1 be the total expected contribution by $\sigma(1), \ldots, \sigma(l_1)$ minus $D - 1/2$. But $l_1 = n$ is a special case. When $l_1 = n$, let $\delta_1 = n$, so that $\frac{\delta_1}{l_1} = 1$ is large enough to not affect the definition of ϵ_0.*

$$\delta_1 = \begin{cases} \sum_{i=1}^{l_1} p_{\sigma(i)} - (D - 1/2) & \text{if } l_1 < n \\ n & \text{if } l_1 = n \end{cases}$$

***Definition of** l_2**:** *Let l_2 be the largest number of the top arms such that the total weight is less than $D - 1/2$.*

$$l_2 = \max\{0 \leq k \leq n |\ \sum_{i=1}^k p_{\sigma(i)} < D - 1/2\}$$

Note that $0 \leq l_2 \leq n$, and $l_1 \geq l_2$. In addition, $l_1 = l_2$ if and only if $l_1 = l_2 = n$.

***Definition of** δ_2**:** *Let δ_2 be $D - 1/2$ minus the total expected contribution by $\sigma(1), \ldots, \sigma(l_2)$. When $l_2 = 0$, let $\delta_2 = 1$, then $\frac{\delta_2}{l_2} = +\infty$, which is large enough to not affect the definition of ϵ_0*

$$\delta_2 = \begin{cases} (D - 1/2) - \sum_{i=1}^{l_2} p_{\sigma(i)} & \text{if } l_2 \geq 1 \\ 1 & \text{if } l_2 = 0 \end{cases}$$

***Definition of** Δ_i**:** *Define the gap between two consecutive arms in the non-increasing order by Δ_i. When $i = 0, n$, let it be 2 which is large enough to keep the ϵ_0 unaffected.*

$$\Delta_i = \begin{cases} p_{\sigma(i)} - p_{\sigma(i+1)} & \text{if } 1 \leq i \leq n-1 \\ 2 & i = 0, n \end{cases}$$

***Definition of** k_1**:** *Let k_1 be the arm with the largest index under the non-increasing order such that the parameter $p_{\sigma(k_1)}$ is the second smallest among $p_{\sigma(1)}, \ldots, p_{\sigma(l_1)}$. When all arms are in a tie, then $k_1 = 0$.*

$$k_1 = \max\{0 \leq i \leq l_1 - 1 |\ \Delta_i > 0\}$$

***Definition of** k_2**:** *Let k_2 be the arm with the largest index under the non-increasing order that can be possibly selected by the offline optimization algorithm. When all arms are in a tie, then $k_2 = n$.*

$$k_2 = \min\{l_1 \leq i \leq n |\ \Delta_i > 0\}$$

In the following, we will first prove some supportive lemmas, then provide a proof of Lemma 3.

Lemma 7. *For any $\epsilon > 0$ satisfying $\epsilon \leq \Delta_k/2$ for some k with $0 \leq k \leq n$, given $\bar{E}_t$ and $\bar{B}_t(\epsilon)$, then we have either $S_t \subseteq \{\sigma(1), \ldots, \sigma(k)\}$ or $\{\sigma(1), \ldots, \sigma(k)\} \subseteq S_t$.*

Proof. When $k = 0, n$, it is trivially true.

When $1 \leq k \leq n-1$. Suppose there is a realization of $\mathcal{F}_{t-1}$ such that $\bar{E}_t$ and $\bar{B}_t(\epsilon)$ hold, but $S_t \not\subseteq \{\sigma(1), \ldots, \sigma(k)\}$ and $\{\sigma(1), \ldots, \sigma(k)\} \not\subseteq S_t$. Fix this realization of $\mathcal{F}_{t-1}$, then there exists $j \in S_t - \{\sigma(1), \ldots, \sigma(k)\}$, $i \in \{\sigma(1), \ldots, \sigma(k)\} - S_t$. We will show that $U_i(t) > U_j(t)$, then by Lemma 6, we have $i \in S_t$ which leads to contradiction.

