AdCell: Ad Allocation in Cellular Networks

Saeed Alaei[‡1], Mohammad T. Hajiaghayi[§12], Vahid Liaghat[¶1], Dan Pei[2], and Barna Saha[‖1]
{saeed, hajiagha, vliaghat, barna}@cs.umd.edu, peidan@research.att.com

[1] University of Maryland, College Park, MD, 20742
[2] AT&T Labs - Research, 180 Park Avenue, Florham Park, NJ 07932

Abstract. With more than four billion usage of cellular phones worldwide, mobile advertising has become an attractive alternative to online advertisements. In this paper, we propose a new targeted advertising policy for Wireless Service Providers (WSPs) via SMS or MMS- namely *AdCell*. In our model, a WSP charges the advertisers for showing their ads. Each advertiser has a valuation for specific types of customers in various times and locations and has a limit on the maximum available budget. Each query is in the form of time and location and is associated with one individual customer. In order to achieve a non-intrusive delivery, only a limited number of ads can be sent to each customer. Recently, new services have been introduced that offer location-based advertising over cellular network that fit in our model (e.g., ShopAlerts by AT&T) .
We consider both online and offline version of the AdCell problem and develop approximation algorithms with constant competitive ratio. For the online version, we assume that the appearances of the queries follow a stochastic distribution and thus consider a Bayesian setting. Furthermore, queries may come from different distributions on different times. This model generalizes several previous advertising models such as online secretary problem [10], online bipartite matching [13,7] and AdWords [18]. Since our problem generalizes the well-known secretary problem, no non-trivial approximation can be guaranteed in the online setting without stochastic assumptions. We propose an online algorithm that is simple, intuitive and easily implementable in practice. It is based on pre-computing a fractional solution for the expected scenario and relies on a novel use of dynamic programming to compute the conditional expectations. We give tight lower bounds on the approximability of some variants of the problem as well. In the offline setting, where full-information is available, we achieve near-optimal bounds, matching the integrality gap of the considered linear program. We believe that our proposed solutions can be used for other advertising settings where personalized advertisement is critical.

Keywords: Mobile Advertisement, AdCell, Online, Matching

‡ Supported in part by NSF Grant CCF-0728839.
§ Supported in part by NSF CAREER Award, ONR Young Investigator Award, and Google Faculty Research Award.
¶ Supported in part by NSF CAREER Award, ONR Young Investigator Award, and Google Faculty Research Award.
‖ Supported in part by NSF Award CCF-0728839, NSF Award CCF-0937865.

1 Introduction

In this paper, we propose a new mobile advertising concept called *Adcell*. More than 4 billion cellular phones are in use world-wide, and with the increasing popularity of smart phones, mobile advertising holds the prospect of significant growth in the near future. Some research firms [1] estimate mobile advertisements to reach a business worth over 10 billion US dollars by 2012. Given the built-in advertisement solutions from popular smart phone OSes, such as iAds for Apple's iOS, mobile advertising market is poised with even faster growth.

In the mobile advertising ecosystem, wireless service providers (WSPs) render the physical delivery infrastructure, but so far WSPs have been more or less left out from profiting via mobile advertising because of several challenges. First, unlike web, search, application, and game providers, WSPs typically do not have users' application context, which makes it difficult to provide targeted advertisements. Deep Packet Inspection (DPI) techniques that examine packet traces in order to understand application context, is often not an option because of privacy and legislation issues (i.e., Federal Wiretap Act). Therefore, a targeted advertising solution for WSPs need to utilize *only the information it is allowed to collect by government and by customers via opt-in mechanisms*. Second, without the luxury of application context, targeted ads from WSPs require *non-intrusive delivery methods*. While users are familiar with other ad forms such as banner, search, in-application, and in-game, push ads with no application context (e.g., via SMS) can be intrusive and annoying if not done carefully. The number and frequency of ads both need to be well-controlled. Third, targeted ads from WSPs should be well personalized such that the users have incentive to read the advertisements and take purchasing actions, especially given the requirement that the number of ads that can be shown to a customer is limited.

In this paper, we propose a new mobile targeted advertising strategy, *AdCell*, for WSPs that deals with the above challenges. It takes advantage of the detailed real-time location information of users. Location can be tracked upon users' consent. This is already being done in some services offered by WSPs, such as Sprint's Family Location and AT&T's Family Map, thus there is no associated privacy or legal complications. To locate a cellular phone, it must emit a roaming signal to contact some nearby antenna tower, but the process does not require an active call. GSM localization is then done by multi-lateration[3] based on the signal strength to nearby antenna masts [22]. Location-based advertisement is not completely new. Foursquare mobile application allows users to explicitly "check in" at places such as bars and restaurants, and the shops can advertise accordingly. Similarly there are also automatic proximity-based advertisements using GPS or bluetooth. For example, some GPS models from Garmin display ads for the nearby business based on the GPS locations [23]. ShopAlerts by AT&T [4] is another application along the same line. On the advertiser side, popular stores such as Starbucks are reported to have attracted significant footfalls via mobile coupons.

