Multi-Armed Bandits with Cost Subsidy

Deeksha Sinha MIT/Facebook
Karthik Abinav Sankararaman Facebook
Abbas Kazerouni Facebook
Vashist Avadhanula Facebook

Abstract

In this paper, we consider a novel variant of the multi-armed bandit (MAB) problem, *MAB with cost subsidy*, which models many real-life applications where the learning agent has to pay to select an arm and is concerned about optimizing cumulative costs and rewards. We present two applications, *intelligent SMS routing problem* and *ad audience optimization problem* faced by several businesses (especially online platforms), and show how our problem uniquely captures key features of these applications. We show that naive generalizations of existing MAB algorithms like Upper Confidence Bound and Thompson Sampling do not perform well for this problem. We then establish a fundamental lower bound on the performance of any online learning algorithm for this problem, highlighting the hardness of our problem in comparison to the classical MAB problem. We also present a simple variant of *explore-then-commit* and establish near-optimal regret bounds for this algorithm. Lastly, we perform extensive numerical simulations to understand the behavior of a suite of algorithms for various instances and recommend a practical guide to employ different algorithms.

1 Introduction

In the traditional (stochastic) MAB problem (Robbins (1952)), the learning agent has access to a set of K actions (arms) with unknown but fixed reward distributions and has to repeatedly select an arm to maximize the cumulative reward. Here, the challenge is designing a policy that balances the tension between acquiring information about arms with little historical observations and exploiting the most rewarding arm based on existing information. The aforementioned exploration-exploitation trade-off has been extensively studied, leading to some simple but extremely effective algorithms like Upper Confidence Bound (Auer et al. (2002a)) and Thompson Sampling (Thompson (1933); Agrawal and Goyal (2017a)), which have been further generalized and applied in a wide range of application domains, including online advertising (Langford and Zhang (2008); Cesa-Bianchi et al. (2014), Oliver and Li (2011)), recommendation systems (Li et al. (2015, 2011); Agrawal et al. (2016)), social networks and crowd sourcing (Anandkumar et al. (2011); Sankararaman et al. (2019), Slivkins and Vaughan (2014)); see Bubeck and Cesa-Bianchi (2012) and Slivkins (2019) for a detailed review. However, most of these approaches cannot be generalized to settings involving multiple metrics (for example, reward and cost) when the underlying trade-offs between these metrics are not known a priori.

In many real-world applications of MAB, some of which we will elaborate below, it is common for the agent to incur costs to play an arm, with *high performing arms* costing more. Though one can model this in the traditional MAB framework by considering cost subtracted from the reward as the modified objective, such a modification is not always meaningful, particularly in settings where the reward and cost associated with an arm represent different quantities (for example, click rate and cost of an ad). In such problems, it is natural for the learning agent to *optimize* for both the metrics, typically avoiding incurring exorbitant costs for a marginal increase in cumulative reward. Motivated by the aforementioned scenario, in this paper, we consider a variant of the MAB problem, where the agent is not only concerned about balancing the exploration-exploitation trade-offs to maximize the cumulative reward but also balance the trade-offs associated with multiple objectives that are intrinsic to several practical applications. More specifically, in this work, we study a stylized problem, where to manage costs, the agent is willing to tolerate a small loss from

Proceedings of the 24th International Conference on Artificial Intelligence and Statistics (AISTATS) 2021, San Diego, California, USA. PMLR: Volume 130. Copyright 2021 by the author(s).

the *highest reward*, measured as the reward that the traditional MAB problem could obtain in the absence of costs. We refer to this problem as MAB problem with a cost subsidy (see Section 1.1 for exact problem formulation), where the subsidy refers to the amount of reward the learning agent is willing to forgo to improve costs. Before explaining our problem and technical contributions in detail, we will elaborate on the applications that motivate this problem.

Intelligent SMS Routing. Many businesses such as banks, delivery services, airlines, hotels, and various online platforms send SMSes (text messages) to their users for various reasons, including two-factor authentication, order confirmations, appointment reminders, transaction alerts, and as a direct marketing line (see Twilio and Uber (2020)). These text messages, referred to as Application-to-Person (A2P) messages, constitute a significant portion of all text messages sent through cellular networks today. In fact, A2P messages are forecasted to be a $86.3 billion business by 2025 (MarketWatch (2020)).

To deliver these messages, businesses typically enlist the support of telecom aggregators, who have private agreements with mobile operators. Each aggregator offers a unique combination of quality, as measured by the fraction of text messages successfully delivered by them and price per message. Surprisingly, it is common for delivery rates of text messages not to be very high (see Canlas et al. (2010); Meng et al. (2007); Zerfos et al. (2006); Osunade and Nurudeen for QoS analysis in different geographies) and for aggregator's quality to fluctuate with time due to various reasons ranging from network outage to traffic congestion. Therefore, the platform's problem of balancing the tension between inferring aggregator's quality through exploration and exploiting the current *best performing aggregator* to maximize the number of messages delivered to users leads to a standard MAB formulation. However, given the large volume of messages that need to be dispatched, an MAB based solution that focuses exclusively on the quality of the aggregator could result in exorbitant spending for the business. A survey of businesses shows that the number of text messages they are willing to send will significantly drop if the cost per SMS is increased by a few cents per SMS (Ovum (2017)). Moreover, in many situations, platforms have backup communication channels such as email-based authentication or notifications via in-app/website features. Though not as effective as a text message in terms of read rate, it can be used if guaranteeing the text message delivery proves to be very costly. Therefore, it is natural for businesses to prefer an aggregator with lower costs as long as their quality is comparable to the aggregator with the best quality.

Ad-audience Optimization. We now describe another real-world application in the context of online advertisements. Many advertisers (especially small-to-medium scale businesses) have increasingly embraced the notion of *auto-targeting* where they let the advertising platform identify a *high-quality* audience group (*e.g.,* Koningstein (2006); Amazon (2019); Facebook (2016); Google (2014)). To enable this, the platform explores the audience-space to identify *cheaper* opportunities that also provide high click-through-rate (ctr) and conversion rate. Here, different audience groups can have different yields, i.e., quality (CTR/conversion rate) for a specific ad. However, it may require vastly different bids to reach different audiences due to auction overlap with other ad campaigns with smaller audience targeting. Thus, the algorithm is faced with a similar trade-off; as long as a particular audience-group gives a high-yield, the goal is to find the cheapest one.