$$U_i(t) = \bar{p}_i(t-1) + \sqrt{\frac{\alpha \log t}{2T_i(t-1)}} > p_i \geq p_{\sigma(k)}$$

$$\begin{aligned}
U_j(t) &= \bar{p}_j(t-1) + \sqrt{\frac{\alpha \log t}{2T_j(t-1)}} \\
&< p_j + 2\sqrt{\frac{\alpha \log t}{2T_j(t-1)}} < p_j + 2\epsilon \\
&\leq p_{\sigma(k+1)} + 2\epsilon \leq p_{\sigma(k+1)} + 2\frac{p_{\sigma(k)} - p_{\sigma(k+1)}}{2} \\
&\leq p_{\sigma(k)} < U_i(t)
\end{aligned}$$

□

Corollary 6. *Given $\bar{E}_t$ and $\bar{B}_t(\epsilon_0)$, then we have either $S_t \subsetneqq \{\sigma(1),\ldots,\sigma(k_1)\}$ or $\{\sigma(1),\ldots,\sigma(k_1)\} \subseteq S_t \subseteq \{\sigma(1),\ldots,\sigma(k_2)\}$, or $\{\sigma(1),\ldots,\sigma(k_2)\} \subsetneqq S_t$.*

Proof. It is easy to see that $k_1 < k_2$ by definition, and by $\epsilon_0 \leq \Delta_{k_1}/2$, $\epsilon_0 \leq \Delta_{k_1}/2$, then by Lemma 7, it is straightforward. □

Finally, we are ready to prove Lemma 3.

Proof of Lemma 3: The major part of the proof is to show that if $\bar{E}_t, \bar{B}_t(\epsilon_0)$ happen, then S_t must be optimal. Then, given that S_t is optimal, it is easy to prove zero regret at time t.

Now, let's first prove $S_t \in \mathbb{S}_t^*$ given $\bar{E}_t, \bar{B}_t(\epsilon_0)$, where $\mathbb{S}^*$ denotes the set of all possible optimal subsets.

Step 1: prove $\bar{E}_t$ and $\bar{B}(\epsilon_0) \Rightarrow S_t \in \mathbb{S}^*$. We will list all possible scenarios, and prove that in each scenario, when $\bar{E}_t$ and $\bar{B}(\epsilon_0)$ hold, we have $S_t \in \mathbb{S}^*$.

Scenario 1: When $l_1 > l_2 + 2$, then we have $\sum_{i=1}^{l_2+1} p_{\sigma(i)} = \sum_{i=1}^{l_2+2} p_{\sigma(i)} = D - 1/2$, so $p_{\sigma(l_2+2)} = 0$. Accordingly, $p_{\sigma(i)} = 0$ for all $i \geq l_2 + 2$. This leads to $l_1 = n$. By Corollary 4, one optimal subset is $\{\sigma(1),\ldots,\sigma(l_2+1)\}$. Besides, other optimal subsets could also be $\{\sigma(1),\ldots,\sigma(l_2+1)\}$ plus some arms with zero probabilities. In addition, it is easy to see that $k_2 = n$ and $k_1 = l_2 + 1$ by definition.

By Corollary 6, we have either $S_t \subsetneqq \{\sigma(1),\ldots,\sigma(l_2+1)\}$ or $\{\sigma(1),\ldots,\sigma(l_2+1)\} \subseteq S_t \subseteq \{\sigma(1),\ldots,\sigma(n)\}$. Since the second possible case guarantees $S_t \in \mathbb{S}^*$, we only need to show that $S_t \subsetneqq \{\sigma(1),\ldots,\sigma(l_2+1)\}$ is impossible. This is done by the following fact:

$$\sum_{i\in S_t} \bar{p}_i(t-1) \leq \sum_{i=1}^{l_2} \bar{p}_{\sigma(i)}(t-1) \leq \sum_{i=1}^{l_2} p_{\sigma(i)} + \epsilon_0 \leq \sum_{i=1}^{l_2} p_{\sigma(i)} + l_2\epsilon_0 < D - 1/2$$

Scenario 2: When $l_1 = l_2 + 1$, then we have $k_1 \leq l_2$. Let's first discuss the optimal subset $S^* = \phi(p, D)$ which could be $\{\sigma(1),\ldots,\sigma(l_1)\}$. Clearly, $k_1 < |S^*| \leq k_2$. In addition, $p_{\sigma(k_1+1)} = \cdots = p_{\sigma(k_2)}$. Now, we know S^* consists of $\{\sigma(1),\ldots,\sigma(k_1)\}$ together with $|S^*| - k_1$ arms that are in a tie.

