Designing Truthful Contextual Multi-Armed Bandits based Sponsored Search Auctions

by

Kumar Abhishek, Shweta Jain, Sujit Prakash Gujar

in

International Conference on Autonomous Agents and Multiagent Systems, AAMAS

Report No: IIIT/TR/2020/-1

Centre for Visual Information Technology
International Institute of Information Technology
Hyderabad - 500 032, INDIA
February 2020

Designing Truthful Contextual Multi-Armed Bandits based Sponsored Search Auctions

Kumar Abhishek
IIIT Hyderabad, India.
kumar.abhishek@research.iiit.ac.in

Shweta Jain
IIT Ropar, India.
shwetajain@iitrpr.ac.in

Sujit Gujar
IIIT Hyderabad, India.
sujit.gujar@iiit.ac.in

Abstract

For sponsored search auctions, we consider contextual multi-armed bandit problem in the presence of strategic agents. In this setting, at each round, an advertising platform (center) runs an auction to select the best-suited ads relevant to the query posted by the user. It is in the best interest of the center to select an ad that has a high expected value (i.e., probability of getting a click × value it derives from a click of the ad). The probability of getting a click (CTR) is unknown to the center and depends on the user's profile (context) posting the query. Further, the value derived for a click is the private information to the advertiser and thus needs to be elicited truthfully. The existing solution in this setting is not practical as it suffers from very high regret ($O(T^{\frac{2}{3}})$).

Towards designing practically useful mechanisms, we first design an elimination-based algorithm *ELinUCB-SB* that is ex-post monotone, which is a sufficient condition for truthfulness. Thus, *ELinUCB-SB* can be naturally extended to ex-post incentive compatible and ex-post individually rational mechanism *M-ELinUCB-SB*. We show via experiments that the proposed mechanisms outperform the existing mechanism in this setting. Theoretically, however, the mechanism may incur linear regret in some instances, which may not occur frequently. To have a theoretically stronger mechanism for regret, we propose a *SupLinUCB*-based allocation rule *SupLinUCB-S*. With the help of *SupLinUCB-S*, we design a mechanism *M-SupLinUCB-S*, which is ex-post incentive compatible and ex-post individually rational. We prove that it has regret $O(n^2\sqrt{dT\log T})$ as against $O(n\sqrt{dT\log T})$ for non-strategic settings; $O(n)$ is price of truthfulness. We demonstrate the efficacy of our mechanisms via simulation and establish superior performance than the existing literature.

1 Introduction

Internet advertising is one of the booming and rapidly increasing industry with revenue volume in billions of dollars [13]. The majority of the revenue generated by search engines like Google, Yahoo, and Bing comes from advertisements displayed on their platform. Typically, for any search query by a user, a search engine/the advertising platform, henceforth a *center*, displays ads along with the relevant results via an auction mechanism known as *sponsored search auction* (SSA). The fundamental difference between traditional advertising and Internet advertising is the payment model. In the former, the advertisers (henceforth *agents*), pay based on the number of impressions whereas in latter, the agents pay only if their ad receives a click. Thus, the probability of an ad getting clicked, referred to as *click-through rate* (CTR), plays a crucial role in SSA. The CTR of an ad is unknown to the center, but it can learn CTRs by displaying the ad repeatedly over a period of time. Each agent also has a private valuation for its ad, which represents its willingness to pay for a click. This valuation needs to be elicited from the agents truthfully.

In the absence of contexts, when the agents report their true valuations, we can model the problem as a Multi-Armed Bandit (MAB) problem [16] with agents playing the role of the arms. As the agents (arms) are strategic, they may misreport their valuations to maximize their utility. To elicit truthful bids from the agents, researchers have used Mechanism Design [2, 20]. However, such mechanisms are oblivious to the learning requirements and fail to avoid manipulations by the agents when learning is involved. In such cases, the researchers have modeled this problem as a MAB mechanism [7, 11, 12, 14]. The authors designed *ex-post truthful* mechanisms wherein the agents are not able to manipulate even when the random clicks are known to them.

Typically an individual user tends to click some specific ads more often than the other ads, which depends upon the profile of an individual user of the platform. In this work, we leverage this fact and use the profile of the user as features (for context) to personalize the ads to increase the number of clicks and hence, the *social welfare*. When the CTRs of the ads depend on the specific context at a particular round, we can model the problem as a *Contextual MAB* (ConMAB) problem [4, 17, 18, 1]. However, a naive implementation of ConMAB is not adequate in the presence of strategic agents.

To the best of our knowledge, contextual information in SSA is considered only in [12]. The authors proposed a novel, theoretically sound, deterministic, exploration-separated mechanism that offers strong game-theoretic properties. However, it faces multiple practical challenges: (i) it incurs high cost of learning (*regret*), (ii) the center needs to know the number of rounds for which it needs to execute SSA, and (iii) the initial rounds being free, a malicious agent may drop off after free rounds; in some cases, all the rounds could be free.

Contributions

In the presence of strategic agents, random context-arrivals, and stochastic clicks, our goal is to design a non-exploration-separated, ex-post truthful mechanism that (i) learns CTRs efficiently (minimizes regret), (ii) may not need prior knowledge of T, and (iii) does not have free rounds. We leverage popular algorithms *LinUCB* [18] and *SupLinUCB* [10] that perform well in estimating CTRs in the contextual setting to build our randomized mechanisms to avoid manipulations by strategic agents. In particular, our contributions are:

- We adapt *LinUCB* to design an ex-post monotone allocation rule *ELinUCB-S* for a single-slot SSA (Theorem 2). We further optimize *ELinUCB-S* by introducing batch level update to propose *ELinUCB-SB* and using resampling procedure by [6], we develop an ex-post truthful mechanism *M-ELinUCB-SB*, which is also ex-post individually rational (Theorem 3). Unlike existing ConMAB mechanism, *M-ELinUCB-SB* does not need to know T.

- For stronger theoretical guarantees, we adapt *SupLinUCB* to design an ex-post monotone allocation rule *SupLinUCB-S* for a single-slot SSA (Theorem 11). We prove that *SupLinUCB-S* has regret $O(n^2\sqrt{dT\log T})$ (Theorem 9) as against $O(n\sqrt{dT\log T})$ for the non-strategic settings; we attribute $O(n)$ as price of truthfulness. Using resampling procedure, we develop *M-SupLinUCB-S* which is ex-post truthful and ex-post individually rational. *M-SupLinUCB-S*, however, needs to know T upfront.

- We study *M-ELinUCB-SB* and *M-SupLinUCB-S* with the existing mechanism *M-Reg*, on simulated data and provide empirical analysis. Empirically, *M-ELinUCB-SB* performs superior to *M-SupLinUCB-S* by large factors for less than million rounds.

2 Preliminaries

First, we define our model and notation.

2.1 Model and Notation

There is a fixed set of agents $\mathcal{N} = \{1, 2, \ldots, n\}$, where each agent has exactly one ad to display and the center has one slot available for allocation. A contextual $n-$armed Multi-Armed Bandit (MAB) mechanism $\mathcal{M}$ proceeds in discrete rounds $t = 1, 2, \ldots, T$. At each round t:

1. $\mathcal{M}$ observes a context $x_t \in [0, 1]^d$ which summarizes the profile of the user arriving at round t.

2. Based on the history, h_t, of allocations, observed clicks, and the context x_t, $\mathcal{M}$ chooses an agent $I_t \in \mathcal{N}$ to display it's ad.

3. A click r_{I_t} is observed which is 1 if it gets clicked and 0 otherwise.

4. Mechanism $\mathcal{M}$ decides the positive payment $p_{I_t,t}$ to be made by the agent I_t to the center. The payment by any other agent is 0.

5. Update $h_t = h_{t-1} \cup \{x_t, \{I_t\}, \{r_{I_t}\}\}$.

6. The mechanism then improves its arm-selection strategy with new observation $(x_t, \{I_t\}, \{r_{I_t}\})$. No feedback is received for the agents that are not selected.

