Advertisement Allocation for Generalized Second Pricing Schemes

Ashish Goel[*]   Mohammad Mahdian[†]   Hamid Nazerzadeh[*‡]   Amin Saberi[*]

Abstract

Recently, there has been a surge of interest in algorithms that allocate advertisement space in an online revenue-competitive manner. Most such algorithms, however, assume a pay-as-you-bid pricing scheme. In this paper, we study the query allocation problem where the ad space is priced using the well-known and widely-used generalized second price (GSP) scheme. We observe that the previous algorithms fail to achieve a bounded competitive ratio under the GSP scheme. On the positive side, we present online constant-competitive algorithms for the problem.

1 Introduction

In the last few years, there has been a lot of interest in online auctions especially in the context of sponsored search. This is not surprising considering that search engines hold thousands of individual auctions every second for the allocation of advertisement next to their search results. The process of choosing and charging the advertisers is a daunting algorithmic and engineering task. The search engines typically take into consideration several factors including the search keyword, the demographics of the user, the frequency of the keyword, as well as the bid, budget and click-through rate of the advertisers for each of these decisions.

Modeling and analyzing these auctions have been a challenge as well: the repeated nature of the auctions as well as budget constraints make the structure of the equilibria rich and complicated. Moreover, misconceptions of the first designers of these mechanisms about the Vickrey auction have lead to decisions that add to this difficulty. Nevertheless, computer scientists and economists have suggested and analyzed several models each capturing a particular aspect of these mechanisms. We start by briefly mentioning two lines of research most related to the present paper. For a more detailed exposition see Lahaie et al. [8].

The static model focuses on a single auction in which a small number (typically less than 20) slots are being auctioned at the same time. The higher slots are more desirable because they receive more clicks. The bid or private information of a bidder is one dimensional and is the expected payoff from a click. The expected payoff to a bidder from not obtaining a slot is assumed to be zero. As is the case for most search engines, the auctioneer runs what is known as the Generalized Second Price (GSP) auction. GSP uses the same allocation rule as VCG: the bidders are sorted based on their expected value (bid times click-through rate) and receive the slots in that order. The expected payment of the bidder who receives a slot is the expected value of the next highest bidder for that slot. It is easy to see that unlike VCG, GSP auctions are not truthful. Varian [12] and Edelman et al. [4] analyze the equilibria of this auction under some additional assumptions, and show that GSP has an equilibrium that is revenue equivalent to the VCG outcome.

Another line of research focuses on the algorithmic aspects of the allocation problem from an online competitive analysis point of view. In this model, the search engine receives the bids of advertisers and their maximum budget for a certain period (e.g. a day). As users search for these keywords during the day, the search engine assigns their ad space to advertisers and charges them the value of their bid for the impression

---

[*]Management Science and Engineering Dept., Stanford Univesrity, CA. Email: {ashishg,hamidnz,saberi}@stanford.edu

[†]Yahoo! Research, Santa Clara. CA, Email: mahdian@yahoo-inc.ocom

[‡]Work performed in part at Yahoo! Research.

of the ad. That model was studied first by Mehta et al. [11] who showed that greedy achieves a competitive ratio of $\frac{1}{2}$ and gave a $1 - 1/e$ competitive algorithm. They also showed that this ratio is essentially tight. Buchbinder et al. [3] gave a primal-dual algorithm and analysis for this problem. They as well as Mahdian et al. [9] also extended that framework to scenarios in which additional information is available, yielding improved worst case competitive ratios. See also Goel and Mehta [6, 7] for other results on this model. It is possible to extend most of the results of these papers to a multi-slot auction but essentially only under a pay-as-you-bid pricing scheme, and not the GSP.

The focus of this paper is on the algorithmic aspects of the online allocation problem, where the search engine is committed to charge under the GSP scheme. We first observe that different policies for selecting advertisers to participated in a keyword auction can significantly affect the revenue. The GSP mechanism is the current industry standard for allocating and pricing the ad slots; however, often to optimize the allocation of the budget or to merely smooth out the allocations throughout the day, search engines use a *throttling algorithm*, i.e., a mechanism that upon the arrival of a new query, decides which advertisers will be allowed to participate in the auction for that query. Once the subset is chosen the search engine is committed to the GSP scheme for sorting and charging the advertisers in that set.