In the following, we will show by contradiction that 1) $S_t \supsetneqq \{\sigma(1),\ldots,\sigma(k_1)\}$, 2) $S_t \subseteq \{\sigma(1),\ldots,\sigma(k_2)\}$, and 3) S_t has exactly $|S^*| - k_1$ arms with parameter $p_{\sigma(k_1+1)}$.

First, we show that $S_t \subseteq \{\sigma(1),\ldots,\sigma(k_1)\}$ is impossible by showing the sum is less than $D - 1/2$

$$\sum_{i\in S_t} \bar{p}_i(t-1) \leq \sum_{i=1}^{k_1} \bar{p}_{\sigma(i)}(t-1) \leq \sum_{i=1}^{l_2} p_{\sigma(i)} + l_2\epsilon_0 < D - 1/2$$

Secondly, we show that $S_t \subseteq \{\sigma(1),\ldots,\sigma(k_2)\}$. Suppose $S_t \supsetneqq \{\sigma(1),\ldots,\sigma(k_2)\}$. First we show that $i_{min} = \arg\min_{i\in S_t}\{U_i(t)\} \in S_t - \{\sigma(1),\ldots,\sigma(k_2)\}$. We only need to show that for any $i \in \{\sigma(1),\ldots,\sigma(k_2)\}$, and $j \in S_t - \{\sigma(1),\ldots,\sigma(k_2)\}$, $U_i(t) > U_j(t)$. The reason is the following.

$$\begin{aligned} U_i(t) &> p_i \geq p_{\sigma(k_2)} && \text{by } \bar{E}_t \\ U_j(t) &< p_j + 2\epsilon_0 && \text{by } \bar{E}_t \text{ and } \bar{B}_t(\epsilon_0) \\ &\leq p_{\sigma(k_2+1)} + 2\frac{\Delta_{k_2}}{2} \\ &\leq p_{\sigma(k_2)} < U_i(t) \end{aligned}$$

Then we show $\sum_{i\in S_t - \{i_{min}\}} \bar{p}_i(t-1) > D - 1/2$, which is impossible if algorithm selects S_t.

$$\begin{aligned} \sum_{i\in S_t-\{i_{min}\}} \bar{p}_i(t-1) &\geq \sum_{i=1}^{k_2} \bar{p}_{\sigma(i)}(t-1) \\ &\geq \sum_{i=1}^{l_1} \bar{p}_{\sigma(i)}(t-1) > \sum_{i=1}^{l_1} p_{\sigma(i)} - l_1\epsilon_0 \\ &= D - 1/2 + \delta_1 - l_1\epsilon_0 \geq D - 1/2 \end{aligned}$$

Thirdly, we need to deal with the ties. It is straightforward to see that, if we select more or less than $|S^*| - k_1$ arms that are in a tie, the sum of parameters is either more than $D - 1/2$ excluding i_{min} or less than $D - 1/2$, using the same trick above.

Scenario 3: When $l_1 = l_2 + 2$, then $\sum_{i=1}^{l_1-1} p_{\sigma(i)} = D - 1/2$, and $k_1 \leq l_2 + 1$. Let's first discuss the optimal subset: note that both $\phi(p, D) = \{\sigma(1),\ldots,\sigma(l_1)\}$ and $\{\sigma(1),\ldots,\sigma(l_1-1)\}$ are optimal. In addition, we have ties $p_{\sigma(k_1+1)} = \cdots = p_{\sigma(k_2)}$. Therefore, an optimal subsets consists of $\{\sigma(1),\ldots,\sigma(k_1)\}$ together with either $l_1 - k_1$ or $l_1 - 1 - k_1$ arms in a tie.