Each agent i is thus characterized by two quantities: (i) private valuation $v_i \in [0,1]$, which represents the willingness to pay for the click received and is constant throughout the rounds, and (ii) click through rate (CTR) of its ad $\mu_i(x_t) \in [0,1]$ which is an unknown parameter and is dependent on the context x_t. Each agent i submits the valuation of getting a click on its ad as bid b_i. We assume that the bids are constant across the rounds (since v_i's are constant). We assume that the CTR of an agent i is linear in d-dimensional context x_t with some unknown coefficient vector θ_i [18]. Thus, the problem reduces to learning the $d-$dimensional vector θ_i for each agent i. The probability of getting a click on the ad of agent i at any given round t is given as:

$$\mathbb{P}[r_{i,t}|x_t] = \mu_i(x_t) = \theta_i^\intercal x_t$$

Thus, the expected valuation of agent i is $v_i\mu_i$. Let b_{-i} be the bid vector of all the agents other than i and b denote the bid vector of all the agents. The utility of an agent i in round t with history h_t is given as:

$$u_{i,t}(b_i, b_{-i}, x_t; h_t; v_i) = \mathbb{1}\{I_t(b, x_t; h_t) = i\} r_{i,t}(v_i - p_{i,t}(b; h_t))$$

and the utility of center is given as:

$$u_t^c(x_t) = \sum_{i=1}^{N} \{I_t(b, x_t; h_t) = i\} r_{i,t} p_{i,t}$$

In this work, our aim is to maximize social welfare similar to [7, 14]. The social welfare at round t is evaluated as sum of utilities of the agents and the center and is given as:

$$sw_t(x_t) = \sum_{i=1}^{N} \{I_t(b, x_t; h_t) = i\} r_{i,t} v_i.$$

When the CTRs are not known, the efficiency of any mechanism is measured by its rate of learning or regret. Thus, our goal reduces to design a mechanism $\mathcal{M}$ that minimizes the social welfare regret which is given as:

$$\mathbb{R}_T(\mathcal{M}) = \sum_{t=1}^{T} [\theta_{i_t^*}^T x_t \cdot b_{i_t^*} - \theta_{I_t}^T x_t \cdot b_{I_t}] \tag{1}$$

Here, $i_t^*(x_t)$ denote the highest expected valuation (based on bids) i.e., $i_t^*(x_t) = argmax_k\{b_k \cdot (\theta_k^T x_t)\}$.

In the following section, we define game theoretic properties relevant to this work.

2.2 Game Theoretic Properties

A mechanism $\mathcal{M} = (\mathcal{A}, \mathcal{P})$ (where $\mathcal{A}$ is the allocation rule and $\mathcal{P}$ is the payment rule) is ex-post truthful (formally called EPIC) if and only if $\mathcal{A}$ is ex-post monotone [19, 3]. That is, for all instances of possible click realizations and context-arrivals, by increasing its bid to $b_i^+ > b_i$, agent i should obtain at least same number of clicks at bid b_i if not more. Formally,

Definition 1. Ex-post monotonicity: Let $\nu_i(b_i, t)$ denote total number of clicks on the ad of agent i in first t rounds. Then, $\mathcal{A}$ is ex-post monotone if for every possible sequence of context-arrivals and click realizations, for each agent $i \in \mathcal{N}, \forall t, \forall b_{-i}$ and two possible bids of i, $b_i^+ \geq b_i$ we have

$$\nu_i(b_i^+, t) \geq \nu_i(b_i, t).$$

Definition 2. EPIC: A mechanism $\mathcal{M} = (\mathcal{A}, \mathcal{P})$ is said to be *ex-post incentive compatible* ($EPIC$) if by misreporting the bid, no agent can gain its total utility more than that it would have obtained by bidding truthfully, i.e., $\forall i, \forall b_{-i}, \forall v_i, \forall h_t, \forall b_i$,

$$\sum_{t=1}^{T} u_{i,t}(v_i, b_{-i}, x_t; h_t; v_i) \geq \sum_{t=1}^{T} u_{i,t}(b_i, b_{-i}, x_t; h_t; v_i).$$

EPIC implies even if an agent has observed all the contexts and all the click realizations, it is in the agent's best interest to report true valuation. Note that, if a mechanism does any randomization, $u_{i,t}(\cdot)$ is replaced by $\mathbb{E}[u_{i,t}(\cdot)]$, where the expectation is taken w.r.t. randomization in the mechanism. Such a mechanism is still truthful for every realization of external randomness such as click realizations, context-arrivals.

One alternative notion of IC which may seem suitable in our model is dynamic IC [8]. We would like the reader to note that in our model, the bids, as well as valuations, are constant throughout the rounds, and not dependent on any round t. Thus, private information and communication with the mechanism are not dynamic. Hence, a strong game-theoretic property of ex-post IC is more apt in our model.

Definition 3. EPIR: A mechanism $\mathcal{M} = (\mathcal{A}, \mathcal{P})$ is said to be *ex-post individually rational* ($EPIR$) if every agent has a non-negative utility with truthful bidding irrespective of the bids of other agents i.e., $\forall i, \forall x_t, \forall v_i, \forall t, \forall h_t$,

$$u_{i,t}(v_i, b_{-i}, x_t; h_t; v_i) \geq 0.$$

The authors in [6] have shown the power of randomization in designing truthful mechanisms by proposing a randomized context-free MAB mechanism that is ex-post truthful and has regret $O(\sqrt{T})$. Thus, by introducing randomness in the mechanism, they showed that it is possible to bypass the impossibility result in [7], which states that any deterministic, truthful MAB mechanism has to be exploration-separated and hence must suffer a regret of $\Omega(T^{2/3})$. The main result of [6] involves designing the black box mechanism using the self-resampling procedure in Algorithm 1, which provides ex-post truthful and IR mechanism if given an ex-post monotone allocation rule (Theorem 1).

Algorithm 1 Non-recursive self-resampling procedure [6]

```
Input: bid $b = \{b_1, \ldots, b_n\}$, parameter $\delta \in (0, 1)$
Output: modified bid $y = \{y_1, \ldots, y_n\}$, $\eta = (\eta_1, \ldots, \eta_n)$
Independently for each agent $i \in \mathcal{N}$
    Sample: $\epsilon_i$ uniformly at random from $[0, 1]$
    with probability $1 - \delta$
        $\eta_i = 1$
    else
        $\eta_i = \epsilon_i^{1/(1-\delta)}$
Construct the vector of modified bids $y = (y_1, \ldots, y_n)$, where $y_i = \eta_i b_i$
```

Theorem 1. *(Theorem 4.5, [6]) Let $\mathcal{A}$ be ex-post monotone allocation rule. Applying the transformation in Algorithm 1 to $\mathcal{A}$ with parameter δ, we obtain a mechanism $\mathcal{M}$ such that $\mathcal{M}$ is EPIC and EPIR.*

There is no previous work done in designing the ex-post monotone allocation rule in the contextual setting. Hence, we address this problem and design two allocation rules, which are ex-post monotone, though it has different properties.

Concerning ConMAB mechanisms for SSA, two works are closely related to our work [12] and [18]. The former considers the strategic agents in ConMAB [12] by proposing a mechanism that we will call *M-Reg*. The mechanism is *exploration-separated* [11, 7] which is deterministic and induces EPIC property. The regret achieved by this mechanism is quite high $O(T^{2/3})$ as compared to $O(\sqrt{T})$ regret in the traditional ConMAB problem. The latter introduces *LinUCB* algorithm, which is particularly of interest to us. Hence we describe it below.

2.3 *LinUCB*

LinUCB [18] is a generic ConMAB algorithm that efficiently learns the CTR of an agent where the CTR model is linear in terms of context and the unknown parameters. The authors experimentally showed the efficacy of the algorithm in approximating the CTRs of news articles in news recommendation. Hence, we choose to adapt it to our setting. *LinUCB* is motivated by *UCB* [5] where upper confidence bound (UCB) is maintained for each agent. To capture the contextual information in ConMAB setting, *LinUCB* uses A_i and c_i for each agent i, where A_i summarizes the information about contexts and c_i corresponding clicks. It maintains upper confidence bound (UCB) for each agent i as $\mu^+_{i,t} \leftarrow \hat{\theta}_i^\top x_t + \alpha\sqrt{x_t^\top A_i^{-1} x_t}$ where $\hat{\theta}_i \leftarrow A_i^{-1} c_i$ and α is learning parameter. At round t, the algorithm selects the agent I_t with the highest UCB $\hat{\mu}_{i,t}$. The statistics for the selected agent I_t is updated as $A_{I_t} \leftarrow A_{I_t} + x_t x_t^\intercal$, $c_{I_t} \leftarrow c_{I_t} + r_{I_t} x_t$, where r_{I_t} is the indicator variable of receiving click.