We now present a novel formulation of a multi-armed bandit problem that captures the key features of these applications. Our goal is to develop a cost-sensitive MAB algorithm that balances both the exploration-exploitation trade-offs as well as the tension between conflicting metrics in a multi-objective setting.

1.1 Problem Formulation

To formally state our problem, given an instance I, in every round $t \in [T]$ the agent chooses an arm $i \in [K]$ and realizes a reward r_t, sampled independently from a fixed, but unknown distribution $\mathcal{F}_i$ with mean μ_i (or μ_i^I) and incurs a cost c_i (or c_i^I), which is known a priori. Here, to manage costs, we allow the agent to be agnostic between arms, whose expected reward is greater than $1-\alpha$ fraction of the highest expected reward, for a fixed and known value of α, which we refer to as the *subsidy factor.* The agent's objective is to learn and pull the cheapest arm among these *high*-quality arms as frequently as possible.

More specifically, let m_* denote the arm with highest expected mean, *i.e.,* $m_* = \text{argmax}_{i\in[K]}\ \mu_i$, and $\mathcal{C}_*$ be the set of arms whose expected reward is within $1-\alpha$ factor of the highest expected reward, *i.e.,* $\mathcal{C}_* = \{i \in [K] \mid \mu_i \geq (1-\alpha)\mu_{m_*}\}$. We refer to the quantity $(1-\alpha)\mu_{m_*}$ as the *smallest tolerated reward.* Without loss of generality, we assume the reward distribution has support $[0, 1]$. The agent's goal is to design a policy (algorithm) π that will learn the cheapest arm whose expected reward is at least as large as the smallest tolerated reward. In other words, the agent needs to learn the identity and simultaneously maximize the number of plays of arm $i_* = \text{argmin}_{i\in\mathcal{C}_*}\ c_i$. Since in the SMS application, the reward is the quality of the chosen aggregator, we will use the terms reward and quality interchangeably.

To measure the performance of any policy π, we propose two notions of regret - quality and cost regret, with the agent's goal being minimizing both of them:

$$\begin{aligned}&\mathsf{Quality\_Reg}_\pi(T,\alpha,\boldsymbol{\mu},\boldsymbol{c})\\&\quad=\mathbf{E}\left[\sum_{t=1}^{T}\max\{(1-\alpha)\mu_{m_*}-\mu_{\pi_t},0\}\right],\\&\mathsf{Cost\_Reg}_\pi(T,\alpha,\boldsymbol{\mu},\boldsymbol{c})\\&\quad=\mathbf{E}\left[\sum_{t=1}^{T}\max\{c_{\pi_t}-c_{i_*},0\}\right],\end{aligned}\tag{1}$$

where $\boldsymbol{c}=(c_1,\cdots c_K),\boldsymbol{\mu}=(\mu_1,\cdots\mu_K)$ and the expectation is over the randomness in policy π. Equivalently, the cost and quality regret of policy π on an instance I of the problem is denoted as $\mathsf{Quality\_Reg}_\pi(T,\alpha,I)$ and $\mathsf{Cost\_Reg}_\pi(T,\alpha,I)$ where the instance is defined by the distributions of the reward and cost of each arm. The objective then is to design a policy that simultaneously minimizes both the cost and quality regret for all possible choices of $\boldsymbol{\mu}$ and $\boldsymbol{c}$ (equivalently all instances I).

Choice of Objective Function. Note that a parametrized linear combination of reward and cost metrics, i.e., $\mu-\lambda c$ for an appropriately chosen λ is a popular approach to balance cost-reward trade-off. From an application stand-point, there are two important considerations that favor using our objective instead of a linear combination of reward and cost metrics. First, in a real-world system, we need explicit control over the parameter α that is not instance-dependent to understand and defend the trade-off between the various objectives. From a product manager's perspective, the *indifference* among the arms within $(1-\alpha)$ of the arm with the highest mean reward is a transparent and interpretable compromise to lower costs. Second, for the intelligent SMS routing application discussed earlier, different sets of aggregators operate in different regions. Thus, separate λ values would need to be configured for each region, making the process cumbersome.

Further note that, the setting considered in this paper is not equivalent to taking a parametrized linear combination of reward and cost metrics. In particular, for any specified subsidy factor α, the value λ required in the linear objective function, for i_* to be the optimal arm would depend on the cost and reward distributions of the arms. Therefore, using a single value of λ and relying on standard MAB algorithms would not lead to the desired outcome for our problem. Moreover, in our setting, the cost and quality regret measures are defined with respect to different benchmark arms (i_* and m_* respectively). The reward of the played arm is compared to $(1-\alpha)$ times the reward of the highest mean reward arm and not directly the mean reward of the arm m_* as would be the case in a single objective based formulation.

1.2 Related Work

Our problem is closely related to the MAB with multiple objectives line of work, which has attracted considerable attention in recent times. The existing literature on multi-objective MAB can be broadly classified into the following categories.

Bandits with Knapsacks (BwK). Bandits with knapsacks (BwK), introduced in the seminal work of Badanidiyuru et al. (2018) is a general framework that considers the standard MAB problem under the presence of additional budget/resource constraints. The BwK problem encapsulates a large number of constrained bandit problems that naturally arise in many application domains, including dynamic pricing, auction bidding, routing, and scheduling (see Tran-Thanh et al. (2012); Agrawal and Devanur (2014); Immorlica et al. (2019)). In this formulation, the agent has access to a set of d finite resources and K arms, each associated with a reward distribution. Upon playing arm a at time t, the agent realizes a reward of r_t and incurs a penalty of $c_t^{(i)}$ for resource i, all drawn from a fixed, but unknown distribution corresponding to the arm. The objective of the agent is to maximize the cumulative reward before one of the resources is completely depleted. Although appealing in many applications, BwK formulation requires *hard* constraint on resources (cost in our setting) and hence, cannot be easily generalized to our problem. In particular, in the cost subsidized MAB problem, the equivalent budget limits depend on the problem instance and therefore cannot be determined a priori.

Pareto Optimality and Composite Objective. The second formulation focuses on identifying Pareto optimal alternatives and uniformly choosing among these options (see Drugan and Nowe (2013); Yahyaa et al. (2014); Paria et al. (2018); Yahyaa and Manderick (2015)). These approaches do not apply to our problem. Some of the Pareto alternatives could have extreme values for one of the metrics, for example, having a meager cost and low quality or extremely high cost and quality, making them undesirable for the applications discussed earlier. Closely related to this line of work is the set of works that focus on a composite objective by appropriately weighting the different metrics (see Paria et al. (2018); Yahyaa and Manderick (2015)). Such formulations also do not immediately apply to our problem. In the SMS and ad applications discussed earlier, it is not acceptable to drop the quality beyond the allowed level irrespective of the cost savings we could obtain. Furthermore, in the SMS application, the trade-offs between quality and costs could vary from region to region, making it hard to identify a good set of weights for the composite objective (see Section 1.1).