We consider three different models for throttling. We refer to the first and the simplest of these models as the *non-throttling model*. In this model *all* advertisers with positive remaining budget participate in the keyword auction. We give a simple example which shows that removing an advertiser from the auction can significantly increase the revenue. This implies that the revenue in the non-throttling model can be arbitrary low.

In the second model, we consider policies in which the auctioneer is allowed to exclude advertisers with positive remaining budget from some of the auctions but it has to remove an advertiser as soon as her budget reaches zero. We call this throttling model the *strict model*, and the throttling algorithms that fit this model strict algorithms. To the best of our knowledge, all throttling algorithms used by search engines are strict. If the frequencies of the queries are known in advance, i.e., the offline setting, the optimal strict allocation can be found using linear programming [1]. In the online setting, first we observe that both the greedy algorithm and the algorithm of Mehta et al. [11] fail to be competitive against an optimal offline strict algorithm. In this paper, we give the first constant-competitive online algorithm for query allocation in the strict model. Our algorithm is a greedy algorithm that at each step solves a dynamic program as a subroutine. The competitive ratio of the algorithm is $\frac{1}{3}$. In fact, we show that the algorithm is competitive against a stronger benchmark.

We observer that allowing an allocation algorithm to give out "free impressions" can also increase the revenue. Hence, we consider a throttling model in which the algorithm can choose any set of advertisers, whether their budget is exhausted or not, and pass it to the GSP mechanism. However, an advertiser cannot be charged more than her budget, so by including an advertiser whose budget is exhausted, the throttling algorithm decides to give her a *free impression*. We refer to this model as the *non-strict model*. Policies in the strict model also belong to the non-strict model. Therefore, the revenue under non-strict model is greater than or equal to the strict model. The idea of increasing the revenue by giving out free impression is reminiscent of the idea of subsidizing weak bidders to increase revenue in auctions [10, 2]. Of course, the non-strict model is not practical because it creates wrong incentives for the bidders. But it provides a good benchmark for comparing the performance of various allocation algorithms. We show that the maximum ratio between the optimal offline solution in the strict and the non-strict models is 2. We also show that the algorithm of Mehta et al. can be combined with a dynamic programming subroutine to obtain a $(1 - \frac{1}{e})$ competitive ratio for this model. Further, we prove that our strict greedy algorithm is $\frac{1}{3}$-competitive with respect to the optimal offline solution in the non-strict algorithms.

Organization and Results In the next section, we formally define the problem, and present a linear programming formulation for it. In Section 3, we compare the optimal revenue in different throttling models. We show that the revenue of non-throttling model can be arbitrary worse than the revenue of the optimal

2

strict model. However, the maximum ratio between the optimal revenue in the non-strict and the strict models is 2. We present a greedy strict algorithm in Section 4. Surprisingly, this algorithm is $\frac{1}{3}$-constant competitive with respect to both optimal offline strict and non-strict algorithms. We give an example which shows this ratio is tight. In Section 5, we show that the greedy algorithm and the algorithm of Mehta et al respectively achieve competitive ratio of $\frac{1}{2}$ and $(1 - \frac{1}{e})$ in the non-strict model.

2 Model

We study a sponsored search setting where m advertisers are bidding to be displayed alongside the search results for a number of different keywords. Let $\mathcal{A}$ be the set of advertisers, and $\mathcal{K}$ be the set of keywords. During the day, a sequence $\mathcal{Q}$ of search queries arrive, each corresponding to one keyword in $\mathcal{K}$. As a query arrives, the search engine picks an ordered list of k ads (sometimes called a *slate* of ads) to be displayed in the k ad slots numbered $1, 2, \ldots, k$. We follow the common assumption that slot number 1 is the "best" slot for all advertisers (typically the "top" slot), and the value of other slots scale in the same way for all advertisers.[1] Formally, this means that there are constants $1 = \theta_1 \geq \theta_2 \geq \cdots \geq \theta_k$ such that if an advertiser has a value of x for being displayed in the first slot, her value for being displayed in slot ℓ is $\theta_\ell x$. Therefore, each advertiser i only needs to specify their bid b_{ij} that represents the maximum she is willing to pay to be displayed in slot 1 for keyword j. In addition to these bids, the advertisers can specify their total budget for a certain period (e.g. a day). We denote the daily budget of advertiser i by B_i. As in the previous work [11, 3, 9, 6], we assume that bids are small compared to the budgets. This assumption is quite reasonable in practice.