LinUCB was originally designed to estimate CTRs of news articles and hence does not capture strategic manipulations. Motivated by *LinUCB*, we build randomized EPIC mechanisms for SSA by developing an ex-post monotone allocation rule, *ELinUCB-SB*, and using the resampling procedure

Algorithm 2 *LinUCB*

```
Inputs: $\alpha \in \mathbb{R}_+$
Initialization:
for all $i \in \mathcal{N}$ do
    $A_i \leftarrow I_d$ ($d$-dimensional identity matrix)
    $c_i \leftarrow 0_{d\times 1}$ ($d$-dimensional zero vector)
for $t = 1, 2, 3, \ldots, T$ do
    Observe context of user as $x_t$
    for all $i \in \mathcal{N}$ do
        $\hat{\theta}_i \leftarrow A_i^{-1} c_i$, $\mu^+_{i,t} \leftarrow \hat{\theta}_i^\intercal x_t + \alpha\sqrt{x_t^\intercal A_i^{-1} x_t}$
    $I_t = argmax_{i\in\mathcal{N}}\, \mu^+_{i,t}$, Observe $r_{I_t} \in \{0, 1\}$
    $A_{I_t} \leftarrow A_{I_t} + x_t x_t^\intercal$, $c_{I_t} \leftarrow c_{I_t} + r_{I_t} x_t$
```

(Algorithm 1) to design a randomized EPIC and EPIR mechanism *M-ElinUCB-SB* [6]. ELinUCB-SB has linear regret. We present it here as the key ideas to adapt *LinUCB* to design truthful mechanisms are useful and carry forward when we design a more complicated mechanism, *M-SupLinCUB-S*, based on *SupLinUCB* by [10].

3 *M-ELinUCB-SB*: Truthful ConMAB Mechanism 1

We first propose a single-slot allocation rule *ELinUCB-S* based on *LinUCB*. We next provide a further optimized algorithm *ELinUCB-SB* that incorporates mini-batch learning, which makes the algorithm efficient both in terms of regret and computation.

3.1 *ELinUCB-S*: LinUCB-Based Single-Slot SSA

Algorithm 3 *ELinUCB-S*: LinUCB-based allocation for single-slot SSA

```
Inputs: $n$, $\alpha \in \mathbb{R}_+$, bid vector $b$
Initialization: $S_{act} = \mathcal{N}$
for all $i \in \mathcal{N}$ do
    $A_i \leftarrow I_d$ ($d$-dimensional identity matrix)
    $c_i \leftarrow 0_{d\times 1}$ ($d$-dimensional zero vector)
    $\mu^+_i \leftarrow b_i; \mu^-_i \leftarrow 0$
for $t = 1, 2, 3, \ldots$ do
    Observe context as $x_t$
    $I_{t'} \leftarrow 1 + (t \mod n)$
    if $I_{t'} \in S_{act}$ then
        Allocate agent $I_{t'}$, i.e., $I_t \leftarrow I_{t'}$
        Observe click as $r_{I_t} \in \{0, 1\}$
        $A_{I_t} \leftarrow A_{I_t} + x_t x_t^\intercal$, $c_{I_t} \leftarrow c_{I_t} + r_{I_t} x_t$, $\hat{\theta}_{I_t} \leftarrow A_{I_t}^{-1} c_{I_t}$
        {Update confidence bound}
        if $\mu^-_{I_t} < \mu^+_{I_t}$ then
            $(\gamma^-_{I_t}, \gamma^+_{I_t}) \leftarrow b_{I_t}(\hat{\theta}_{I_t} x_t \mp \alpha\sqrt{x_t^\intercal A_{I_t}^{-1} x_t})$
            if $\max(\mu^-_{I_t}, \gamma^-_{I_t}) < \min(\mu^+_{I_t}, \gamma^+_{I_t})$ then
                $(\mu^-_{I_t}, \mu^+_{I_t}) \leftarrow (\max(\mu^-_{I_t}, \gamma^-_{I_t}), \min(\mu^+_{I_t}, \gamma^+_{I_t}))$
            else
                $(\mu^-_{I_t}, \mu^+_{I_t}) \leftarrow \left(\frac{\mu^-_{I_t}+\mu^+_{I_t}}{2}, \frac{\mu^-_{I_t}+\mu^+_{I_t}}{2}\right)$
    else
        $I_t \leftarrow argmax_i\, b_i \cdot (\hat{\theta}_i^T x_t)$, $\ni I_t \in S_{act}$
        Observe click as $r_{I_t} \in \{0, 1\}$
    for all agent $i \in S_{act}$ do
        if $\mu^+_i < \max_{k\in S_{act}} \mu^-_k$ then
            Remove $i$ from $S_{act}$
```

ELinUCB-S (Algorithm 3) for single-slot allocation maintains a set of active agents S_{act}. At each round, algorithm evaluates whether an agent should be retained in S_{act} or not. Once an agent is evicted from S_{act}, it can not be added back. For better understanding about the working of the algorithm, we virtually divide the *LinUCB-S* into 4 subroutines: i) Initialization (lines[1-7]) ii) Exploration (lines[11-20]) iii) Exploitation (lines[22-23]) iv) Elimination (lines[24-26]).[1] For each agent i, the algorithm maintains lower confidence bound (LCB) and upper confidence bound (UCB) as μ_i^- and μ_i^+ respectively.

At each round t, the algorithm observes context x_t. It determines the index of agent $I_{t'}$ whose turn is to display the ad based on round robin order, as stated in line[9]. The algorithm then checks if $I_{t'} \in S_{act}$. If it evaluates to true the algorithm runs Exploration subroutine else Exploitation. In Exploration subroutine the algorithm allocates the slot to $I_{t'}$, observes click $r_{I_t'}$ and updates its parameters. The confidence bounds are updated if and only if the size of confidence interval decreases (line[18]). In Exploitation subroutine, the agent with the maximum estimated expected valuation among the agents in S_{act} is allocated the slot and observes click r_{I_t}. It is important to note that no parameter is updated during Exploitation subroutine which is crucial for the ex-post monotonicity property. At the end of each round, Elimination subroutine is executed which removes the agents $j \in S_{act}$ from S_{act} if UCB of agent j is less than LCB of any other agent in S_{act}.

The intuition driving the algorithm is after sufficient exploration the confidence interval becomes sufficiently small, hence the agents which are close to optimal continues to remain in S_{act} and sub-optimal agents are eliminated.