[3] The process of locating an object by accurately computing the time difference of arrival of a signal emitted from that object to three or more receivers.

[4] http://shopalerts.att.com/sho/att/index.html?ref=portal

Most of the existing mobile advertising models are On-Demand, however, AdCell sends the ads via SMS, MMS, or similar methods without any prior notice. Thus to deal with the non-intrusive delivery challenge, we propose user subscription to advertising services that deliver only a *fixed number* of ads per month to its subscribers (as it is the case in AT&T ShopAlerts). The constraint of delivering limited number of ads to each customer adds the main algorithmic challenge in the AdCell model (details in Section 1.1). In order to overcome the incentive challenge, the WSP can “pay” users to read ads and purchase based on them through a reward program in the form of credit for monthly wireless bill. To begin with, both customers and advertisers should sign-up for the AdCell-service provided by the WSP (e.g., currently there are 9 chain-companies participating in ShopAlerts). Customers enrolled for the service should sign an agreement that their *location* information will be tracked; but solely for the advertisement purpose. Advertisers (e.g., stores) provide their advertisements and a maximum chargeable budget to the WSP. The WSP selects proper ads (these, for example, may depend on time and distance of a customer from a store) and sends them (via SMS) to the customers. The WSP charges the advertisers for showing their ads and also for successful ads. An ad is deemed successful if a customer visits the advertised store. Depending on the service plan, customers are entitled to receive different number of advertisements per month. Several logistics need to be employed to improve AdCell experience and enthuse customers into participation. We provide more details about these logistics in the full paper.

1.1 AdCell Model & Problem Formulation

In the AdCell model, advertisers bid for individual customers based on their location and time. The triple (k, ℓ, t) where k is a customer, ℓ is a neighborhood (location) and t is a time forms a *query* and there is a bid amount (possibly zero) associated with each query for each advertiser. This definition of query allows advertisers to customize their bids based on customers, neighborhoods and time. We assume a customer can only be in one neighborhood at any particular time and thus at any time t and for each customer k, the queries (k, ℓ_1, t) and (k, ℓ_2, t) are mutually exclusive, for all distinct l_1, l_2. Neighborhoods are places of interest such as shopping malls, airports, etc. We assume that queries are generated at certain times (e.g., every half hour) and only if a customer stays within a neighborhood for a specified minimum amount of time. The formal problem definition of *AdCell Allocation* is as follows:

AdCell Allocation *There are m advertisers, n queries and s customers. Advertiser i has a total budget b_i and bids u_{ij} for each query j. Furthermore, for each customer $k \in [s]$, let S_k denote the queries corresponding to customer k and c_k denote the maximum number of ads which can be sent to customer k. The capacity c_k is associated with customer k and is dictated by the AdCell plan the customer has signed up for. Advertiser i pays u_{ij} if his advertisement is shown for query j and if his budget is not exceeded. That is, if x_{ij} is an indicator variable set to 1, when advertisement for advertiser i is shown on query j, then advertiser i pays a total amount of $\min(\sum_j x_{ij}u_{ij}, b_i)$. The goal of AdCell Allocation is to specify an advertisement allocation plan such that the total payment $\sum_i \min(\sum_j x_{ij}u_{ij}, b_i)$ is maximized.*

The AdCell problem is a generalization of the budgeted AdWords allocation problem [4,21] with capacity constraint on each customer and thus is NP-hard. Along with the offline version of the problem, we also consider its online version where queries arrive online and a decision to assign a query to an advertiser has to be done right away. With arbitrary queries/bids and optimizing for the worst case, one cannot obtain any approximation algorithm with ratio better than $\frac{1}{n}$. This follows from the observation that online AdCell problem also generalizes the *secretary problem* for which no deterministic or randomized online algorithm can get approximation ratio better than $\frac{1}{n}$ in the worst case.[5]. Therefore, we consider a stochastic setting.

For the online AdCell problem, we assume that each query j arrives with probability p_j. Upon arrival, each query has to be either allocated or discarded right away. We note that each query encodes a customer id, a location id and a time stamp. Also associated with each query, there is a probability, and a vector consisting of the bids for all advertisers for that query. Furthermore, we assume that all queries with different arrival times or from different customers are independent, however queries from the same customer with the same arrival time are mutually exclusive (i.e., a customer cannot be in multiple locations at the same time).