For the sake of transparency and also for technical reasons, search engines often commit to using a simple mechanism for allocating slots to advertisers and determining how much each advertiser should pay. The current industry standard is a mechanism called the *generalized second-price* (GSP) auction[4, 12]. This mechanism sorts advertisers based on their bids, and assigns the first k advertisers to the k available slots in this order. The price for the ℓ'th advertiser, $1 \leq \ell \leq k$, is equal to θ_ℓ times the bid of the next advertiser for this keyword (i.e., the advertiser in slot $\ell + 1$ if $\ell < k$, or the first advertiser that is not displayed). The search engine charges each advertiser the minimum of her price and her remaining budget.[2]

To optimize the allocation of the budget or to merely smooth out the allocation of advertisers throughout the day, search engines often use a *throttling algorithm*, i.e., a mechanism that upon the arrival of a new query, decides which advertisers will be allowed to participate in the auction for that query. The output of this algorithm is a set of advertisers. These advertisers will be sorted as in the GSP mechanism, the slots will be allocated to the advertisers in this order, and price for the i'th advertiser in this order equal to θ_i times the bid of the $i + 1$'st advertiser for the same keyword (for simplicity, assume $\theta_i = 0$ for $i > k$). The goal is to design a throttling algorithm (which we also refer to as an allocation algorithm) that generates the maximum total revenue for the search engine.

We consider three different models for throttling. The first two models impose some restrictions on advertisers that are allowed to participated in a keyword auction. The third model has no restrictions.

- In the *non-throttling model* an advertisers is removed from the auction only, and if only, her budget is exhausted. Under the GSP scheme, the query allocation algorithm is trivial in this model: Allocate the query to the advertisers with k highest bid; then, charge advertisers according to the GSP, and remove advertisers with exhausted budget.

- In the *strict model*, as soon as the amount an advertiser is charged reaches her budget, the system

---

[1]In reality, search engines charge the advertisers per click instead of per impression. However, because of the large number queries in practice, the pay-per-click system performs similar to a pay-per-impression system in expectation, where the value per impression is the value per click times the *click-through rate* (CTR). The assumption that the value-per-impression for different slots scales the same way for different advertisers is equivalent to the widely-used assumption of *separability* of CTRs. Our results work in a slightly more general setting where the scaling factors $\theta_{j\ell}$ can also depend on the keyword j.

[2] Given the assumption that bids are small compared to budgets, these "last query discounts" are negligible.

removes the advertiser. However, a throttling algorithm in this model can also remove advertisers with positive budget from the auction.

- In the *non-strict model*, the budget enforcement is left to the throttling algorithm. The algorithm can choose any set of advertisers, whether their budget is exhausted or not, and pass it to the GSP mechanism. However, an advertiser cannot be charged more than her budget, so by including an advertiser whose budget is exhausted, the throttling algorithm decides to give her a *free impression*.

The main difficulty faced by the throttling algorithm is that it does not know the sequence $\mathcal{Q}$ of queries that arrive during the day; rather, the sequence is revealed to the algorithm in an online fashion. Below, we show how the allocation problem can be formulated as a linear program in the offline case, i.e., when query volumes are known in advance.

2.1 Offline setting: The LP formulation

As mentioned, the query allocation algorithm is trivial in the non-throttling model. Abrams et al. [1] present a linear program to solve the offline problem in the strict model. In this section, we give a similar linear program for the non-strict model.