Algorithm 4 *ELinUCB-SB*: LinUCB-based batch allocation rule for single-slot SSA

```
1:  **Inputs:** $n, T$, $\alpha \in \mathbb{R}_+$, bid vector $b$, batch size $bs$
2:  **Initialization:** $S_{act} = \mathcal{N}$, $x^{'} \leftarrow 0_{d\times 1}, T^{'} = \lfloor \frac{T}{bs} \rfloor$
3:  **for all** $i \in \mathcal{N}$ **do**
4:      $A_i \leftarrow I_d$ (d-dimensional identity matrix)
5:      $c_i \leftarrow 0_{d\times 1}$ (d-dimensional zero vector)
6:      $\mu_i^+ \leftarrow b_i; \mu_i^- \leftarrow 0$
7:  **for** $t^{'} = 1, 2, 3, \ldots, T^{'}$ **do**
8:      $I_{t^{'}} \leftarrow 1 + (t^{'} - 1) \mod n$
9:      **if** $I_{t^{'}} \in S_{act}$ **then**
10:         **for** $t = (t^{'} - 1)bs, \ldots, (t^{'} \cdot bs - 1)$ **do**
11:             Observe context as $x_t$
12:             $I_t \leftarrow I_{t^{'}}$,
13:             $x^{'} \leftarrow ((t-1)x^{'} + x_t)/t$ (averaging over contexts)
14:             Observe click as $r_{I_t} \in \{0, 1\}$
15:             $A_{I_t} \leftarrow A_{I_t} + x_t x_t^\intercal$, $c_{I_t} \leftarrow c_{I_t} + r_{I_t} x_t$, $\hat{\theta}_{I_t} \leftarrow A_{I_t}^{-1} c_{I_t}$
16:         **if** $\mu_{I_t}^- < \mu_{I_t}^+$ **then**
17:             $(\gamma_{I_t}^-, \gamma_{I_t}^+) \leftarrow b_{I_t}(\hat{\theta}_{I_t}^\intercal x^{'} \mp \alpha\sqrt{(x')^\intercal A_{I_t}^{-1} x'})$
18:             **if** $\max(\mu_{I_t}^-, \gamma_{I_t}^-) < \min(\mu_{I_t}^+, \gamma_{I_t}^+)$ **then**
19:                 $(\mu_{I_t}^-, \mu_{I_t}^+) \leftarrow (\max(\mu_{I_t}^-, \gamma_{I_t}^-), \min(\mu_{I_t}^+, \gamma_{I_t}^+))$
20:             **else**
21:                 $\left(\mu_{I_t}^-, \mu_{I_t}^+\right) \leftarrow \left(\frac{\mu_{I_t}^- + \mu_{I_t}^+}{2}, \frac{\mu_{I_t}^- + \mu_{I_t}^+}{2}\right)$
22:     **else**
23:         **for** $t = (t^{'} - 1)bs, \ldots, (t^{'} \cdot bs - 1)$ **do**
24:             Observe $x_t$
25:             $I_t \leftarrow argmax_i\, b_i \cdot (\hat{\theta}_i^T x_t), \ \ni I_t \in S_{act}$
26:             Observe click as $r_{I_t} \in \{0, 1\}$
27:     **for all** agent $i \in S_{act}$ **do**
28:         **if** $\mu_i^+ < \max_{k \in S_{act}} \mu_k^-$ **then**
29:             Remove $i$ from $S_{act}$
```

3.2 Regret Analysis of *ELinUCB-SB*

Although the algorithm *ELinUCB-S* seems promising, the dynamic and varying nature of contexts and its arrival order may lead to the elimination of an optimal agent. Hence, it may continue to allocate sub-optimal agents in subsequent rounds leading to high regret on specific instances, which is evident from our simulation of the algorithm (Fig.1c). The updates in μ_i^+, μ_i^- depend upon the context in such a way that μ_i^+ is non-increasing and μ_i^- is non-decreasing, as stated and proved

[1] Note that this is virtual division and proposed algorithms is not actually exploration-separated where initial rounds are only exploration as well as free and then exploitation where no update happens.

in Claim 2. These updates being irreversible needs to be carefully handled to optimize regret. To counter this problem, we design *ELinUCB-SB* (Algorithm 4) in which we have introduced a subtle, yet important use of mini-batch. The algorithm *ELinUCB-SB* allocates an agent for *bs* number of rounds instead of one round. It follows similar rules for allocating agents and maintaining the active set S_{act}. It updates the bounds of agents by taking the average over the contexts arrived in *bs* rounds. Updating the bounds over the average of context after the completion of batch allocation handles the variance in contexts and its arrivals, thus reducing the regret significantly.

It can be shown that eventually, *ELinUCB-SB* will eliminate all but one arm. The remaining arm will be the dominant arm in most of the contexts. However, we can construct examples where this arm is not the best for at least one context, which has non-zero probability and thus leading to $O(T)$ regret. However, the round number at which it happens is generally very high, which we validate experimentally. Even though *ELinUCB-SB* incurs linear regret theoretically, it performs well in experiments and has interesting monotonicity properties; the proofs we leverage while designing ex-post truthful ConMAB mechanism with sub-linear regret in the next section.

3.3 Monotonicity of *ELinUCB-S*

We now prove ex-post monotonicity property for the proposed allocation rule.

For a fixed sequence of context-arrivals $\{x_t\}_t$, and click realization ρ, let $S_{act}(b,t)$ be the set of active agents in the beginning of round t when agents bid $b = (b_i, b_{-i})$. For each agent i, let $\mu_i^-(b,t)$ and $\mu_i^+(b,t)$ be the values of μ_i^- and μ_i^+ in the round t and similarly when agents bid b'. We prove ex-post monotonicity with the following claims.

Claim 1. For fixed context-arrivals $\{x_t\}_t$ and click realization ρ, let two bid vectors be b and b', then for each round t, if $i \in S_{act}(b,t) \cap S_{act}(b^{'},t)$, then:

$$\mu_i^-(b,t)/b_i = \mu_i^-(b',t)/b'_i \text{ and } \mu_i^+(b,t)/b_i = \mu_i^+(b',t)/b'_i$$

Proof. μ_i^+ and μ_i^- are updated only in Exploration subroutine which is based on round-robin order and hence does not depend on bid. Thus, the claim follows from the fact that contexts and click realizations are fixed. ■

Claim 2. For a fixed bid vector b, and each agent $i : \mu_i^- \leq \mu_i^+$, then for all $(t-1, t)$ consecutive rounds μ_i^- is non-decreasing and μ_i^+ is non-increasing.

Proof. From lines[15-20] of the Algorithm 3, we have: $\mu_i^-(b,t-1) \leq \mu_i^-(b,t) \leq \mu_i^+(b,t) \leq \mu_i^+(b,t-1)$. Hence the claim holds. ■

Claim 3. For any two bid vectors b^+ and b, where $b_i^+ \geq b_i$, $b_j^+ = b_j \ \forall j \neq i$ and $i \in S_{act}(b^+,\tau) \cap S_{act}(b,\tau)$, then $\forall \tau \in \{1, 2, \ldots, T\}$, $S_{act}(b^+,\tau) \subseteq S_{act}(b,\tau)$ holds.

Proof. The condition $i \in S_{act}(b^+,\tau) \cap S_{act}(b,\tau)$ implies that if $i \in S_{act}(b^+,\tau)$, then $i \in S_{act}(b,\tau)$, hence satisfying the claim for i. For $j \neq i$, we will use induction on t. The claim trivially holds for $t = 1$. Let, $t \leq \tau$ be the last round such that $S_{act}(b^+,t) = S_{act}(b,t)$ and $S_{act}(b^+,t+1) \neq S_{act}(b,t+1)$. In this case, we prove that: $\forall j \neq i, j \in S_{act}(b^+,t+1) \implies j \in S_{act}(b,t+1)$.
Since, $j \neq i$, $\mu_j^+(b^+,t+1) = \mu_j^+(b,t+1)$, $\mu_z^-(b^+,t+1) = \mu_z^-(b,t+1) \ \forall z \neq i$, and $\mu_i^-(b^+,t+1) \geq \mu_i^-(b,t+1)$ from Claim 1. Thus,

$$\begin{aligned} \mu_j^+(b^+,t+1) &> \max_{z \in S_{act}(b^+,t)} \mu_z^-(b^+,t+1) \\ &\geq \max_{z \in S_{act}(b^+,t)} \mu_z^-(b,t+1) \\ \implies \mu_j^+(b,t+1) &> \max_{z \in S_{act}(b,t)} \mu_z^-(b,t+1) \end{aligned}$$

(Since $S_{act}(b,t) = S_{act}(b^+,t)$). Hence, $j \in S_{act}(b,t+1)$. From, induction hypothesis, $\forall t'$ s.t. $\tau > t' \geq t$, $S_{act}(b^+,t') \subseteq S_{act}(b,t')$. We will now prove that $S_{act}(b^+,t'+1) \subseteq S_{act}(b,t'+1)$.

Consider any $j \in S_{act}(b^+,t') \cap S_{act}(b,t')$: we will prove that if $j \in S_{act}(b^+,t'+1)$, then $j \in S_{act}(b,t'+1)$.

Since $j \in S_{act}(b^+,t') \cap S_{act}(b,t')$, $\mu_j^+(b^+,t'+1) = \mu_j^+(b,t'+1)$. Also, $\forall z \in S_{act}(b,t') \cap S_{act}(b^+,t')$ and $z \neq i$, $\mu_z^-(b^+,t'+1) = \mu_z^-(b,t'+1)$. Further, $\forall z$ such that $z \in S_{act}(b,t')$ but $z \notin S_{act}(b^+,t')$, $\exists l \in S_{act}(b^+,t')$ such that $\mu_z^+(b^+,t'+1) < \mu_l^-(b^+,t'+1) \implies \mu_z^-(b^+,t'+1) < \mu_l^-(b^+,t'+1)$.