1.2 Our Results and Techniques

Here we provide a summary of our results and techniques. We consider both the offline and online version of the problem. In the offline version, we assume that we know exactly which queries arrive. In the online version, we only know the arrival probabilities of queries (i.e., $p_1, \cdots, p_m$).

We can write the AdCell problem as the following random integer program in which $\mathbf{I}_j$ is the indicator random variable which is 1 if query j arrives and 0 otherwise:

$$\begin{aligned} &\text{maximize.} && \sum_i \min(\sum_j \mathbf{X}_{ij} u_{ij}, b_i) && (IP_{BC}) \\ &\forall j \in [n]: && \sum_i \mathbf{X}_{ij} \leq \mathbf{I}_j && (F) \\ &\forall k \in [s]: && \sum_{j \in S_k} \sum_i \mathbf{X}_{ij} \leq c_k && (C) \\ & && \mathbf{X}_{ij} \in \{0,1\} \end{aligned}$$

We will refer to the variant of the problem explained above as IP_{BC}. We also consider variants in which there are either budget constraints or capacity constraints but not both. We refer to these variants as IP_B and IP_C respectively. The above integer program can be relaxed to obtain a linear program LP_{BC}, where we maximize $\sum_i \sum_j \mathbf{X}_{ij} u_{ij}$ with the constraints (F), (C) and additional budget constraint $\sum_j \mathbf{X}_{ij} u_{ij} \leq b_i$ which we refer to by (B). We relax $\mathbf{X}_{ij} \in \{0,1\}$ to $\mathbf{X}_{ij} \in [0,1]$. We also refer to the variant of

[5] The reduction of the *secretary problem* to AdCell problem is as follows: consider a single advertiser with large enough budget and a single customer with a capacity of 1. The queries correspond to secretaries and the bids correspond to the values of the secretaries. So we can only allocate one query to the advertiser.

this linear program with only either constraints of type (B) or constraints of type (C) as LP_B and LP_C.

In the offline version, for all $i \in [m]$ and $j \in [n]$, the values of $\mathbf{I}_j$ are precisely known. For the online version, we assume to know the $E[\mathbf{I}_j]$ in advance and we learn the actual value of $\mathbf{I}_j$ online. We note a crucial difference between our model and the i.i.d model. In i.i.d model the probability of the arrival of a query is independent of the time, i.e., queries arrive from the same distribution on each time. However, in AdCell model a query encodes time (in addition to location and customer id), hence we may have a different distribution on each time. This implies a prophet inequality setting in which on each time, an onlooker has to decide according to a given value where this value may come from a different distribution on different times (e.g. see [14,11]).

A summary of our results are shown in Table 1. In the online version, we compare the expected revenue of our solution with the expected revenue of the optimal offline algorithm. We should emphasis that we make no assumptions about bid to budget ratios (e.g., bids could be as large as budgets). In the offline version, our result matches the known bounds on the integrality gap.

We now briefly describe our main techniques.

Breaking into smaller sub-problems that can be optimally solved using conditional expectation. Theoretically, ignoring the computational issues, any online stochastic optimization problem can be solved optimally using conditional expectation as follows: At any time a decision needs to be made, compute the total expected objective conditioned on each possible decision, then chose the one with the highest total expectation. These conditional expectations can be computed by backward induction, possibly using a dynamic program. However for most problems, including the AdCell problem, the size of this dynamic program is exponential which makes it impractical. We avoid this issue by using a randomized strategy to break the problem into smaller subproblems such that each subproblem can be solved by a quadratic dynamic program.

Using an LP to analyze the performance of an optimal online algorithm against an optimal offline fractional solution. Note that we compare the expected objective value of our algorithm against the expected objective value of the optimal offline fractional solution. Therefore for each subproblem, even though we use an optimal online algorithm, we still need to compare its expected objective value against the expected objective value of the optimal offline solution for that subproblem. Basically, we need to compare the expected objective of an stochastic online algorithm, which works by

Offline Version	**Online Version**
– A $\frac{3}{4}$-approximation algorithm. – A $\frac{4-\epsilon}{4}$-approximation algorithm when $\forall_i \max_j u_{ij} \leq \epsilon b_i$.	– A $\left(\frac{1}{2} - \frac{1}{e}\right)$-approximation algorithm. – A $\left(1 - \frac{1}{e}\right)$-approximation algorithm with only budget constraints. – A $\frac{1}{2}$-approximation algorithm with only capacity constraints.