In the following, we will show by contradiction that 1) $S_t \supsetneqq \{\sigma(1),\ldots,\sigma(s)\}$ for $s = \min(k_1, l_2)$, 2) $S_t \subseteq \{\sigma(1),\ldots,\sigma(k_2)\}$, and 3) S_t has either $l_1 - k_1$ or $l_1 - 1 - k_1$ arms with parameter $p_{\sigma(k_1+1)}$.

Firstly, we can easily show that $S_t \subseteq \{\sigma(1),\ldots,\sigma(s)\} \neq [n]$ is impossible for $s = \min(k_1, l_2)$. This is done by showing the sum is less than $D - 1/2$

$$\sum_{i=1}^{s} \bar{p}_{\sigma(i)} \leq \sum_{i=1}^{l_2} p_{\sigma(i)} + l_2\epsilon_0 < D - 1/2$$

Secondly, we can also show that $S_t \supsetneqq \{\sigma(1),\ldots,\sigma(k_2)\}$ is impossible using the same proof in Scenario 2.

Thirdly, we need to deal with the ties. It is straightforward to see that, if we select more or less than the needed tieing arms, the sum of weights is either more than $D - 1/2$ excluding i_{min} or less than $D - 1/2$, using the same trick above.

Step 2: prove $\mathbb{E}\, R_t(S_t) I_{\{\bar{E}_t, \bar{B}_t(\epsilon_0)\}} = 0$. Notice that $\bar{E}_t$ and $\bar{B}_t(\epsilon_0)$ are determined by $\mathcal{F}_{t-1}$. Consider $\mathbb{S}^*$ to be the set of all optimal subsets,

$$\begin{aligned} &\mathbb{E}\, I_{\{\bar{E}_t, \bar{B}_t(\epsilon_0),\ S_t\in\mathbb{S}^*\}} R_t(S_t) \\ &= \sum_{s\in\mathbb{S}^*} \mathbb{E}\, I_{\{\bar{E}_t, \bar{B}_t(\epsilon_0),\ S_t=s\}} R_t(S_t) \end{aligned}$$

$$= \sum_{s\in\mathbb{S}^*} \mathbb{P}(\bar{E}_t, \bar{B}_t(\epsilon_0),\ S_t = s)\,\mathbb{E}\,R_t(s) = 0$$

where the second inequality is because $R_t(s)$ and $I_{\{\bar{E}_t,\bar{B}_t(\epsilon_0),\ S_t=s\}}$ are independent and independence leads to uncorrelation.

□

D. Proof of Theorem 4

Step 1: Preparation. We note that the single-step regret is bounded by $\bar{M}$:

$$R_t(S_t) \le \max(D_t^2, (n-D_t)^2) \le \max(\bar{D}^2, n^2) = \bar{M}$$

Step 2: Divide T time steps into four parts and bound the regret of each part.

Next, we divide the pseudo-regret in T time steps into four parts by E_t and $B_t(\epsilon)$ for any $0 < \epsilon \le \beta$, and bound each part separately.

(1) Initialization: $\mathbb{E}\,R_1(S_1) \le \bar{M}$.

(2) Time steps when E_t happens: Notice that Lemma 1 still holds in time-varying case if replacing M with $\bar{M}$, so the second part is bounded by

$$\mathbb{E}\sum_{t=2}^{T} I_{E_t} R_t(S_t) \le \frac{2\bar{M}n}{\alpha-2}$$

(3) Time steps when $\bar{E}_t$, $B_t(\epsilon)$ happen: Notice that Lemma 2 still holds in time-varying case, thus

$$\mathbb{E}\sum_{t=2}^{T} R_t(S_t) I_{\{\bar{E}_t,B_t(\epsilon)\}} \le \frac{\alpha\bar{M}n\log T}{2\epsilon^2}$$

(4) Time steps when $\bar{E}_t$, $\bar{B}_t(\epsilon)$ happen: We will show in Lemma 8 that $\mathbb{E}\,R_t(S_t) I_{\{\bar{E}_t,\bar{B}_t(\epsilon)\}} \le g(\epsilon)$, then

$$\mathbb{E}\sum_{t=2}^{T} R_t(S_t) I_{\{\bar{E}_t,\bar{B}_t(\epsilon)\}} \le T g(\epsilon)$$

where $g(\epsilon) = ((2n+2)\epsilon + 2)(2n+2)\epsilon$.