Thus, $\max_{z \in S_{act}(b,t'+1)} \mu_z^-(b^+, t'+1) \leq \max_{z \in S_{act}(b^+,t'+1)} \mu_z^-(b^+, t'+1)$. Thus, $j \in S_{act}(b^+, t'+1)$ implies

$$\begin{aligned} \mu_j^+(b^+.t'+1) &\geq \max_{z \in S_{act}(b^+,t')} \mu_z^-(b^+, t'+1) \\ \implies \mu_j^+(b, t'+1) &\geq \max_{z \in S_{act}(b,t')} \mu_z^-(b^+, t'+1) \\ \implies \mu_j^+(b, t'+1) &\geq \max_{z \in S_{act}(b,t')} \mu_z^-(b, t'+1) \end{aligned}$$

■

Claim 4. For fixed context-arrivals, fixed click realizations, and fixed bids of the agents except i, that is, for a fixed b_{-i}, if $i \in S_{act}(t, b)$, then $i \in S_{act}(t, b^+)$ where $b = (b_i, b_{-i})$ and $b^+ = (b_i^+, b_{-i})$; $b_i^+ > b_i$.

Proof. Let $\tau^* \geq 1$ be the last round for i s.t. it is in active set with both bids. From Claim 1, $\mu_i^+(b^+, \tau^*) = \frac{b_i^+}{b_i} \mu_i^+(b, \tau^*) > \mu_i^+(b, \tau^*)$ as $i \in S_{act}(b, \tau^*) \cap S_{act}(b^+, \tau^*)$. As the context-arrivals, click realizations and bids of the remaining agents are fixed, if agent i becomes inactive with bid b_i^+ then

$$\begin{aligned} \mu_i^+(b^+, \tau^*) &< \max_{k \in S_{act}(b^+,\tau^*-1)} \mu_k^-(b^+, \tau^*) \\ \implies \mu_i^+(b, \tau^*) &< \max_{k \in S_{act}(b^+,\tau^*-1)} \mu_k^-(b^+, \tau^*) \\ \implies \mu_i^+(b, \tau^*) &< \max_{k \in S_{act}(b,\tau^*-1)} \mu_k^-(b, \tau^*). \end{aligned}$$

The last line follows from Claim 3. Thus, i is also inactive with bid b_i. ■

Theorem 2. *The allocation rule induced by ELinUCB-S (Algorithm 3) is ex-post monotone.*

Proof. For a fixed context-arrivals $\{x_t\}_t$, click realization ρ, bids of agents except i, i.e., b_{-i} and two possible bids $b_i^+ > b_i$, let τ and τ^+ be the last round for i s.t. i is in active set with bids b_i and b_i^+ respectively. From Claim 4, $\tau^+ \geq \tau$. Thus, i will receive more number of rounds with bid b_i^+ as compared with bid b_i. ■

Proposition 1. The allocation rule induced by ELinUCB-SB (Algorithm 4) is ex-post monotone.

Proof. The difference between Algorithm 3 and Algorithm 4 is the introduction of batch allocation. In Algorithm 4 the unit of one round is equivalent to bs rounds in Algorithm 3. Hence, by replacing variable t with $t^{'}$ where $t \in \{1, 2, \ldots, T\}$ and $t^{'} \in \{1, 2, \ldots, \lfloor \frac{T}{bs} \rfloor\}$ will satisfy all the claims (Claim 1-4). Thus, *ELinUCB-SB* (Algorithm 4) is still ex-post monotone. ■

M-ELinUCB-SB

We now propose the following mechanism *M-ELinUCB-SB* for the single-slot SSA. A mechanism is defined as $\mathcal{M} = (\mathcal{A}, \mathcal{P})$. The outline of both mechanisms is defined in Mechanism 5. For both the mechanisms, we apply the resampling procedure [6] on the bids and the allocation in both cases are based on the modified bids, where δ is resampling parameter. For *M-ELinUCB-SB*, $\mathcal{A}$ is given by *ELinUCB-SB*. The payment $\mathcal{P}$ at round t, corresponding context x_t, $\forall i \in \mathcal{N}$ is given by $p_{i,t} = b_i \cdot (\mathbb{1}\{I_t = i\})$ if $\eta_i = 1$ and $p_{i,t} = b_i \cdot (\mathbb{1}\{I_t = i\}) \cdot (1 - \frac{1}{\delta})$ if $\eta_i < 1$.

Algorithm 5 *M-ELinUCB-SB*: LinUCB-based ex-post truthful mechanism

```
1: Input: bid vector $b$, resampling parameter $\delta$
2: Run self-resampling procedure on bid vector $b$, obtain modified bid vector $y = (y_1, \ldots, y_n)$, $\eta = (\eta_1, \ldots, \eta_n)$
3: Allocate according to $\mathcal{A}(y, t)$
4: For each agent $i$, assign payment $p_{i,t} = b_i \cdot \mathcal{A}_{i,t}(y, t) \cdot \begin{cases} 1 & \text{if } \eta_i = 1 \\ 1 - \frac{1}{\delta} & \text{if } \eta_i < 1 \end{cases}$
```

3.4 *M-ELinUCB-SB*: Game Theoretic Analysis

Theorem 3. M-ELinUCB-SB *is ex-post incentive compatible (EPIC) and ex-post individually rational (EPIR) mechanism.*

Proof. The result follows from Theorem 1 and by ex-post monotonicity of $\mathcal{A}$ defined in Algorithm 4. ■

4 *M-SupLinUCB-S*: Truthful ConMAB Mechanism 2

As the mechanism with allocation rule in Algorithm 4 can incur linear regret, in this section, we propose a new ConMAB mechanism for SSA that achieves sub-linear regret. First, we explain how we adapt SupLinUCB [10] for SSA to derive an ex-post monotone allocation algorithm SupLinUCB-S in the next subsection. In Section 4.2, we prove the regret bound on SupLinUCB-S. Then we prove the monotonicity of it. Finally, we design a truthful mechanism M-SupLinUCB-S.

4.1 *SupLinUCB-S*

Chu et al. [10] proposed SupLinUCB for contextual MAB settings with linear payoffs. First let us emphasize the major differences between SupLinUCB ([10]) and SupLinUCB-S (proposed here, Algorithm 6). (i) Chu et al. have considered a common θ to be learned across all the agent and for each round t the contexts are different for each agent whereas in our setting we have independent θ_i to be learned for each agent i whereas the context across each agent is same. (ii) We have adapted their algorithm for auction setting such that it satisfies *ex-post monotonicity* property, which is necessary to design ex-post truthful mechanisms. Our algorithm is presented in 6.

Algorithm 6 *SupLinUCB-S:(Adapted from SupLinUCB by [10] to satisfy monotonicity property)*

```
Initialization: $S \leftarrow \ln T$, $\Psi^s_{i,t} \leftarrow \phi$ for all $s \in [\ln T]$
for t = 1,2,..., T do
  $s \leftarrow 1$ and $\hat{A}_1 \leftarrow \mathcal{N}$
  $j \leftarrow 1 + (t \bmod n)$
  repeat
    Use BaseLinUCB-S with $\{\Psi^s_{i,t}\}_{i \in \mathcal{N}}$ and context vector $x_t$ to calculate the width $w^s_{i,t}$ and upper confidence bound $ucb^s_{i,t} = (\hat{r}^s_{i,t} + w^s_{i,t})$, $\forall i \in \hat{A}_s$
    if $j \in \hat{A}_s$ and $w^s_{j,t} > 2^{-s}$ then
      Select $I_t = j$
      Update the index sets at all levels:
      $\Psi^{s'}_{i,t+1} \leftarrow \begin{cases} \Psi^{s'}_{i,t} \cup \{t\} & \text{if } s = s' \\ \Psi^{s'}_{i,t} & \text{otherwise} \end{cases}$
    else if $w^s_{i,t} \leq \frac{1}{\sqrt{T}}, \forall i \in \hat{A}_s$ then
      Select $I_t = argmax_{i \in \hat{A}_s} b_i \cdot (\hat{r}^s_{i,t} + w^s_{i,t})$
      Update index sets at all levels for $I_t$:
      $\Psi^{s'}_{I_t,t+1} \leftarrow \Psi^{s'}_{I_t,t}$, $\forall s' \in [S]$
    else if $w^s_{i,t} \leq 2^{-s}, \forall i \in \hat{A}_s$ then
      $\hat{A}_{s+1} \leftarrow \{i \in \hat{A}_s | b_i \cdot (\hat{r}^s_{i,t} + w^s_{i,t}) \geq \max_{a \in \hat{A}_s} b_a \cdot (\hat{r}^s_{a,t} + w^s_{a,t}) - 2^{1-s}\}$
      $s \leftarrow s + 1$
    else
      Select $I_t = argmax_{i \in \hat{A}_s} b_i \cdot (\hat{r}^s_{i,t} + w^s_{i,t})$
  until $I_t$ is selected
```