Table 1. Summary of Our Results

maximizing conditional expectation at each step, against the expected objective value of its optimal offline solution. To do this, we create a minimization linear program that encodes the dynamic program and whose optimal objective is the minimum ratio of the expected objective value of the online algorithm to the expected objective value of the optimal offline solution. We then prove a lower bound of $\frac{1}{2}$ on the objective value of this linear program by constructing a feasible solution for its dual obtaining an objective value of $\frac{1}{2}$.

Rounding method of [20] and handling hard capacities. Handling "hard capacities", those that cannot be violated, is generally tricky in various settings including facility location and many covering problems [5,8,19]. The AdCell problem is a generalization of the budgeted AdWords allocation problem with hard capacities on queries involving each customer. Our essential idea is to iteratively round the fractional LP solution to an integral one based on the current LP structure. The algorithm uses the rounding technique of [20] and is significantly harder than its uncapacitated version.

Due to the interest of the space we differ the omitted proofs to the full paper.

2 Related Work

Online advertising alongside search results is a multi-billion dollar business [15] and is a major source of revenue for search engines like Google, Yahoo and Bing. A related ad allocation problem is the AdWords assignment problem [18] that was motivated by sponsored search auctions. When modeled as an online bipartite assignment problem, each edge has a weight, and there is a budget on each advertiser representing the upper bound on the total weight of edges that might be assigned to it. In the offline setting, this problem is NP-Hard, and several approximations have been proposed [3,2,4,21]. For the online setting, it is typical to assume that edge weights (i.e., bids) are much smaller than the budgets, in which case there exists a $(1 - 1/e)$-competitive online algorithm [18]. Recently, Devanur and Hayes [6] improved the competitive ratio to $(1 - \epsilon)$ in the stochastic case where the sequence of arrivals is a random permutation.

Another related problem is the online bipartite matching problem which is introduced by Karp, Vazirani, and Vazirani [13]. They proved that a simple randomized online algorithm achieves a $(1-1/e)$-competitive ratio and this factor is the best possible. Online bipartite matching has been considered under stochastic assumptions in [9,7,17], where improvements over $(1 - 1/e)$ approximation factor have been shown. The most recent of of them is the work of Manshadi et al. [17] that presents an online algorithm with a competitive ratio of 0.702. They also show that no online algorithm can achieve a competitive ratio better than 0.823. More recently, Mahdian et al.[16] and Mehta et al.[12] improved the competitive ratio to 0.696 for unknown distributions.

3 Online Setting

In this section, we present three online algorithms for the three variants of the problem mentioned in the pervious section (i.e., IP_B, IP_C and IP_{BC}).

First, we present the following lemma which provides a means of computing an upper bound on the expected revenue of any algorithm (both online and offline) for the AdCell problem.

Lemma 1 (Expectation Linear Program). *Consider a general random linear program in which* $\mathfrak{b}$ *is a vector of random variables:*

(Random LP)

$$\begin{aligned} &\textit{maximize.} && c^T x \\ &\textit{s.t.} && Ax \leq \mathfrak{b}; \;\; x \geq 0 \end{aligned}$$

Let $OPT(\mathfrak{b})$ *denote the optimal value of this program as a function of the random variables. Now consider the following linear program:*

(Expectation LP)

$$\begin{aligned} &\textit{maximize.} && c^T x \\ &\textit{s.t.} && Ax \leq E[\mathfrak{b}]; \;\; x \geq 0 \end{aligned}$$

We refer to this as the "Expectation Linear Program" *corresponding to the* "Random Linear Program". *Let* $\overline{OPT}$ *denote the optimal value value of this program. Assuming that the original linear program is feasible for all possible draws of the random variables, it always holds that* $E[OPT(\mathfrak{b})] \leq \overline{OPT}$.

Proof. Let $x^*(\mathfrak{b})$ denote the optimal assignment as a function of $\mathfrak{b}$. Since the random LP is feasible for all realizations of $\mathfrak{b}$, we have $Ax^*(\mathfrak{b}) \leq \mathfrak{b}$. Taking the expectation from both sides, we get $AE[x^*(\mathfrak{b})] \leq E[\mathfrak{b}]$. So, by setting $x = E[x^*(\mathfrak{b})]$ we get a feasible solution for the expectation LP. Furthermore, the objective value resulting from this assignment is equal to the expected optimal value of the random LP. The optimal value of the expectation LP might however be higher so its optimal value is an upper bound on the expected optimal value of random LP.