Step 3: Combine the regret bounds in four parts. Combining four bounds above, we have for any $\alpha > 2$, the regret of CUCB-Avg is upper bounded by,

$$Regret(T) \le \bar{M} + \frac{2\bar{M}n}{\alpha-2} + Tg(\epsilon) + \frac{\alpha\bar{M}n\log T}{2\epsilon^2}$$

Step 4: Pick $\epsilon \sim O(\left(\frac{\log T}{T}\right)^{\frac{1}{3}})$.

Let $\epsilon = \min(\beta, \left(\frac{\bar{M}\log T}{T}\right)^{\frac{1}{3}})$, and notice that $g(\epsilon) \le 3n\epsilon(3n\epsilon+2) = 9n^2\epsilon^2 + 6n\epsilon$ for $n \ge 1$. Then we can get the regret bound in Theorem 4.

□

Lemma 8. *For any $0 \le \epsilon \le \beta$, $\mathbb{E}\,R_t(S_t) I_{\{\bar{E}_t,\bar{B}_t(\epsilon)\}} \le g(\epsilon)$, where $g(\epsilon) = ((2n+2)\epsilon+2)(2n+2)\epsilon$; the expectation is taken with respect to random contribution $X_{t,i}$ and random online algorithm output S_t.*

Proof. The proof is rather technical, and detained to Appendix E and F. □

E. Proof of Lemma 8 with Assumption (A1)

Before the proof, we introduce some notations. Define the offline optimization output as $S_t^* = \phi(p, D_t) = \{\sigma(1), \ldots, \sigma(r_t)\}$, where $r_t = |\phi(p, D_t)|$. In addition, define $\delta_{t,1}$ and $\delta_{t,2}$ following the definition (7) in static case

$$\begin{aligned} \sum_{i=1}^{r_t} p_{\sigma(i)} &= D_t - 1/2 + \delta_{t,1}, \\ \sum_{i=1}^{r_t-1} p_{\sigma(i)} &= D_t - 1/2 - \delta_{t,2}. \end{aligned} \tag{25}$$

Note that $\delta_{t,2} \ge 0$, and when $r_t < n$, $\delta_{t,1} > 0$.

Lastly, we note that we have been using S_t as the random subset selected by the online algorithm. In the following, we are going to discuss specific subset, which is deterministic. To avoid confusion, we will use lower-case $s \subseteq [n]$ to denote a deterministic subset. There is one exception: the optimal subset S_t^* is deterministic but its notation uses upper-case letter to be aligned with the previous notation.

Proof of Lemma 8 with Assumption (A1):

Let $\mathcal{L}_t$ denote the set containing all possible subsets that S_t might be given $\bar{E}_t$ and $\bar{B}_t(\epsilon)$.

Step 1: Convert the problem to proving $\mathbb{E}\,R_t(s) \le g(\epsilon)$ for all $s \in \mathcal{L}_t$: Notice that

$$\begin{aligned} &\mathbb{E}\,R_t(S_t) I_{\{\bar{E}_t,\bar{B}_t(\epsilon)\}} = \mathbb{E}\,R_t(S_t) I_{\{\bar{E}_t,\bar{B}_t(\epsilon),S_t\in\mathcal{L}_t\}} \\ &= \sum_{s\in\mathcal{L}_t} \mathbb{E}\,R_t(s) I_{\{\bar{E}_t,\bar{B}_t(\epsilon),S_t=s\}} \\ &= \sum_{s\in\mathcal{L}_t} \mathbb{P}(\bar{E}_t, \bar{B}_t(\epsilon), S_t = s)\,\mathbb{E}\,R_t(s) \end{aligned}$$

where the last equality is because $R_t(s)$ and $I_{\{\bar{E}_t,B_t(\epsilon),S_t=s\}}$ are independent and independent random variables are uncorrelated. Since $\sum_{s\in\mathcal{L}_t} \mathbb{P}(\bar{E}_t, \bar{B}_t(\epsilon), S_t = s) \le 1$, if we can prove $\mathbb{E}\,R_t(s) \le g(\epsilon)$ for all $s \in \mathcal{L}_t$, we are done.