Algorithm 7 *BaseLinUCB-S: (Adapted from BaseLinUCB by [10])*

```
Inputs: $\alpha \in \mathbb{R}_+$, $\Psi_{i,t} \subseteq \{1, 2, \ldots, t-1\}$
$A_{i,t} \leftarrow I_d + \sum_{\tau \in \Psi_{i,t}} x^\intercal_\tau x_\tau$
$c_{i,t} \leftarrow \sum_{\tau \in \Psi_{i,t}} r_{i,\tau} x_\tau$
$\theta_{i,t} \leftarrow A^{-1}_{i,t} c_{i,t}$
Observe context vector as $x_t \in [0,1]^d$
for $i \in \mathcal{N}$ do
  $w^s_{i,t} \leftarrow \alpha \sqrt{x^\intercal_t A^{-1}_{i,t} x_t}$
  $\hat{r}^s_{i,t} \leftarrow \theta^\intercal_{i,t} x_t$
```

In the next section, we prove the regret bounds on *SupLinUCB-S*, highlighting the steps which differ from regret analysis of *SupLinUCB*.

4.2 Regret Analysis of *SupLinUCB-S*

For convenience, let $s_{i,t} = \sqrt{x_t^\intercal A_{i,t}^{-1} x_t}$, $ucb^s_{i,t} = (\hat{r}^s_{i,t} + w^s_{i,t})$ and $0 < b_i <= 1, \forall i \in \mathcal{N}$. For all round t, stage s and given context x_t, $i_t^*(x_t) = argmax_{i \in \hat{A}_s} b_i \cdot \mathbb{E}[r_{i,t}|x_t]$. The regret analysis is along the similar lines with [10] with changes deemed necessary to incorporate in our setting. In Lemmas 4, 5, and 6, we need to work for each agent as we have different θ_is for different agents.

Lemma 4. (Lemma 2, [10]) For each $s \in [S]$ and $i \in \mathcal{N}$, suppose $\psi^s_{i,t+1} = \psi^s_{i,t} \cup \{t\}$. Then, eigenvalues of $A_{i,t}$ can be arranged so that $\lambda^j_{i,t} \le \lambda^j_{i,t+1}$, for all j and

$$s^2_{i,t} \le 10 \sum_{j=1}^{d} \frac{\lambda^j_{i,t+1} - \lambda^j_{i,t}}{\lambda^j_{i,t}}$$

Lemma 5. (Lemma 3, [10]) Using notation in *BaseLinUCB-S* and assuming $|\psi^s_{i,T+1}| \ge 2$, we have

$$\sum_{t \in \psi^s_{i,T+1}} s_{i,t} \le 5\sqrt{d|\psi^s_{i,T+1}| \ln |\psi^s_{i,T+1}|}$$

Lemma 6. (Lemma 4, [10]). For each $s \in [S]$, each $t \in [T]$, and any fixed sequence of feature vectors x_t, with $t \in \psi^s_{I_t,t}$, the corresponding rewards $r_{I_t,t}$ are independent random variables such that $\mathbb{E}[r_{I_t,t}] = \theta_i^\intercal x_t$.

In our settings, rewards of the arms also have bid component which plays an important role, thus we need the following lemma.

Lemma 7. With probability $1 - \kappa S$, for any $t \in [T]$ and any $s \in [S]$, the following hold:

1. $b_i \cdot ucb^s_{i,t} - 2 \cdot w^s_{i,t} \le b_i \cdot \mathbb{E}[r_{i,t}] \le b_i \cdot ucb^s_{i,t}$ for all i
2. $i_t^*(x_t) \in \hat{A}_s$
3. $b_{i_t^*(x_t)} \cdot \mathbb{E}[r_{i^*(x_t),t}] - b_i \cdot \mathbb{E}[r_{i,t}] \le 2^{3-s}$

Proof. From Lemma 15 of [4], we have, $ucb^s_{i,t} - 2 \cdot w^s_{i,t} \le \mathbb{E}[r_{i,t}] \le ucb^s_{i,t}$ for all i. As $b_i > 0$, after multiplying with b_i inequality still holds, i.e., $b_i \cdot ucb^s_{i,t} - 2 \cdot b_i w^s_{i,t} \le b_i \cdot \mathbb{E}[r_{i,t}] \le b_i \cdot ucb^s_{i,t}$. From our assumption $b_i \le 1$, hence the first part holds.

The lemma trivially holds for $s = 1$. For $s > 1$, $\hat{A}_s \subseteq \hat{A}_{s-1}$ and from the algorithm it is clear that $w^s_{i,t} \le 2^{-(s-1)}$ and $w^s_{i_t^*(x_t)} \le 2^{-(s-1)}$. From part 1 of the lemma and using the above fact, for any $j \in \hat{A}_s$ we have $b_{i_t^*(x_t)} ucb^{(s-1)}_{i_t^*(x_t),t} \ge b_{i_t^*(x_t)} \mathbb{E}[r_{i_t^*(x_t),t}]$ and $b_j \mathbb{E}[r_{j,t}] \ge b_j ucb^s_{j,t} - 2 \cdot 2^{-(s-1)}$. From definition, $b_{i_t^*(x_t)} \mathbb{E}[r_{i_t^*(x_t),t}] \ge b_j \mathbb{E}[r_{j,t}]$. Using this and above inequalities, agent $i_t^*(x_t)$ will belong to $\hat{A}_s, \forall s$ (i.e., will never be eliminated for context x_t) due to the rule defined in Line[14], Algorithm 6. Hence part 2 of the lemma is proved.

From Line[14] Algorithm 6, $b_i ucb^s_{i,t} \ge b_{i_t^*(x_t)} ucb^s_{i_t^*(x_t),t} - 2 \cdot 2^{-(s-1)}$. Using part 1 of the lemma and above inequality the proof of part 3 follows. ■

Lemma 8. (Lemma 6, [10]) For all $s \in [S]$ and $i \in \mathcal{N}$,

$$|\psi^s_{i,T+1}| \le 5 \cdot 2^s (1 + \alpha^2) \sqrt{d|\psi^s_{i,t+1}|}$$

All the above can be summarized as the following theorem.

Theorem 9. *SupLinUCB-S has regret* $O(n^2\sqrt{dT \ln T})$ *with probability at least* $1 - \kappa$ *if it is run with* $\alpha = \sqrt{\frac{1}{2} \ln \frac{2nT}{\kappa}}$.

Proof. The proof is similar to the proof of Theorem 6 of [4] but requires additional terms to consolidate the difference between the algorithms, problem setting, and regret definition. We have restricted the learning during round-robin ordering only Lines[7-9] due to which we have an additional decision rule for agent selection as in Line[17]. Note that this additional rule was not in [10]. Hence the main challenge is to bound the number of rounds agent selection, which is done using this decision rule. (We refer it ψ^s_{ext} in our analysis.)

Let ψ_0 be the set of rounds in which the agent was selected in Lines[10-12]. Let ψ^s_{ext} be the set of rounds agent was selected in Lines[16-17] and $\psi^s_{T+1} = \bigcup_i \psi^s_{i,T+1}$.

Claim 5. At each stage s, $|\psi^s_{est}| <= (n-1) \cdot |\psi^s_{T+1}|$.