As we will see next, not only does the expectation LP provide an upper bound on the expected revenue, it also leads to a good approximate algorithm for the online allocation as we explain in the following online allocation algorithm. We adopt the notation of using an overline to denote the expectation linear program corresponding to a random linear program (e.g. $\overline{LP}_{BC}$ for LP_{BC}). Next we present an online algorithm for the variant of the problem in which there are only budget constrains but not capacity constraints.

Algorithm 1 (STOCHASTIC ONLINE ALLOCATOR FOR IP_B)

- *Compute an optimal assignment for the corresponding expectation LP (i.e.* $\overline{LP}_B$*). Let* x^*_{ij} *denote this assignment. Note that* x^*_{ij} *might be a fractional assignment.*
- *If query* j *arrives, for each* $i \in [m]$ *allocate the query to advertiser* i *with probability* $\frac{x^*_{ij}}{p_j}$.

Theorem 1. *The expected revenue of 1 is at least* $1 - \frac{1}{e}$ *of the optimal value of the expectation LP (i.e.,* $\overline{LP}_B$*) which implies that the expected revenue of 1 it is at least* $1 - \frac{1}{e}$ *of the expected revenue of the optimal offline allocation too. Note that this result holds even if* u_{ij}*'s are not small compared to* b_i*. Furthermore, this result holds even if we relax the independence requirement in the original problem and require negative correlation instead.*

Note that allowing negative correlation instead of independence makes the above model much more general than it may seem at first. For example, suppose there is a query that may arrive at several different times but may only arrive at most once or only a limited number of times, we can model this by creating a new query for each possible instance of the original query. These new copies are however negatively correlated.

Remark 1. It is worth mentioning that there is an integrality gap of $1 - \frac{1}{e}$ between the optimal value of the integral allocation and the optimal value of the expectation LP. So the lower bound of Theorem 1 is tight. To see this, consider a single advertiser and n queries. Suppose $p_j = \frac{1}{n}$ and $u_{1j} = 1$ for all j. The optimal value of $\overline{LP}_B$ is 1 but even the expected optimal revenue of the offline optimal allocation is $1 - \frac{1}{e}$ when $n \to \infty$ because with probability $(1 - \frac{1}{n})^n$ no query arrives.

To prove Theorem 1, we use the following theorem:

Theorem 2. *Let* C *be an arbitrary positive number and let* $\mathbf{X}_1, \cdots, \mathbf{X}_n$ *be independent random variables (or negatively correlated) such that* $\mathbf{X}_i \in [0, C]$*. Let* $\mu = E[\sum_i \mathbf{X}_i]$*. Then:*

$$E[\min(\textstyle\sum_i \mathbf{X}_i, C)] \geq (1 - \frac{1}{e^{\mu/C}})C$$

Furthermore, if $\mu \leq C$ *then the right hand side is at least* $(1 - \frac{1}{e})\mu$*.*

Proof (Theorem 1). We apply Theorem 2 to each advertiser i separately. From the perspective of advertiser i, each query is allocated to her with probability x^*_{ij} and by constraint (B) we can argue that have $\mu = \sum_j x^*_{ij} u_{ij} \leq b_i = C$ so $\mu \leq C$ and by Theorem 2, the expected revenue from advertiser i is at least $(1 - \frac{1}{e})(\sum_j x^*_{ij} u_{ij})$. Therefore, overall, we achieve at least $1 - \frac{1}{e}$ of the optimal value of the expectation LP and that completes the proof.

Next we present an online algorithm for the variant of the problem in which there are only capacity constrains but not budget constraints.

Algorithm 2 (STOCHASTIC ONLINE ALLOCATOR FOR IP_C)

- *Compute an optimal assignment for the corresponding expectation LP (i.e.* $\overline{LP}_C$*). Let* x^*_{ij} *denote this assignment. Note that* x^*_{ij} *might be a fractional assignment.*
- *Partition the items to sets* $T_1, \cdots, T_u$ *in increasing order of their arrival time and such that all of the items in the same set have the same arrival time.*
- *For each* $k \in [s], t \in [u], r \in [c_k]$*, let* $E^r_{k,t}$ *denote the expected revenue of the algorithm from queries in* S_k *(i.e., associated with customer* k*) that arrive at or after* T_t *and assuming that the remaining capacity of customer* k *is* r*. We formally define* $E^r_{k,t}$ *later.*

– *If query j arrives then choose one of the advertisers at random with advertiser i chosen with a probability of $\frac{x^*_{ij}}{p_j}$. Let k and T_t be respectively the customer and the partition which query j belongs to. Also, let r be the remaining capacity of customer k (i.e. r is c_k minus the number of queries from customer k that have been allocated so far). If $u_{ij} + E^{r-1}_{k,t+1} \geq E^r_{k,t+1}$ then allocate query j to advertiser i otherwise discard query j.*