Step 2: Prove $\mathbb{E}\,R_t(s) \le g(\epsilon)$ by discussing different cases of $s \in \mathcal{L}_t$.

We will divide $\mathcal{L}_t$ into different cases, and prove $\mathbb{E}\,R_t(s) \le g(\epsilon)$ in each case.

First of all, if $s = S_t^*$, then $\mathbb{E}\,R_t(s) = 0$.

When $s \ne S_t^*$, $s \in \mathcal{L}_t$ must satisfy either $s \subsetneqq S_t^*$ or $S_t^* \subsetneqq s$, by Lemma 7 and $\epsilon \le \frac{\Delta_{r_t}}{2}$. When $s \subsetneqq S_t^*$, denote $K_t = S_t^* - s$. When $S_t^* \subsetneqq s$, denote $H_t = s - S_t^*$.

Based on H_t and K_t, we will consider four cases.

Case 1: $s \subsetneqq S_t^*$ and $|K_t| = 1$

By Lemma 7 and $\epsilon \le \frac{\Delta_{r_t-1}}{2}$, $s = \{\sigma(1), \ldots, \sigma(r_t-1)\}$.

$$\begin{aligned} \mathbb{E}\,R_t(s) &= (\sum_{i=1}^{r_t-1} p_{\sigma(i)} - D_t)^2 \\ &\quad - (\sum_{i=1}^{r_t} p_{\sigma(i)} - D_t)^2 - p_{\sigma(r_t)}(1 - p_{\sigma(r_t)}) \\ &= -p_{\sigma(r_t)}(2\sum_{i=1}^{r_t-1} p_{\sigma(i)} - 2D_t + p_{\sigma(r_t)}) \\ &\quad - p_{\sigma(r_t)}(1 - p_{\sigma(r_t)}) \end{aligned}$$

$$= -p_{\sigma(r_t)}(2\sum_{i=1}^{r_t-1} p_{\sigma(i)} - 2D_t + 1)$$
$$= 2p_{\sigma(r_t)}\delta_{t,2} \le 2\delta_{t,2}$$

Since $S_t = \{\sigma(1), \ldots, \sigma(r_t - 1)\}$,

$$D_t - 1/2 < \sum_{i=1}^{r_t-1} \bar{p}_i(t-1) \le \sum_{i=1}^{r_t-1}(p_i + \epsilon)$$
$$= D_t - 1/2 - \delta_{t,2} + (r_t - 1)\epsilon$$

So $\delta_{t,2} < (r_t - 1)\epsilon < n\epsilon$. So $\mathbb{E}\, R_t(s) \le 2n\epsilon \le ((2n+2)\epsilon + 2)(2n+2)\epsilon$.

Case 2: $s \subsetneqq S_t^*$ and $|K_t| \ge 2$:

$$\mathbb{E}\, R_t(s) = (\sum_{i\in s} p_i - D_t)^2$$
$$+ \sum_{i\in s} p_i(1-p_i) - (\sum_{i\in S_t^*} p_i - D_t)^2 - \sum_{i\in S_t^*} p_i(1-p_i)$$
$$\le (2\sum_{i\in S_t^*} p_i - 2D_t - \sum_{i\in K_t} p_i)(-\sum_{i\in K_t} p_i) - \sum_{i\in K_t} p_i(1-p_i)$$
$$\le (2\sum_{i\in S_t^*} p_i - 2D_t - \sum_{i\in K_t} p_i)(-\sum_{i\in K_t} p_i)$$
$$\le (1 - 2\delta_{t,1} + \sum_{i\in K_t} p_i)(\sum_{i\in K_t} p_i)$$
$$\le (1 + \sum_{i\in K_t} p_i)(\sum_{i\in K_t} p_i)$$
$$\le (2n\epsilon + 1)2n\epsilon \le ((2n+2)\epsilon + 2)(2n+2)\epsilon$$