Conservative Bandits and Bandits with Safety Constraints. Two other lines of work that are recently receiving increased attention, particularly from practitioners, are *bandits with safety constraints* (see Daulton et al. (2019); Amani et al. (2020); Galichet et al. (2013)) and *conservative bandits* (see Wu et al. (2016); Kazerouni et al. (2017)). In both these formulation, the algorithm chooses one of the arms and receives a reward and a cost associated with it. The goal of the algorithms is to maximize the total reward obtained while ensuring that either the chosen arm is within a pre-specified threshold (when costs of arms are unknown a priori) or the reward of the arm is at least a specified fraction of a known benchmark arm. Neither of these models exactly captures the requirements of our applications: a) we do not have a hard constraint on the acceptable cost of a pulled arm. In particular, choosing low-quality aggregators to avoid high costs (even for a few rounds) could be disastrous since it leads to bad user experience on the platform and eventual churn, and b) the equivalent benchmark arm in our case, i.e., the arm with the highest mean reward is not known a priori.

Best Arm Identification. Apart from the closely related works mentioned above, our problem of identifying the *cheapest* arm whose expected reward is within an acceptable margin from the highest reward can be formulated as a stylized version of the *best-arm identification problem* (Katz-Samuels and Scott (2019); Jamieson and Nowak (2014); Chen et al. (2014); Cao et al. (2015); Chen et al. (2016)). However, in many settings and particularly applications discussed earlier, the agent's objective is optimizing cumulative reward and not just identifying the *best arm.*

1.3 Our Contributions

Novel Problem Formulation. In this work, we propose a stylized model, *MAB with a cost subsidy*, and introduce new performance metrics that uniquely capture the salient features of many real-world online learning problems involving multiple objectives. For this problem, we first show that naive generalization of popular algorithms like Upper Confidence Bound (UCB) and Thompson Sampling (TS) could lead to poor performance on the metrics. In particular, we show that the naive generalization of TS for this problem would lead to a linear cost regret for some problem instances.

Lower Bound. We establish a fundamental limit on the performance of *any online algorithm* for our problem. More specifically, we show that any online learning algorithm will incur a regret of $\Omega(K^{1/3}T^{2/3})$ on either the cost or the quality metric (refer to (1)), further establishing the *hardness of our problem* relative to the standard MAB problem, for which it is possible to design algorithms that achieve worst-case regret bound of $\tilde{O}(\sqrt{KT})$. We introduce a novel reduction technique to derive the above lower bound, which is of independent interest.

Cost Subsidized Explore-Then-Commit. We present a simple algorithm based on the *explore-then-commit* (ETC) principle and show that it achieves near-optimal performance guarantees. In particular, we establish that our algorithm achieves a worst-case bound of $O(K^{1/3}T^{2/3}\sqrt{\log T})$ for both cost and quality regret. A key challenge in generalizing the ETC algorithm for this problem arises from having to balance between two asymmetric objectives. We also discuss generalizations of the algorithm for settings where the cost of the arms is not known a priori. Furthermore, we consider a special scenario of bounded costs, where naive generalizations of TS and UCB work reasonably well and establish worst-case regret bounds.

Numerical Simulation. Lastly, we perform extensive simulations to understand various regimes of the problem parameters and compare different algorithms. More specifically, we consider scenarios where naive generalizations of UCB and TS, which have been adapted in real-life implementations (see Daulton et al. (2019)) perform well and settings where they perform poorly, which should be of interest to practitioners.

1.4 Outline

The rest of this paper is structured as follows. In Section 2, we show that the naive generalization to TS or UCB algorithms performs poorly, and in Section 3, we establish lower bounds on the performance of any algorithm for MAB with cost subsidy problem. In Section 4, we present a variation of the ETC algorithm and show that it achieves a near-optimal regret bound of $\tilde{O}(K^{1/3}T^{2/3})$ for both the metrics. In section 5, we show that with additional assumptions, it is possible to show improved performance bounds for naive generalization of existing algorithms. Finally, in section 6, we perform numerical simulations to explore various regimes of the instance-space.

2 Performance of Existing MAB Algorithms

In this section, we consider a natural extension of two popular MAB algorithms, TS and UCB, for our problem and show that such adaptations perform poorly. This highlights the challenges involved in developing good algorithms for the MAB problem with cost subsidy. In particular, we establish theoretically that for some problem instances, the TS variant incurs a linear cost regret and observe similar performance for the UCB variant empirically. Our key focus on TS in this section is primarily motivated by the superior performance observed over a stream of recent papers in the context of TS versus more traditional approaches such as UCB (see Scott (2010); Oliver and Li (2011); May et al. (2012); Agrawal et al. (2017)).

We present the details of TS and UCB adaptations in Algorithm 1, which we will refer to as Cost-Subsidized TS(CS-TS) and Cost-Subsidized UCB(CS-UCB) respectively. These extensions are inspired by Daulton et al. (2019), which demonstrates empirical efficacy on a related (but different) problem. Briefly, in the CS-TS(CS-UCB) variation, we follow the standard TS (UCB) algorithm and obtain a quality score which is a sample from the posterior distribution (upper confidence bound) for each arm. We then construct a feasible set of arms whose quality scores are greater than $1-\alpha$ fraction of the highest quality score. Finally, we pull the cheapest arm among the feasible set of arms.

We will now show that CS-TS with Gaussian priors and posteriors (i.e., Gaussian distribution with mean $\hat{\mu}_i(t)$ and variance $1/T_i(t)$) described in Algorithm 1 incurs a linear cost regret in the worst case. More precisely, we prove the following result.

Theorem 1. *For any given K, α, T there exists an instance ϕ of problem such that* $\mathsf{Quality\_Reg}_{\mathsf{CS-TS}}(T,\alpha,\phi) + \mathsf{Cost\_Reg}_{\mathsf{CS-TS}}(T,\alpha,\phi)$ *is* $\Omega(T)$.