We can now define $E^r_{k,t}$ recursively as follows:

$$E^r_{k,t} = \sum_{j \in T_t} \sum_{i \in [m]} x^*_{ij} \max(u_{ij} + E^{r-1}_{k,t+1}, E^r_{k,t+1}) + (1 - \sum_{j \in T_t} \sum_{i \in [m]} x^*_{ij}) E^r_{k,t+1} \qquad (\text{EXP}_k)$$

Also define $E^0_{k,t} = 0$ and $E^r_{k,u+1} = 0$. Note that we can efficiently compute $E^r_{k,t}$ using dynamic programming.

The main difference between 1 and 2 is that in the former whenever we choose an advertiser at random, we always allocate the query to that advertiser (assuming they have enough budget). However, in the latter, we run a dynamic program for each customer k and once an advertiser is picked at random, the query is allocated to this advertiser only if doing so increases the expected revenue associated with customer k.

Theorem 3. *The expected revenue of 2 is at least $\frac{1}{2}$ of the optimal value of the expectation LP (i.e., $\overline{LP}_C$) which implies that the expected revenue of 2 it is at least $\frac{1}{2}$ of the expected revenue of the optimal offline allocation for IP_C too.*

Remark 2. The approximation ratio of 2 is tight. There is no online algorithm that can achieve in expectation better than $\frac{1}{2}$ of the revenue of the optimal offline allocation without making further assumptions. We show this by providing a simple example. Consider an advertiser with a large enough budget and a single customer with a capacity of 1 and two queries. The queries arrive independently with probabilities $p_1 = 1-\epsilon$ and $p_2 = \epsilon$ with the first query having an earlier arrival time. The advertiser has submitted the bids $b_{11} = 1$ and $b_{12} = \frac{1-\epsilon}{\epsilon}$. Observe that no online algorithm can get a revenue better than $(1-\epsilon)\times 1+\epsilon^2 \frac{1-\epsilon}{\epsilon} \approx 1$ in expectation because at the time query 1 arrives, the online algorithm does not know whether or not the second query is going to arrive and the expected revenue from the second query is just $1 - \epsilon$. However, the optimal offline solution would allocate the second query if it arrives and otherwise would allocate the first query so its revenue is $\epsilon \frac{1-\epsilon}{\epsilon} + (1 - \epsilon)^2 \times 1 \approx 2$ in expectation.

Next, we show that an algorithm similar to the previous one can be used when there are both budget constraints and capacity constraints.

Algorithm 3 (STOCHASTIC ONLINE ALLOCATOR FOR IP_{BC})
Run the same algorithm as in 2 except that now x^*_{ij} is a fractional solution of $\overline{LP}_{BC}$ instead of $\overline{LP}_C$.

Theorem 4. *The expected revenue of 3 is at least $\frac{1}{2} - \frac{1}{e}$ of the optimal value of the expectation LP (i.e., $\overline{LP}_{BC}$) which implies that the expected revenue of 3 it is at least $\frac{1}{2} - \frac{1}{e}$ of the expected revenue of the optimal offline allocation too.*

We prove the last two theorems by defining a simple stochastic uniform knapsack problem which will be used as a building block in our analysis. Due to the interest of the space we have moved the proofs to the full paper.

4 Offline Setting

In the offline setting, we explicitly know all the queries, that is all the customers, locations, items triplets on which advertisers put their bids. We want to obtain an allocation of advertisers to queries such that the total payment obtained from all the advertisers is maximized. Each advertiser pays an amount equal to the minimum of his budget and the total bid value on all the queries assigned to him. Since, the problem is NP-Hard, we can only obtain an approximation algorithm achieving revenue close to the optimal. The fractional optimal solution of LP_{BC} (with explicit values for $\mathcal{I}_j, j \in [n]$) acts as an upper bound on the optimal revenue. We round the fractional optimal solution to a nearby integer solution and establish the following bound.