For any stage s, let us take case of n consecutive rounds. Let us assume for each of the n rounds, selection of agent is done in Lines[16-17]. But note that selection of agent in this decision block is done if and only if there exist an agent k such that $j \neq k$ (where j is designated agent for the round) and $w^s_{k,\cdot} > 2^{-s}$. But after n consecutive rounds each agent has got its designated round once (Line[4]). Hence, if for some agent k, $w^s_{k,\cdot} > 2^{-s}$, then this agent k should be selected on its designated round. Hence our assumption of selection of agent in Lines[16-17] for n consecutive round is wrong. From above reasoning, it is clear that at least for one round out of n rounds, one of the agent must be selected at its designated round. Hence, at most for $n-1$ rounds agent is selected in Line[16-17] out of n rounds, until condition in Line[10] is achieved. Thus, we can say that at each stage s, $|\psi^s_{est}| <= (n-1) \cdot |\psi^s_{T+1}|$.
As $2^{-S} \leq 1/\sqrt{T}$, we have $\{1, \dots, T\} = \psi_0 \cup \bigcup_s \psi^s_{T+1} \bigcup_s \cup \psi^s_{est}$. Using the claim and the lemmas,

$$\begin{aligned}
\mathbb{R}_T =& \sum_{t=1}^{T} [b_{i^*_t(x_t)} \mathbb{E}[r_{i^*_t(x_t),t}] - b_{I_t} \mathbb{E}[r_{I_t,t}]] \\
=& \sum_{t \in \psi_0} [b_{i^*_t(x_t)} \mathbb{E}[r_{i^*_t(x_t),t}] - b_{I_t} \mathbb{E}[r_{I_t,t}]] \\
&+ \sum_{s=1}^{S} \Bigg[\sum_{t \in \psi^s_{T+1}} [b_{i^*_t(x_t)} \mathbb{E}[r_{i^*_t(x_t),t}] - b_{I_t} \mathbb{E}[r_{I_t,t}]] \\
&+ \sum_{t \in \psi^s_{est}} [b_{i^*_t(x_t)} \mathbb{E}[r_{i^*_t(x_t),t}] - b_{I_t} \mathbb{E}[r_{I_t,t}]] \Bigg] \\
\leq& \frac{2}{\sqrt{T}} |\psi_0| + \sum_{s=1}^{S} n \cdot \sum_{t \in \psi^s_{T+1}} [b_{i^*_t(x_t)} \mathbb{E}[r_{i^*_t(x_t),t}] - b_{I_t} \mathbb{E}[r_{I_t,t}]] \\
=& \frac{2}{\sqrt{T}} |\psi_0| + \sum_{s=1}^{S} n \sum_{i}^{|\mathcal{N}|} \sum_{t \in \psi^s_{i,T+1}} [b_{i^*_t(x_t)} \mathbb{E}[r_{i^*_t(x_t),t}] - b_{I_t} \mathbb{E}[r_{I_t,t}]] \\
\leq& \frac{2}{\sqrt{T}} |\psi_0| + n \sum_{i}^{|\mathcal{N}|} \sum_{s=1}^{S} 8 \cdot 2^{-s} \cdot |\psi^s_{i,T+1}| \\
\leq& \frac{2}{\sqrt{T}} |\psi_0| + n \sum_{i}^{|\mathcal{N}|} \sum_{s=1}^{S} 40 \cdot (1 + \ln(2Tn/\kappa)) \cdot \sqrt{d |\psi^s_{i,T+1}|} \\
\leq& \frac{2}{\sqrt{T}} |\psi_0| + n \sum_{i}^{|\mathcal{N}|} 40 \cdot (1 + \ln(2Tn/\kappa) \cdot \sqrt{STd} \\
\leq& 2\sqrt{T} + 40n^2 \cdot (1 + \ln(2Tn/\kappa) \cdot \sqrt{STd}
\end{aligned}$$

■

Theorem 10. *The allocation rule induced by SupLinUCB-S (Algorithm 6) is ex-post monotone.*

Proof. The allocation rules Algorithm 3 and Algorithm 6 are similar in the way it learns and eliminates agents. Both algorithms learn only when a designated agent is selected based on round-robin ordering, and the elimination is based on bids, *UCB* and *LCB* estimates. The difference between elimination rules is the need for the width of an agent to reach threshold 2^{-s} at stage s. Due to the above similarities, the proof follows on the similar lines of the proof of Theorem 2, and hence we skip it for ease of exposition. ■

M-SupLinUCB-S

The mechanism *M-SupLinUCB-S* follows the same structure as that of mechanism *M-ELinUCB-SB*. The only change is that the allocation rule $\mathcal{A}$ is given by *SupLinUCB-S* (Algorithm 6).

4.3 *M-SupLinUCB-S*: Game-Theoretic Analysis

Theorem 11. M-SupLinUCB-S *is ex-post incentive compatible (EPIC) and ex-post individually rational (EPIR) mechanism.*

Proof. The result follows from Theorem 1 and by ex-post monotonicity of $\mathcal{A}$ defined in Algorithm 6. ■

5 Experimental Analysis

5.1 Data Preparation

Our simulated data follow the structure and information availability found in a real-world system. Typically, a center has access to user features such as gender, age, geographic features, device model, and behavioral categories (which summarizes the user's past preferences), which constitute the context. Note that each of the stated features can be discretized. Considering the above facts, we created the corpus of users χ: with $d = 4$, for each feature, we randomly select 4 possible different values from 0 to 100, and then by taking all possible combination of features, we generated random 256 (4^4) users. We normalize each $x \in \chi$ such that $x \in [0,1]^d$, with $||x||_2 = 1$ and store these normalized contexts as a database χ_{db}.

We then select x_t uniformly at random from the context database χ_{db} for each round to generate a stochastic context. For each agent (advertiser), we generate $\theta_i \sim U([0,1]^d)$ and then normalize it s.t. $||\theta_i||_2 = 1$. To simulate the clicks, at round t with the sampled x_t, we generate a click $r_{i,t}$ from Bernoulli distribution with parameter $\theta_i^T x_t$. We conduct experiments for 40 iterations, and for each iteration, we randomly generate a sequence of contexts from χ_{db} for $T = 10^6$ rounds. We generate valuation of agent i for a click to be v_i sampled from uniform distribution $[0,1]$ and assume the agents bid truthfully; due to truthfulness properties of our mechanisms.