Proof Sketch. The proof closely follows the lower bound argument in Agrawal and Goyal (2017b). We briefly describe the intuition behind the result. Consider a scenario where the highest reward arm is an expensive arm while all other arms are cheap and have rewards marginally above the smallest tolerated reward. In the traditional MAB problem, the anti-concentration property of the Gaussian distribution (see Agrawal and Goyal (2017b)) ensures samples from the *good arm* would be large enough with sufficient frequency, ensuring appropriate exploration and good performance. However, in our problem, the anti-concentration property would result in playing the expensive arm too often since the difference in the mean qualities is small, incurring a linear cost regret while achieving zero quality regret. A complete proof of the theorem is provided in Appendix C. □

The algorithm's poor performance is not limited only to the above instance and usage of Gaussian prior. More generally, the CS-TS and CS-UCB algorithms seem to perform poorly whenever the mean reward of the optimal arm is very close to the smallest tolerated reward. We illustrate this through another empirical example. Consider the following instance with two arms, each having Bernoulli rewards and $T = 10,000$. The costs of the two arms are $c_1 = 0$ and $c_2 = 1$. The expected qualities are $\mu_1 = 0.5(1-\alpha) + 1/\sqrt{T}, \mu_2 = 0.5$ with $\alpha = 0.1$. The prior of the mean reward of both the arms is a Beta(1,1) distribution. Here, the quality regret will be zero irrespective of which arm is played. But both CS-TS and CS-UCB incur significant cost regret as shown in Figure 1. (In the figure, we also plot the performance of the key algorithm we propose in the paper (Algorithm 2) and note that it has much superior performance compared to CS-TS and CS-UCB.)

Algorithm 1: Cost Subsidized TS and UCB Algorithms

```
Result: Arm $I_t$ to be pulled in each round $t \in [T]$
Input  : $T, K$, prior distribution for mean
         rewards of all arms $\{\nu_i\}_{i=1}^{K}$, reward
         likelihood function $\{L_i\}_{i=1}^{K}$
$T_i(1) = 0 \ \forall i \in [K]$;
for $t \in [K]$ do
    $I_t = t$;
    Play arm $I_t$ and observe reward $r_t$;
    $T_i(t+1) = T_i(t) + \mathbf{1}\{I_t = i\} \ \forall i \in [K]$;
end
for $t \in [K+1, T]$ do
    for $i \in [K]$ do
        $\hat{\mu}_i(t) \leftarrow \left(\sum_{\tau=1}^{t-1} r_\tau \mathbf{I}\{I_\tau = i\}\right) / T_i(t)$;
        $\beta_i(t) \leftarrow \sqrt{(2\log T)\,/T_i(t)}$;
        UCB: $\mu_i^{score}(t) \leftarrow \min\{\hat{\mu}_i(t) + \beta_i(t), 1\}$;
        TS: Sample $\mu_i^{score}(t)$ from the posterior
          distribution of arm $i$,
          $\nu_i\big(\cdot \,|\{r_s\}_{s\in\{1,2,\cdots t-1\} \text{ s.t. } I_s=i}, L_i\big)$;
    end
    $m_t = \arg\max_i \mu_i^{score}(t)$;
    $Feas(t) = \{i : \mu_i^{score}(t) - (1-\alpha)\mu_{m_t}^{score} \geq 0\}$;
    $I_t = \arg\min_{i \in Feas(t)} c_i$;
    Play arm $I_t$ and observe reward $r_t$;
    $T_i(t+1) = T_i(t) + \mathbf{1}\{I_t = i\} \ \forall i \in [K]$;
end
```

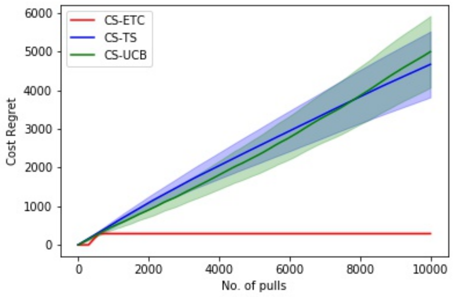

Figure 1: Cost regret of various algorithms for an instance where the mean reward of the optimal arm is very close to the smallest tolerated reward. CS-TS and CS-UCB incur significant regret. But CS-ETC attains low cost regret. The width of the error bands is two standard deviations based on 50 runs of the simulation.

3 Lower Bound

In this section, we establish that any policy must incur a regret of $\Omega(K^{1/3}T^{2/3})$ on at least one of the regret metrics. More precisely, we prove the following result.

Theorem 2. *For any given α, K, T and (possibly randomized) policy π, there exists an instance ϕ of problem* (1) *with $K+1$ arms such that $\mathsf{Quality\_Reg}_\pi(T,\alpha,\phi) + \mathsf{Cost\_Reg}_\pi(T,\alpha,\phi)$ is $\Omega\left((1-\alpha)^2 K^{\frac{1}{3}} T^{\frac{2}{3}}\right)$ when $0 \le \alpha \le 1$ and $1 \le K \le T$.*

3.1 Proof Overview

We consider the following families of instances to establish the lower bound. We first prove the result for $\alpha = 0$ and then establish a reduction for $\alpha = \theta$ for $0 \le \theta < 1$, to the special case of $\alpha = 0$.

Definition 1 (Family of instances $\Phi_{\theta,p,\epsilon}$)**.** *Define a family of instances $\Phi_{\theta,p,\epsilon}$ consisting of instances $\Phi^0_{\theta,p,\epsilon}, \Phi^1_{\theta,p,\epsilon}, \cdots \Phi^K_{\theta,p,\epsilon}$ each with $K+1$ Bernoulli arms indexed by $0, 1, \cdots, K$. For the instance $\Phi^0_{\theta,p,\epsilon}$, the costs and mean reward of the j-th arm are*

$$c_j^{\Phi^0_{\theta,p,\epsilon}} = \begin{cases} 0 & j = 0 \\ 1 & j \neq 0 \end{cases}, \qquad \mu_j^{\Phi^0_{\theta,p,\epsilon}} = \begin{cases} p & j = 0 \\ \frac{p}{1-\theta} & j \neq 0 \end{cases}.$$

for $0 \le j \le K$. For the instance $\Phi^a_{\theta,p,\epsilon}$ with $1 \le a \le K$, the costs and mean rewards of the j-th arm are

$$c_j^{\Phi^a_{\theta,p,\epsilon}} = \begin{cases} 0 & j = 0 \\ 1 & j \neq 0 \end{cases}, \qquad \mu_j^{\Phi^a_{\theta,p,\epsilon}} = \begin{cases} p & j = 0 \\ \frac{p+\epsilon}{1-\theta} & j = a \\ \frac{p}{1-\theta} & otherwise \end{cases}.$$

for $0 \le j \le K$, where $0 \le \theta < 1, 0 < p \le 1/2, \epsilon > 0$ and $(p+\epsilon)/(1-\theta) < 1$.