The second last inequality is because $\sum_{i\in K_t} p_i \le 2n\epsilon$. The proof is the following. First, by definition of S_t^*,

$$\sum_{i\in S_t^*} p_i \le D_t - 1/2 + p_{\sigma(r_t)} \tag{26}$$

Then by algorithm design and $\bar{E}_t$ and $\bar{B}_t(\epsilon)$, the selected subset $S_t = s$ satisfies

$$D_t - 1/2 < \sum_{i\in s} \bar{p}_i(t-1) \le \sum_{i\in s}(p_i + \epsilon) \tag{27}$$

Finally, let (26) minus (27), we can prove $\sum_{i\in K_t} p_i \le n\epsilon$.

$$\sum_{i\in K_t} p_i - |s|\epsilon < p_{\sigma(r_t)} \le \frac{\sum_{i\in K_t} p_i}{|K_t|} \le \frac{\sum_{i\in K_t} p_i}{2}$$
$$\Rightarrow \sum_{i\in K_t} p_i \le 2n\epsilon$$

Case 3: $s \supsetneqq S_t^*$ and $|H_t| = 1$.

In this case, $S_t^* \ne [n]$, and $\sum_{i=1}^n p_i > D_t - 1/2$.

By Lemma 7 and $\epsilon \le \frac{\Delta_{r_t+1}}{2}$, $H_t = \{p_{\sigma(r_t+1)}\}$, and $s = \{\sigma(1), \ldots, \sigma(r_t+1)\}$.

$$\mathbb{E}\, R_t(s) = (\sum_{i=1}^{r_t+1} p_{\sigma(i)} - D_t)^2$$
$$- (\sum_{i=1}^{r_t} p_{\sigma(i)} - D_t)^2 + p_{\sigma(r_t+1)}(1 - p_{\sigma(r_t+1)})$$
$$= p_{\sigma(r_t+1)}(2\sum_{i=1}^{r_t} p_{\sigma(i)} - 2D_t + p_{\sigma(r_t+1)})$$
$$+ p_{\sigma(r_t+1)}(1 - p_{\sigma(r_t+1)})$$
$$= p_{\sigma(r_t+1)}(2\sum_{i=1}^{r_t} p_{\sigma(i)} - 2D_t + 1)$$
$$= 2p_{\sigma(r_t)}\delta_{t,1} \le 2\delta_{t,1}$$
$$\le 2n\epsilon$$

The last inequality is because $\delta_{t,1} \le n\epsilon$. The reason is the following. Since $S_t = \{\sigma(1), \ldots, \sigma(r_t + 1)\}$,

$$D_t - 1/2 \ge \sum_{i=1}^{r_t} \bar{p}_i(t-1) \ge \sum_{i=1}^{r_t}(p_{\sigma(i)} - \epsilon)$$
$$\ge D_t - 1/2 + \delta_{t,1} - r_t\epsilon$$

So $\delta_{t,1} < (r_t)\epsilon < n\epsilon$. So $\mathbb{E}\, R_t(s) \le 2n\epsilon$.

Case 4: $s \supsetneqq S_t^*$ and $|H_t| \ge 2$.

In this case, $S_t^* \ne [n]$, and $\sum_{i=1}^n p_i > D_t - 1/2$.