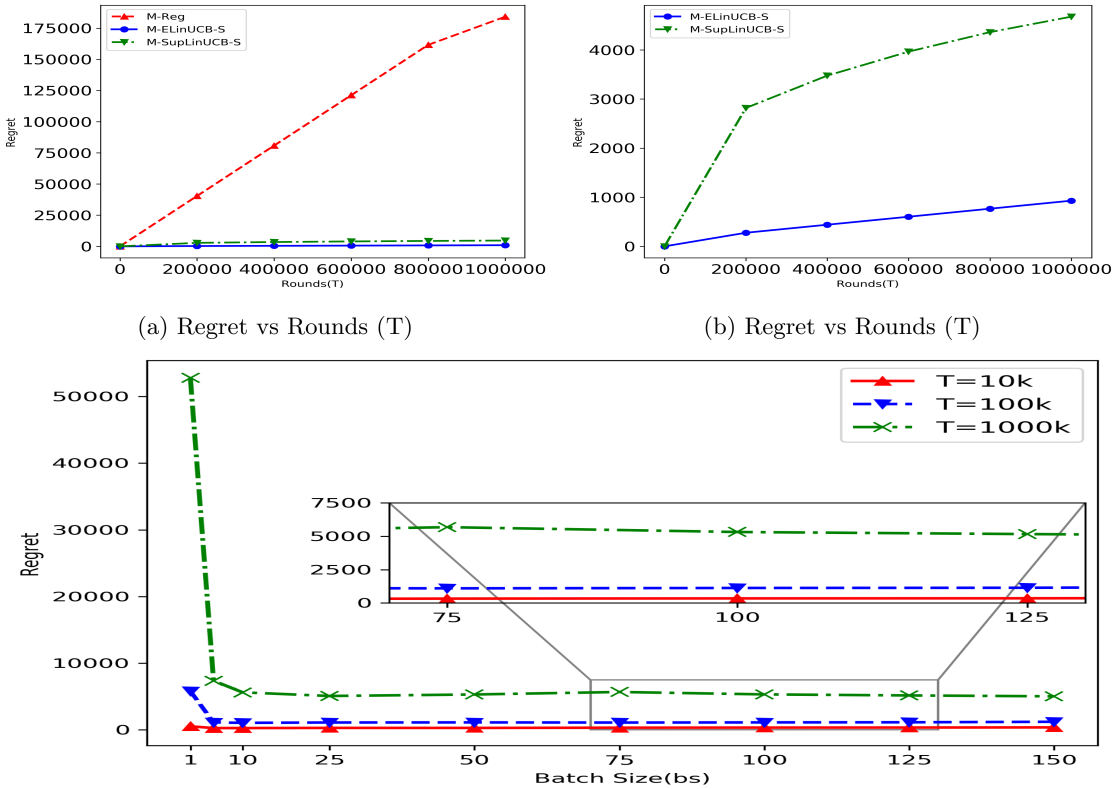
5.2 Results and Comparison

From our experimentation, we found learning parameter $\alpha = 1$ and batch size $bs = 100$ to be suitable for *M-ELinUCB-SB*. The metric of comparison between the mechanisms is regret, which is averaged over 40 iterations. Fig.1a compares the regret of *M-Reg*, *M-ELinUCB-SB*, and *M-SupLinUCB-SB* for $n = 7$. When $n > 7$, *M-Reg* becomes infeasible as the number of exploration rounds λ exceeds the total number of rounds T, for $T = 10^6$. In terms of regret, it is evident that both the mechanisms *M-ELinUCB-SB* and *M-SupLinUCB-SB* outperform *M-Reg* by a very large margin. Fig.1b highlights difference in experimental regret *M-ELinUCB-SB* and *M-SupLinUCB-SB* (it is zoomed version from Fig. 1a). We can see that *M-ELinUCB-SB* experimentally performs approximately 5 times better than *M-SupLinUCB-SB*; albeit the results are validated only on the randomly generated 256 contexts (χ_{db}).

Though in theory, *M-ELinUCB-SB* has the worst regret, from simulations, the slope being very small, for reasonable values of T it outperforms *M-Reg*. Our experiments show that *M-ELinUCB-SB* and *M-SupLinUCB-SB*, both have nearly negligible regret as compared to *M-Reg*. Fig.1c compares the regret incurred by *M-ELinUCB-SB* with varying batch size $bs \in \{1, 5, 10, 25, 50, 75, 100, 125, 150\}$ for $T \in \{10k, 100k, 1000k\}$. From the figure, it is easy to infer the significant improvement in regret when we move from batch size $bs = 1$ to greater batch sizes. One may need to tune bs based on different experimental setup.

6 Conclusion

We believe that ours is the first attempt to design a non-exploration-separated ConMAB mechanism. We focused on designing ConMAB mechanisms for sponsored search auction (SSA). For a single-slot, we first designed LinUCB-based ex-post monotone allocation rule *ELinUCB-S*. We show that the introduction of batch size in *ELinUCB-S* significantly improves the regret while satisfying the ex-post monotone property. With this observation, we present *ELinUCB-SB*. Through simulations, we see that in practice, it performs better for regret; however, theoretically, it may



(a) Regret vs Rounds (T)

(b) Regret vs Rounds (T)

(c) Regret vs Batch Size (bs)

Figure 1: Regret comparisons

incur linear regret on carefully chosen contexts. To achieve sub-linear regret, we proposed another ex-post monotone allocation rule, *SupLinUCB-S*. We further extended these allocation rules to mechanisms, *M-ELinUCB-SB*, and *M-SupLinUCB-SB* satisfying EPIC and EPIR properties. We showed our mechanism performs significantly better than the existing mechanism *M-Reg* [12]. In summary, *M-SupLinUCB-S* is novel, truthful ConMAB mechanism that outperforms *M-Reg* in every aspect.

Although our mechanisms are randomized, they are game theoretically sound and scalable as compared to *M-Reg*. Further, in terms of regret, *M-ELinUCB-SB* and *M-SupLinUCB-S* outperforms *M-Reg* in experiments and theoretically *M-SupLinUCB-S* matches the regret in non-strategic setting up to a factor of $O(n)$ which is the price of truthfulness. Though we presented mechanisms for single slot SSA, they can be generalized to multi-slot SSA using similar techniques. Along with SSA, this work can form a baseline for other applications such as crowdsourcing [9], smart grids [15], where similar setting arises to learn the stochastic parameters in the presence of strategic agents.

References

[1] Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. In *Proceedings of the 24th International Conference on Neural Information Processing Systems*, NIPS'11, pages 2312–2320, USA, 2011. Curran Associates Inc.

[2] Gagan Aggarwal, Ashish Goel, and Rajeev Motwani. Truthful auctions for pricing search keywords. In *Proceedings of the 7th ACM Conference on Electronic Commerce*, EC '06, pages 1–7, New York, NY, USA, 2006. ACM.

[3] A. Archer and É. Tardos. Truthful mechanisms for one-parameter agents. In *Proceedings of the 42Nd IEEE Symposium on Foundations of Computer Science*, FOCS '01, pages 482–, Washington, DC, USA, 2001. IEEE Computer Society.

[4] Peter Auer. Using confidence bounds for exploitation-exploration trade-offs. *J. Mach. Learn. Res.*, 3:397–422, March 2003.

[5] Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2-3):235–256, 2002.

[6] Moshe Babaioff, Robert D. Kleinberg, and Aleksandrs Slivkins. Truthful mechanisms with implicit payment computation. *J. ACM*, 62(2):10:1–10:37, May 2015.

[7] Moshe Babaioff, Yogeshwer Sharma, and Aleksandrs Slivkins. Characterizing truthful multi-armed bandit mechanisms: Extended abstract. In *Proceedings of the 10th ACM Conference on Electronic Commerce*, EC '09, pages 79–88, New York, NY, USA, 2009. ACM.

[8] Dirk Bergemann and Juuso Välimäki. The dynamic pivot mechanism. *Econometrica*, 78(2):771–789, 2010.

[9] Arpita Biswas, Shweta Jain, Debmalya Mandal, and Y. Narahari. A truthful budget feasible multi-armed bandit mechanism for crowdsourcing time critical tasks. In *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*, AAMAS '15, pages 1101–1109, Richland, SC, 2015. International Foundation for Autonomous Agents and Multiagent Systems.

[10] Wei Chu, Lihong Li, Lev Reyzin, and Robert E. Schapire. Contextual bandits with linear payoff functions. *Journal of Machine Learning Research - Proceedings Track*, 15:208–214, 01 2011.

[11] Nikhil R. Devanur and Sham M. Kakade. The price of truthfulness for pay-per-click auctions. In *Proceedings of the 10th ACM Conference on Electronic Commerce*, EC '09, pages 99–106, New York, NY, USA, 2009. ACM.

[12] Nicola Gatti, Alessandro Lazaric, and Francesco Trovò. A truthful learning mechanism for contextual multi-slot sponsored search auctions with externalities. In *Proceedings of the 13th ACM Conference on Electronic Commerce*, EC '12, pages 605–622, New York, NY, USA, 2012. ACM.

[13] IAB. Iab internet advertising revenue report. 2018 first half-year results., 2018.

[14] Shweta Jain, Sujit Gujar, Satyanath Bhat, Onno Zoeter, and Y Narahari. A quality assuring, cost optimal multi-armed bandit mechanism for expertsourcing. *Artificial Intelligence*, 254:44–63, 2018.

[15] Shweta Jain, Balakrishnan Narayanaswamy, and Y. Narahari. A multiarmed bandit incentive mechanism for crowdsourcing demand response in smart grids. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, AAAI'14, pages 721–727. AAAI Press, 2014.

[16] T Lai. Asymptotically efficient adaptive allocation rules. *Advances in Applied Mathematics*, 6:4–22, 1985.

[17] John Langford and Tong Zhang. The epoch-greedy algorithm for contextual multi-armed bandits. In *Proceedings of the 20th International Conference on Neural Information Processing Systems*, NIPS'07, pages 817–824, USA, 2007. Curran Associates Inc.

[18] Lihong Li, Wei Chu, John Langford, and Robert E. Schapire. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th International Conference on World Wide Web*, WWW '10, pages 661–670, New York, NY, USA, 2010. ACM.

[19] Roger B Myerson. Optimal auction design. *Mathematics of operations research*, 6(1):58–73, 1981.

[20] Noam Nisan, Tim Roughgarden, Eva Tardos, and Vijay V. Vazirani. *Algorithmic Game Theory*. Cambridge University Press, New York, NY, USA, 2007.