Lemma 1. *For any given p, K, T and any (possibly randomized) policy π, there exists an instance ϕ (from the family $\Phi_{0,p,\epsilon}$) such that $\mathsf{Quality\_Reg}_\pi(T,0,\phi) + \mathsf{Cost\_Reg}_\pi(T,0,\phi)$ is $\Omega\left(pK^{\frac{1}{3}}T^{\frac{2}{3}}\right)$ when $0 < p \le 1/2$ and $1 \le K \le T$.*

Lemma 1 establishes that when $\alpha = 0$, any policy must incur a regret of $\Omega(K^{1/3}T^{2/3})$ on an instance from the family $\Phi_{0,p,\epsilon}$. To prove Lemma 1, we argue that any online learning algorithm will not be able to differentiate the instance $\Phi^0_{0,p,\epsilon}$ from the instance $\Phi^a_{0,p,\epsilon}$ for $1 \le a \le K$ and therefore, must either incur a high cost regret if the algorithm does not select 0^{th} arm frequently or high quality regret if the algorithm selects 0^{th} arm frequently. More specifically, any online algorithm would require $O(1/\epsilon^2)$ samples or rounds to distinguish instance $\Phi^0_{0,p,\epsilon}$ from instance $\Phi^a_{0,p,\epsilon}$ for $1 \le a \le K$. Hence, any policy π can avoid high quality regret by *exploring sufficiently* for $O(1/\epsilon^2)$ rounds, incurring a cost regret of $O(1/\epsilon^2)$ or incur zero cost regret at the expense of $O(T\epsilon)$ regret on the reward metric. This suggests a trade-off between $1/\epsilon^2$ and $T\epsilon$, which are of the same magnitude at $\epsilon = T^{-1/3}$ resulting in the aforementioned lower bound. The complete proof generalizes techniques from the standard MAB lower bound proof and is provided in Appendix B. □

Now, we generalize the above result for $\alpha = 0$ to any α for $0 \le \alpha \le 1$. The main idea in our reduction is to show that if there exists an algorithm π_α for $\alpha > 0$ such that $\mathsf{Quality\_Reg}_\pi(T,\alpha,\phi) + \mathsf{Cost\_Reg}_\pi(T,\alpha,\phi)$ is $o(K^{1/3}T^{2/3})$ on every instance in the family $\Phi_{\alpha,p,\epsilon}$, then we can use π_α as a subroutine to construct an algorithm π for problem (1) such that $\mathsf{Quality\_Reg}_\pi(T,0,\phi) + \mathsf{Cost\_Reg}_\pi(T,0,\phi)$ is $o(K^{1/3}T^{2/3})$ on every instance in $\Phi_{0,p,\epsilon}$, thus contradicting the lower bound of Lemma 1. This will prove Theorem 2 by contradiction. To construct the aforementioned sub-routine, we leverage techniques from *Bernoulli factory* (Keane and O'Brien (1994); Huber (2013)) to generate a sample from a Bernoulli random variable with parameter $\mu/(1-\alpha)$ using samples from a Bernoulli random variable with parameter μ, for any $0 < \mu < 1-\alpha < 1$. We provide the exact sub-routine and complete proof in Appendix B.

4 Explore-Then-Commit based algorithm

We propose an explore-then-commit algorithm, named $\mathsf{Cost\text{-}Subsidized\ Explore\text{-}Then\text{-}Commit}$ (CS-ETC), to have better worst-case performance guarantees as compared to the extensions of the TS and UCB algorithms. As the name suggests, first, this algorithm plays each arm for a specified number of rounds. After sufficient exploration, the algorithm continues in a UCB-like fashion. In every round, based on the upper and lower confidence bounds on the reward of each arm, a feasible set of arms is constructed as an estimate of all arms having mean reward greater than the smallest tolerated reward. The lowest cost arm in this feasible set is then pulled. This is detailed in Algorithm 2. The key question that arises in this algorithm is how many exploration rounds are needed before exploitation can begin. We establish that $O\left((T/K)^{2/3}\right)$ rounds are sufficient for exploration in the following result (proof in Appendix C).

Algorithm 2: Cost-Subsidized Explore-Then-Commit

```
Result: Arm $I_t$ to be pulled in each round $t \in [T]$
Input  : $K, T$, no. of exploration pulls per arm $\tau$
$T_i(1) = 0 \ \forall i \in [K]$
Pure exploration phase:
for $t \in [1, K\tau]$ do
    $I_t = t \mod K$;
    Pull arm $I_t$ to obtain reward $r_t$;
    $T_i(t+1) = T_i(t) + \mathbf{1}\{I_t = i\} \ \forall i \in [K]$;
end
UCB phase:
for $t \in [K\tau + 1, T]$ do
    $\hat{\mu}_i(t) \leftarrow \left(\sum_{\tau=1}^{t-1} r_\tau \mathbf{I}\{I_\tau = i\}\right) / T_i(t) \ \forall i \in [K]$;
    $\beta_i(t) \leftarrow \sqrt{(2\log T) / T_i(t)} \ \forall i \in [K]$;
    $\mu_i^{\mathsf{UCB}}(t) \leftarrow \min\{\hat{\mu}_i(t) + \beta_i(t), 1\} \ \forall i \in [K]$;
    $\mu_i^{\mathsf{LCB}}(t) \leftarrow \max\{\hat{\mu}_i(t) - \beta_i(t), 0\} \ \forall i \in [K]$;
    $m_t = \arg\max_i \mu_i^{\mathsf{LCB}}(t)$;
    $Feas(t) = \{i : \mu_i^{\mathsf{UCB}}(t) \geq (1-\alpha)\mu_{m_t}^{\mathsf{LCB}}(t)\}$;
    $I_t = \arg\min_{i \in Feas(t)} c_i$;
    Pull arm $I_t$ to obtain reward $r_t$;
    $T_i(t+1) = T_i(t) + \mathbf{1}\{I_t = i\} \ \forall i \in [K]$;
end
```

Theorem 3. *For an instance ϕ with K arms, when the number of exploration pulls of each arm $\tau = (T/K)^{2/3}$, then the sum of cost and quality regret incurred by CS-ETC (Algorithm 2) on any instance ϕ i.e.* $\mathsf{Quality\_Reg}_{CS-ETC}(T,\alpha,\phi) + \mathsf{Cost\_Reg}_{CS-ETC}(T,\alpha,\phi)$ *is $O(K^{1/3}T^{2/3}\sqrt{\log T})$.*

The key reason that sufficient exploration is needed for our problem is that there can be arms with mean rewards very close to each other but significantly different costs. If cost regret were not of concern, then playing either arm would have led to satisfactory performance by giving low quality regret. The need to perform well on both cost and quality regrets necessitates differentiating between the two arms and finding the one with the cheapest cost among the arms with mean reward above the smallest tolerated reward.