Theorem 5. *Given a fractional optimal solution for LP_{BC}, we can obtain an integral solution for AdCell with budget and capacity constraints that obtains at least a profit of $\frac{4 - \max_i \frac{u_{i,max}}{b_i}}{4}$ of the profit obtained by optimal fractional allocation and maintains all the capacity constraints exactly.*

We note that this approximation ratio is best possible using the considered LP relaxation due to an integrality gap example from [4]. The problem considered in [4] is an uncapacitated version of the AdCell problem, that is there is no capacity constraint (C) on the customers. Capacity constraint restricts how many queries/advertisements can be assigned to each customer. We can represent all the queries associated with each customer as a set; these sets are therefore disjoint and has integer hard capacities associated with them. Our approximation ratio matches the best known bound from [4,21] for the uncapacitated case. For space limitation, most of the details have been moved to the full paper. Here, we give a high-level description of the algorithm. Our algorithm is based on applying the rounding technique of [20] through several iterations. The essential idea of the proposed rounding is to apply a procedure called **Rand-move** to the variables of a suitably chosen subset of constraints from the original linear program. These sub-system must be underdetermined to ensure that the rounding proceeds without violating any constraint and at least one variable becomes integral. The trick lies on choosing a proper sub-system at each step of rounding, which again depends on a detailed case analysis of the LP structure.

Let y^* denote the LP optimal solution. We begin by simplifying the assignment given by y^*. Consider a bipartite graph $G(\mathcal{B}, \mathcal{I}, E^*)$ with advertisers $\mathcal{B}$ on one side, queries $\mathcal{I}$ on the other side and add an edge (i, j) between a advertiser i and query j, if $y^*_{i,j} \in (0, 1)$. That is, define $E^* = \{(i, j) | \ 1 > y^*_{i,j} > 0\}$. Our first claim is that y^* can be modified without affecting the optimal fractional value and the constraints such that $G(\mathcal{B}, \mathcal{I}, E^*)$ is a forest. The proof follows from Claim 2.1 of [4]; we additionally show that such assumption of forest structure maintains the capacity constraints.

Lemma 2. *Bipartite graph $G = (\mathcal{B}, \mathcal{I}, E^*)$ induced by the edges E^* can be converted to a forest maintaining the optimal objective function value.*

Proof. Consider the graph $G = (\mathcal{B}, \mathcal{I}, E^*)$ and consider one connected component of it. We will argue for each component separately and similarly.

Cycle Breaking: Suppose there is a cycle in the chosen component. Since G is bipartite, the cycle has even length. Let the cycle be $C = \langle i_1, j_1, i_2, j_2, \ldots, i_l, j_l, i_1 \rangle$, that is consider the cycle to start from a advertiser node. Consider a strictly positive value α and consider the following update of the y^* values over the edges in the cycle C. We add $z_{a,b}$ to edge (a, b), where

R1. $z_{i_1,j_1} = -\beta$
R2. If we are at an query node j_t, $t \in [1, l]$, then $z_{j_t,i_{t+1}} = -z_{i_t,j_t}$
R3. If we are at a advertiser node i_t, $t \in [1, l]$, then $z_{i_t,j_t} = -\frac{b_{i_t,j_{t-1}} z_{j_{t-1},i_t}}{b_{i_t,j_t}}$

β is chosen such that after the update, all the variables lie in $[0, 1]$ and at least one variable gets rounded to 0 or 1, thus the cycle is broken. Note that the entire update is a function of z_{i_1,j_1}. For any query node, its total contribution in (Assign) constraint of LP1 remains unchanged. For any advertiser node, except i_1, its contribution in (Advertiser) constraint and thus in the objective function remains the same. In addition, since the assign constraints remain unaffected, all the capacity constraints are satisfied. For advertiser i_1, its contribution decreases by $z_{i_1,j_1} b_{i_1,j_1}$ and increases by $z_{j_l,i_1} b_{i_1,j_l} = z_{i_1,j_1} b_{i_1,j_l} \frac{b_{i_2,j_1} b_{i_3,j_2} \ldots b_{i_{l-1},j_{l-2}}}{b_{i_2,j_2} b_{i_3,j_3} \ldots b_{i_{l-1},j_{l-1}}}$.
If $b_{i_1,j_1} \leq b_{i_1,j_l} \frac{b_{i_2,j_1} b_{i_3,j_2} \ldots b_{i_{l-1},j_{l-2}}}{b_{i_2,j_2} b_{i_3,j_3} \ldots b_{i_{l-1},j_{l-1}}}$, then instead of adding z_{j_l,i_1} on the last edge, we add some $c < z_{j_l,i_1}$ such that $z_{i_1,j_1} b_{i_1,j_1} = c b_{i_1,j_l}$. Thus, we are able to maintain the objective function exactly. The assign constraint on the last query j_l can only decrease by this change and hence all the capacity constraints are maintained as well.

Otherwise, $b_{i_1,j_1} > b_{i_1,j_l} \frac{b_{i_2,j_1} b_{i_3,j_2} \ldots b_{i_{l-1},j_{l-2}}}{b_{i_2,j_2} b_{i_3,j_3} \ldots b_{i_{l-1},j_{l-1}}}$. In that case, we traverse the cycle in the reverse order, that is, we start by decreasing on z_{i_1,j_l} first and proceed similarly.