$$\mathbb{E}\, R_t(s) = (\sum_{i\in s} p_i - D_t)^2$$
$$+ \sum_{i\in s} p_i(1-p_i) - (\sum_{i\in S_t^*} p_i - D_t)^2 - \sum_{i\in S_t^*} p_i(1-p_i)$$
$$= (2\sum_{i\in S_t^*} p_i - 2D_t + \sum_{i\in H_t} p_i)(\sum_{i\in H_t} p_i) + \sum_{i\in H_t} p_i(1-p_i)$$
$$\le (2\sum_{i\in S_t^*} p_i - 2D_t + \sum_{i\in H_t} p_i)(\sum_{i\in H_t} p_i) + \sum_{i\in H_t} p_i$$
$$\le (2\sum_{i\in S_t^*} p_i - 2D_t + \sum_{i\in H_t} p_i + 1)(\sum_{i\in H_t} p_i)$$
$$\le (2\delta_{t,1} + \sum_{i\in H_t} p_i)(\sum_{i\in H_t} p_i)$$
$$\le (2 + \sum_{i\in H_t} p_i)(\sum_{i\in H_t} p_i)$$
$$\le ((2n+2)\epsilon + 2)(2n+2)\epsilon$$

The last inequality is because $\sum_{i\in H_t} p_i \le (2n+2)\epsilon$. The proof is the following. Suppose $s = \{\sigma(1), \ldots, \sigma(m)\}$. First, by definition of S_t^*, and the fact that $S_t^* \ne [n]$, we have

$$\sum_{i\in S_t^*} p_i > D_t - 1/2 \tag{28}$$

Then by algorithm design and $\bar{E}_t$ and $\bar{B}_t(\epsilon)$, the selected subset $S_t = s$ satisfies

$$D_t - 1/2 + p_{\sigma(m)} + \epsilon \ge D_t - 1/2 + \bar{p}_{\sigma(m)}(t-1)$$
$$\ge \sum_{i\in s} \bar{p}_i(t-1) \ge \sum_{i\in s}(p_i - \epsilon) \tag{29}$$

Finally, let (28) minus (29), we can prove $\sum_{i\in H_t} p_i \le (2n+2)\epsilon$.

$$\sum_{i\in H_t} p_i - m\epsilon \le p_{\sigma(m)} + \epsilon \le \frac{\sum_{i\in H_t} p_i}{|H_t|} + \epsilon \le \frac{\sum_{i\in H_t} p_i}{2} + \epsilon$$
$$\Rightarrow \sum_{i\in H_t} p_i \le 2(m+1)\epsilon \le 2(n+1)\epsilon$$

□

F. Proof of Lemma 8 without Assumption (A1)

The proof is very similar to the proof of Lemma 8 with Assumption (A1) in Appendix E. We only explain the difference here. Note that when there are ties, the offline algorithm $\phi(p, D_t)$ may have multiple possible outputs, and these outputs share the same expected single-step loss $\mathbb{E}\, R_t$.

Define $k_1(t)$ and $k_2(t)$ following the Definition 1 but with respect to D_t instead of D. Based on our discussion before, we know $\phi(p, D_t)$ includes $\{\sigma(1), \ldots, \sigma(k_1(t))\}$ and $|\phi(p, D_t)| - k_1$ arms whose parameter is equal to $p_{\sigma(k_1)}$.

By Corollary 6, S_t must be one of the following cases

i) $S_t \subseteq \{\sigma(1), \ldots, \sigma(k_1(t))\}$,

ii) $\{\sigma(1), \ldots, \sigma(k_1)\} \subsetneqq S_t \subseteq \{\sigma(1), \ldots, \sigma(k_2)\}$, and S_t has less than $|\phi(p, D_t)| - k_1$ arms whose parameter is equal to $p_{\sigma(k_1)}$.

iii) S_t is one possible output of $\phi(p, D_t)$

iv) $\{\sigma(1), \ldots, \sigma(k_1)\} \subsetneqq S_t \subseteq \{\sigma(1), \ldots, \sigma(k_2)\}$, and S_t has more than $|\phi(p, D_t)| - k_1$ arms whose parameter is equal to $p_{\sigma(k_1)}$.

v) $\{\sigma(1), \ldots, \sigma(k_2)\} \subsetneqq S_t$.

Case i) and Case ii) can be bounded in the same way as Case 1 and 2 in the proof of Lemma 8 with Assumption (A1). Case iii) has zero regret. Case iv) and v) can be bounded in the same way as Case 3 and 4 in the proof of Lemma 8 with Assumption (A1). Then, we conclude the proof.