The regret guarantee mainly stems from the exploration phase of the algorithm. In fact, an algorithm that estimates the optimal arm only once after the exploration phase and pulls that arm for the remaining time will have the same regret upper bound as CS-ETC. But we empirically observed that the non-asymptotic performance of this algorithm is worse as compared to Algorithm 2.

5 Performance with Constraints on Costs and Rewards

In this section, we present some extensions of the previous results.

5.1 Consistent Cost and Quality

The lower bound result in Theorem 2 is motivated by an extreme instance where arms with very similar mean rewards have very different costs. This raises the following question - can better-performing algorithms be obtained if the rewards and costs are *consistent* with each other? We show that this is indeed the case. Motivated by the instance which led to the worst-case performance, we consider a constraint that gives an upper bound on the difference in costs of every pair of arms by a multiple of the difference in the qualities of these arms. Under this constraint, CS-UCB has good performance as per the following result with the proof in Appendix C.

Theorem 4. *If for an instance ϕ with K arms, $|c_i - c_j| \leq \delta|\mu_i - \mu_j| \ \forall i,j \in [K]$ and any (possibly unknown) $\delta > 0$, then* $\mathsf{Quality\_Reg}_{CS-UCB}(T,\alpha,\phi) + \mathsf{Cost\_Reg}_{CS-UCB}(T,\alpha,\phi)$ *is $O((1+\delta)\sqrt{KT\log T})$.*

Note that, in general, δ can be unknown. Hence, even with the above assumption on the consistency of cost and quality, a priori any algorithm cannot get a bound on the quality difference between arms, only by virtue of knowing their costs.

5.2 Unknown Costs

In some applications, the costs of the arms may also be unknown and random. Hence, in addition to the mean reward, the mean costs also need to be estimated. Without loss of generality, we assume that the distribution of the random cost of each arm has support [0,1]. Not knowing the arm's cost does not fundamentally change the regret minimization problem we have discussed in the above sections. Clearly, the lower bound result is still valid. Algorithm 2 can be generalized to the unknown costs setting with a minor modification in the UCB phase of the algorithm. The modified UCB phase is described in Algorithm 3. In this algorithm, we maintain confidence bounds on the costs of each arm. Instead of picking the arm with the lowest cost among all feasible arms, the algorithm now picks the arm with the lowest lower confidence bound on cost. Theorem 3 holds for this modified algorithm also.

Similarly, when costs and quality are consistent as described in Section 5.1, the CS-UCB algorithm can be modified to pick the arm with the lowest lower confidence bound on cost and Theorem 4 holds.

```
Algorithm 3: CS-ETC with Unknown Costs
Result: Arm $I_t$ to be pulled in each round $t \in [T]$
input  : $K, T$, Number of exploration pulls $\tau$
$T_i(1) = 0 \ \forall i \in [K]$
Pure exploration phase:
for $t \in [1, Kf(K,T)]$ do
    $I_t = t \mod K$;
    Pull arm $I_t$ to obtain reward $r_t$ and cost $\chi_t$;
    $T_i(t+1) = T_i(t) + \mathbf{1}\{I_t = i\} \ \forall i \in [K]$;
end
UCB phase:
for $t \in [Kf(K,T)+1, T]$ do
    $\hat{\mu}_i(t) \leftarrow \frac{\sum_{\tau=1}^{t-1} r_\tau \mathbf{I}\{I_\tau = i\}}{T_i(t)} \ \forall i \in [K]$;
    $\hat{c}_i(t) \leftarrow \frac{\sum_{\tau=1}^{t-1} \chi_\tau \mathbf{I}\{I_\tau = i\}}{T_i(t)} \ \forall i \in [K]$;
    $\beta_i(t) \leftarrow \sqrt{\frac{2 \log T}{T_i(t)}} \ \forall i \in [K]$;
    $\mu_i^{\mathsf{UCB}}(t) \leftarrow \min\{\hat{\mu}_i(t) + \beta_i(t), 1\} \ \forall i \in [K]$;
    $\mu_i^{\mathsf{LCB}}(t) \leftarrow \max\{\hat{\mu}_i(t) - \beta_i(t), 0\} \ \forall i \in [K]$;
    $c_i^{\mathsf{LCB}}(t) \leftarrow \max\{\hat{c}_i(t) - \beta_i(t), 0\} \ \forall i \in [K]$;
    $m_t = \arg\max_i \mu_i^{\mathsf{LCB}}(t)$;
    $Feas(t) = \{i : \mu_i^{\mathsf{UCB}}(t) > (1-\alpha)\mu_{m_t}^{\mathsf{LCB}}(t)\}$;
    $I_t = \arg\min_{i \in Feas(t)} c_i^{\mathsf{LCB}}$;
    Pull arm $I_t$ to obtain reward $r_t$ and cost $\chi_t$;
    $T_i(t+1) = T_i(t) + \mathbf{1}\{I_t = i\} \ \forall i \in [K]$;
end
```

6 Numerical Experiments

In the previous sections, we have shown theoretical results on the worst-case performance of different algorithms for (1). Now, we illustrate the empirical performance of these algorithms. We shed light on which algorithm performs better in what regime of parameter values. The key quantity that differentiates the performance of different algorithms is how close the mean rewards of different arms are. We consider a setting with two Bernoulli arms and vary the mean reward of one arm (the cheaper arm) while keeping the other quantities (reward distribution of the other arm and costs of both arms) fixed. The values of these parameters are described in Table 1. The reward in each round follows a Bernoulli distribution, whereas the cost is a known fixed value. The cost and quality regret at time T of the different algorithms are plotted in Figure 2.