Once, we have such a forest structure, several cases arise and depending on the cases, we define a suitable sub-system on which to apply the rounding technique. There are three major cases.

(i) There is a tree with two leaf advertiser nodes: in that case, we show that applying our rounding technique only diminishes the objective function by little and all constraints are maintained.

(ii) No tree contains two leaf advertisers, but there is a tree that contains one leaf advertiser: we start with a leaf advertiser and construct a path spanning several trees such that we either end up with a combined path with advertisers on both side or a query node in one side such that the capacity constraint on the set containing that query is not met with equality (non-tight constraint). This is the most nontrivial case and a detailed discussion is given in the full paper.

(iii) No tree contains any leaf advertiser nodes: in that case we again form a combined path spanning several trees such that the queries on two ends of the combined path come from sets with non-tight capacity constraints.

References

1. M. Agrawal. Overview of mobile advertising. `http://www.telecomcircle.com/2009/11/overview-of-mobile-advertising`.
2. N. Andelman and Y. Mansour. Auctions with budget constraints. In *In 9th Scandinavian Workshop on Algorithm Theory (SWAT)*, 2004.
3. Y. Azar, B. Birnbaum, A. R. Karlin, C. Mathieu, and C. T. Nguyen. Improved approximation algorithms for budgeted allocations. *ICALP '08*. 186-197, 2008.
4. D. Chakrabarty and G. Goel. On the approximability of budgeted allocations and improved lower bounds for submodular welfare maximization and gap. *SIAM J. Comp.* p2189, 2010.
5. J. Chuzhoy and J. S. Naor. Covering problems with hard capacities. *FOCS '02*. p481, 2002.
6. N. R. Devenur and T. P. Hayes. The adwords problem: online keyword matching with budgeted bidders under random permutations. *EC '09*. 71-78, 2009.
7. J. Feldman, A. Mehta, V. Mirrokni, and S. Muthukrishnan. Online stochastic matching: Beating 1-1/e. *FOCS '09*. 117-126, 2009.
8. R. Gandhi, E. Halperin, S. Khuller, G. Kortsarz, and A. Srinivasan. An improved approximation algorithm for vertex cover with hard capacities. *ICALP '03*. 164-175, 2003.
9. G. Goel and A. Mehta. Online budgeted matching in random input models with applications to adwords. *SODA '08*. 982-991, 2008.
10. M. T. Hajiaghayi, R. Kleinberg, and D. C. Parkes. Adaptive limited-supply online auctions. *EC '04*. 71-80, 2004.
11. M. T. Hajiaghayi, R. D. Kleinberg, and T. Sandholm. Automated online mechanism design and prophet inequalities. In *AAAI*, 2007.
12. C. Karande, A. Mehta, and P. Tripathi. Online bipartite matching with unknown distributions. In *STOC*, 2011.
13. R. M. Karp, U. V. Vazirani, and V. V. Vazirani. An optimal algorithm for on-line bipartite matching. *STOC '90*. 352-358, 1990.
14. U. Krengel and L. Sucheston. Semiamarts and finite values. *Bull. Am. Math. Soc.*, 1977.
15. S. Lahaie. An analysis of alternative slot auction designs for sponsored search. *EC '06*. 218-227, 2006.
16. M. Mahdian and Q. Yan. Online bipartite matching with random arrivals: An approach based on strongly factor-revealing lps. In *STOC*, 2011.
17. V. H. Manshadi, S. O. Gharan, and A. Saberi. Online stochastic matching: Online actions based on offline statistics. *SODA '11*, 2011.
18. A. Mehta, A. Saberi, U. Vazirani, and V. Vazirani. Adwords and generalized online matching. *J. ACM*, 54, 2007.
19. M. Pál, E. Tardos, and T. Wexler. Facility location with nonuniform hard capacities. *FOCS '01*. 329-338, 2001.
20. B. Saha and A. Srinivasan. A new approximation technique for resource-allocation problems. *ICS '10*. 342-357, 2010.
21. A. Srinivasan. Budgeted allocations in the full-information setting. *APPROX '08*. 247-253, 2008.
22. S. Wang, J. Min, and B. K. Yi. Location based services for mobiles: Technologies and standards. *ICC '08*, 2008.
23. F. Zahradnik. Garmin to offer free, advertising-supported traffic detection and avoidance. `http://gps.about.com/b/2008/09/15/garmin-to-offer-free-advertising-supported-traffic-detection-and-avoidance.htm`.