Parameter	**Value**
Mean reward of arm 1 (μ_1)	0.5
Mean reward of arm 2 (μ_2)	0.3-0.6
Cost of arm 1 (c_1)	1
Cost of arm 2 (c_2)	0
Subsidy factor (α)	0.1
Time horizon (T)	5000

Table 1: Parameter values

We observe that the performance of CS-TS and CS-UCB are close to each other for the entire range of mean reward values. To compare these algorithms' performance with CS-ETC, we focus on how close the mean reward of the lower mean reward arm is to the smallest tolerated reward. When $\mu_2 \leq 0.5$ ($\mu_2 > 0.5$), the lowest tolerated reward is 0.45 ($0.9\mu_2$). In terms of quality regret, when μ_2 is much smaller than 0.45, CS-TS and CS-UCB perform much better than CS-ETC. This is because the number of exploration rounds in the CS-ETC algorithm is fixed (independent of the difference in mean rewards of the two arms), leading to higher quality regret when μ_2 is much smaller than 0.45. On the other hand, because of the large difference in μ_2 and 0.45, CS-TS and CS-UCB algorithms can easily find the optimal arm and incur low quality regret. The cost regret of all algorithms is 0 because the optimal arm is the expensive arm.

When μ_2 is close to (and less than) 0.45, CS-TS and CS-UCB incur much higher cost regret as compared to CS-ETC. This is in line with the intuition established in Section 2. Here, CS-TS and CS-UCB are unable to effectively conclude that the second (cheaper) arm is optimal. Thus, they pull the first (expensive) arm many times, leading to high cost regret. On the other hand, CS-ETC, after the exploration rounds, can correctly identify the second arm as the optimal arm.

Thus, we recommend using the CS-TS/CS-UCB algorithm when the mean rewards of arms are well-differentiated and CS-ETC when the mean rewards are close to one another (as is often the case in the SMS application). This is in line with the notion that algorithms that perform well in the worst case might not have good performance for an average case.

6.1 Conclusion and Future Work

In this paper, we have proposed a new variant of the MAB problem, which factors costs associated with playing an arm and introduces new metrics that uniquely capture the features of multiple real-world

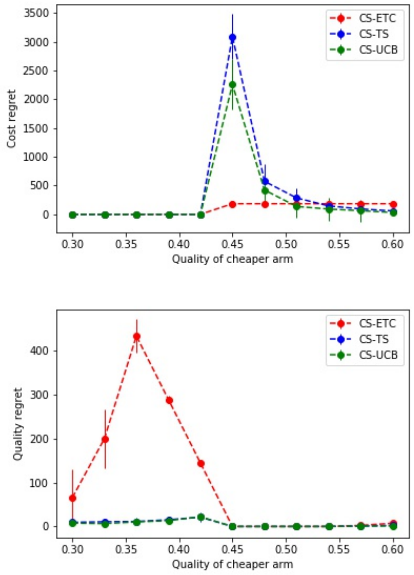

Figure 2: Performance of algorithms with varying mean reward of the cheaper arm. The length of the error bars correspond to two standard deviations in regret obtained by running the experiment 50 times.

applications. We argue about the *hardness* of this problem by establishing fundamental limits on the performance of any online algorithm and also demonstrating that traditional MAB algorithms perform poorly from both a theoretical and empirical standpoint. We present a simple near-optimal algorithm, and through numerical simulations, we prescribe an ideal algorithmic choice for different problem regimes.

An interesting direction for future work is defining a notion of instance-wise optimality and devising algorithms that achieve this. Insights from this can also help improve the CS-ETC algorithm to obtain better empirical performance. Another important question that naturally arises from this work is developing an algorithm for the adversarial variant of the MAB with cost subsidy problem. In particular, it is not immediately clear if EXP3 (Auer et al. (2002b)) family of algorithms, which are popular for non-stochastic MAB problem, can be generalized to settings where the reward distribution is not stationary.

Acknowledgements

We want to thank Nicolas Stier-Moses for introducing us to the SMS routing application and helping us in formulating it as an MAB problem. We would also like to thank three anonymous reviewers for their detailed feedback, which sharpened our arguments.

References

M. Abramowitz and I. A. Stegun. *Handbook of mathematical functions with formulas, graphs, and mathematical tables*, volume 55. US Government printing office, 1948.

S. Agrawal and N. R. Devanur. Bandits with concave rewards and convex knapsacks. In *Proceedings of the Fifteenth ACM Conference on Economics and Computation*, EC '14, page 989–1006, New York, NY, USA, 2014. Association for Computing Machinery. ISBN 9781450325653.

S. Agrawal and N. Goyal. Near-optimal regret bounds for thompson sampling. *J. ACM*, 64(5), 2017a.

S. Agrawal and N. Goyal. Near-optimal regret bounds for thompson sampling. *Journal of the ACM (JACM)*, 64(5):1–24, 2017b.

S. Agrawal, V. Avadhanula, V. Goyal, and A. Zeevi. A near-optimal exploration-exploitation approach for assortment selection. *Proceedings of the 2016 ACM Conference on Economics and Computation (EC)*, pages 599–600, 2016.

S. Agrawal, V. Avadhanula, V. Goyal, and A. Zeevi. Thompson sampling for the mnl-bandit. In *Conference on Learning Theory*, pages 76–78, 2017.

S. Amani, M. Alizadeh, and C. Thrampoulidis. Generalized linear bandits with safety constraints. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3562–3566. IEEE, 2020.

Amazon. Amazon auto-targeting, 2019. URL `https://tinyurl.com/yx9lyfwq`.

A. Anandkumar, N. Michael, A. K. Tang, and A. Swami. Distributed algorithms for learning and cognitive medium access with logarithmic regret. *IEEE Journal on Selected Areas in Communications*, 29(4):731–745, 2011.

P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire. The nonstochastic multiarmed bandit problem. *SIAM Journal on Computing*, 32(1):48–77, 2002a.

P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire. The nonstochastic multiarmed bandit problem. *SIAM journal on computing*, 32(1):48–77, 2002b.

A. Badanidiyuru, R. Kleinberg, and A. Slivkins. Bandits with knapsacks. *J. ACM*, 65(3):13:1–13:55, Mar. 2018. ISSN 0004-5411. doi: 10.1145/3164539. URL `http://doi.acm.org/10.1145/3164539`.

S. Bubeck and N. Cesa-Bianchi. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations and Trends® in Machine Learning*, 5(1):1–122, 2012.

M. Canlas, K. P. Cruz, M. K. Dimarucut, P. Uyengco, G. Tangonan, M. L. Guico, N. Libatique, and C. Pineda. A quantitative analysis of the quality of service of short message service in the philippines. In *2010 IEEE International Conference on Communication Systems*, pages 710–714. IEEE, 2010.

W. Cao, J. Li, Y. Tao, and Z. Li. On top-k selection in multi-armed bandits and hidden bipartite graphs. In *Advances in Neural Information Processing Systems*, pages 1036–1044, 2015.

N. Cesa-Bianchi, C. Gentile, and Y. Mansour. Regret minimization for reserve prices in second-price auctions. *IEEE Transactions on Information Theory*, 61(1):549–564, 2014.

L. Chen, A. Gupta, and J. Li. Pure exploration of multi-armed bandit under matroid constraints. In *Conference on Learning Theory*, pages 647–669, 2016.

S. Chen, T. Lin, I. King, M. R. Lyu, and W. Chen. Combinatorial pure exploration of multi-armed bandits. In *Advances in Neural Information Processing Systems*, pages 379–387, 2014.

S. Daulton, S. Singh, V. Avadhanula, D. Dimmery, and E. Bakshy. Thompson sampling for contextual bandit problems with auxiliary safety constraints. *arXiv preprint arXiv:1911.00638*, 2019.

M. M. Drugan and A. Nowe. Designing multi-objective multi-armed bandits algorithms: A study. In *The 2013 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2013.

Facebook. Facebook targeting expansion, 2016. URL `https://tinyurl.com/y3ss2j8g`.

N. Galichet, M. Sebag, and O. Teytaud. Exploration vs exploitation vs safety: Risk-aware multi-armed bandits. In *Asian Conference on Machine Learning*, pages 245–260, 2013.

Google. Google auto-targeting, 2014. URL `https://tinyurl.com/y3c4bdaj`.

M. Huber. Nearly optimal bernoulli factories for linear functions. *arXiv preprint arXiv:1308.1562*, 2013.

N. Immorlica, K. A. Sankararaman, R. Schapire, and A. Slivkins. Adversarial bandits with knapsacks. In *2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 202–219, 2019.

K. Jamieson and R. Nowak. Best-arm identification algorithms for multi-armed bandits in the fixed confidence setting. In *2014 48th Annual Conference on Information Sciences and Systems (CISS)*, pages 1–6. IEEE, 2014.

J. Katz-Samuels and C. Scott. Top feasible arm identification. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 1593–1601, 2019.

A. Kazerouni, M. Ghavamzadeh, Y. A. Yadkori, and B. Van Roy. Conservative contextual linear bandits. In *Advances in Neural Information Processing Systems*, pages 3910–3919, 2017.

M. Keane and G. L. O'Brien. A bernoulli factory. *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, 4(2):213–219, 1994.

R. Koningstein. Suggesting and/or providing targeting information for advertisements, July 6 2006. US Patent App. 11/026,508.

J. Langford and T. Zhang. The epoch-greedy algorithm for multi-armed bandits with side information. In *Advances in neural information processing systems*, pages 817–824, 2008.

L. Li, W. Chu, J. Langford, and X. Wang. Unbiased offline evaluation of contextual-bandit-based news article recommendation algorithms. In *Proceedings of the fourth ACM international conference on Web search and data mining*, pages 297–306, 2011.

L. Li, S. Chen, J. Kleban, and A. Gupta. Counterfactual estimation and optimization of click metrics in search engines: A case study. In *Proceedings of the 24th International Conference on World Wide Web*, pages 929–934, 2015.

MarketWatch. Marketwatch a2p report, 2020. URL `https://rb.gy/0w96oi`.

B. C. May, N. Korda, A. Lee, and D. S. Leslie. Optimistic bayesian sampling in contextual-bandit problems. *Journal of Machine Learning Research*, (13): 2069–2106, 2012.

X. Meng, P. Zerfos, V. Samanta, S. H. Wong, and S. Lu. Analysis of the reliability of a nationwide short message service. In *IEEE INFOCOM 2007-26th IEEE International Conference on Computer Communications*, pages 1811–1819. IEEE, 2007.

C. Oliver and L. Li. An empirical evaluation of thompson sampling. *In Advances in Neural Information Processing Systems (NIPS)*, 24:2249?2257, 2011.

O. Osunade and S. O. Nurudeen. Route optimization for delivery of short message service in telecommunication networks.

Ovum. Sustaining a2p sms growth while securing mobile network, 2017. URL `https://rb.gy/qqonzd`.

B. Paria, K. Kandasamy, and B. Póczos. A flexible framework for multi-objective bayesian optimization using random scalarizations. *arXiv preprint arXiv:1805.12168*, 2018.

H. Robbins. Some aspects of the sequential design of experiments. *Bulletin of the American Mathematical Society*, 58(5):527–535, 1952.

A. Sankararaman, A. Ganesh, and S. Shakkottai. Social learning in multi agent multi armed bandits. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 3(3):1–35, 2019.

S. L. Scott. A modern bayesian look at the multi-armed bandit. *Applied Stochastic Models in Business and Industry*, 26(6):639–658, 2010.

A. Slivkins. Introduction to multi-armed bandits. *Foundations and Trends® in Machine Learning*, 12(1-2):1–286, 2019. ISSN 1935-8237. URL http://dx.doi.org/10.1561/2200000068.

A. Slivkins and J. W. Vaughan. Online decision making in crowdsourcing markets: Theoretical challenges. *ACM SIGecom Exchanges*, 12(2):4–23, 2014.

W. Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3/4):285–294, 1933.

L. Tran-Thanh, A. Chapman, A. Rogers, and N. R. Jennings. Knapsack based optimal policies for budget–limited multi–armed bandits. In *Twenty-Sixth AAAI Conference on Artificial Intelligence*, 2012.

Twilio and Uber. Uber built a great ridesharing experience with sms & voice, 2020. URL https://customers.twilio.com/208/uber/.

Y. Wu, R. Shariff, T. Lattimore, and C. Szepesvári. Conservative bandits. In *International Conference on Machine Learning*, pages 1254–1262, 2016.

S. Yahyaa and B. Manderick. Thompson sampling for multi-objective multi-armed bandits problem. In *Proceedings*, page 47. Presses universitaires de Louvain, 2015.

S. Q. Yahyaa, M. M. Drugan, and B. Manderick. Annealing-pareto multi-objective multi-armed bandit algorithm. In *2014 IEEE Symposium on Adaptive Dynamic Programming and Reinforcement Learning (ADPRL)*, pages 1–8. IEEE, 2014.

P. Zerfos, X. Meng, S. H. Wong, V. Samanta, and S. Lu. A study of the short message service of a nationwide cellular network. In *Proceedings of the 6th ACM SIGCOMM conference on Internet measurement*, pages 263–268, 2006.