Let $\mathcal{S}_j$ be the family of all subsets of advertisers who are interested in keyword j. For a keyword j, a set $S \in \mathcal{S}_j$, and an advertiser $i \in S$, we define $p_{GSP}(i, j, S)$ as the price for advertiser i in a GSP auction for keyword j, when the set of advertisers participating in the auction is S. For simplicity of notation, when it is clear that the GSP mechanism is used, we denote the price by $p(i, j, S)$; also let $p(i, j, S) = 0$ for $i \notin S$. Define $\pi(j, S)$ to be the revenue of the search engine from allocating keyword j to set S, i.e.,

$$\pi(j, S) = \sum_{i \in S} p(i, j, S). \tag{1}$$

Let n_j be the number of the queries for keyword j that arrive during the day. For every keyword j and set $S \in \mathcal{S}_j$, the linear program has a variable $x_{j,S}$ indicating the number of queries for keyword j to whom the set S is assigned to. To formulate the non-strict model, we need an additional variable y_i that indicates the total value of queries allocated for free to advertiser i. The primal linear program in the non-strict model is as follows:

$$
\begin{aligned}
\text{maximize} \quad & \sum_{j \in \mathcal{K}} \sum_{S \in \mathcal{S}_j} \pi(j, S) x_{j,S} - \sum_i y_i \\
\text{subject to} \quad & \sum_{S \in \mathcal{S}_j} x_{j,S} \leq n_j && \forall j \in \mathcal{K} \\
& \sum_{j \in \mathcal{K}} \sum_{S \in \mathcal{S}_j} p(i, j, S) x_{j,S} - y_i \leq B_i && \forall i \in \mathcal{A} \\
& x_{j,S} \geq 0 && \forall j \in \mathcal{K}, S \in \mathcal{S}_j \\
& y_i \geq 0 && \forall i \in \mathcal{A}
\end{aligned}
$$

The dual of this program can be written as follows:

$$
\begin{aligned}
\text{minimize} \quad & \sum_{j \in \mathcal{K}} n_j \alpha_j + \sum_{i \in \mathcal{A}} \beta_i B_i \\
\text{subject to} \quad & \alpha_j + \sum_{i \in S} p(i, j, S) \beta_i \geq \pi(j, S) && \forall j \in \mathcal{K}, S \in \mathcal{S}_j \\
& \beta_i \leq 1 && \forall i \in \mathcal{A} \\
& \alpha_j, \beta_i \geq 0 && \forall i \in \mathcal{A}, j \in \mathcal{K}
\end{aligned}
$$

4

The linear programming formulation of the strict model can be obtained from the above primal program by setting all y_i's to zero. This corresponds to removing the $\beta_i \leq 1$ constraints from the dual.

Abrams et al. [1] proposed a solution based on the simplex algorithm and the column generation method. Alternatively, one can obtain a polynomial time algorithm for this linear program using the ellipsoid method in combination with a separation oracle based on the dynamic programming subroutine, see Section 5.

3 Effects of different throttling models on the revenue

We start this section with two examples. In the first example we observe that removing an advertiser from an auction can significantly increase the revenue. This implies that the revenue of the non-throttling model can be arbitrary low. The second example shows that sometimes giving out queries for free increases the revenue. However, as proved in Theorem 4, the increase in the revenue is essentially bounded by 2.

Example The example consists of only one keyword and three advertisers. The first advertiser has an unlimited budget, and a bid of \$2 for each impression. The second advertiser has a bid of \$1 for each impression and a budget of \$1. The third advertiser, has a bid of \$$\epsilon < 1$ and an unlimited budget. Assume $k = 3$ and $\theta_1 = \theta_2 = \theta_3 = 1$. In the non-throttling model, at the beginning, all advertisers participate in the auction. However, after $\lceil \frac{1}{\epsilon} \rceil$ impressions, the budget of the second advertiser will be exhausted and this advertiser will be removed from the auction. Therefore, for the rest of the day, each query will bring a revenue of \$$\epsilon$. However, if an algorithm, from the beginning, removes advertiser 3 from the auction, each query will bring a revenue of \$1. The ratio between the revenue of the (strict) algorithm that removes the third advertiser and the non-throttling algorithm approaches $\frac{1}{\epsilon}$, as the number of queries increases.

Corollary 1 *The ratio between the optimal revenue in the strict and the non-throttling models can be made arbitrary large.*

The next example shows that giving out queries for free may increase the revenue.

Example For a large integer M, there are $M + 2$ advertisers, one keyword, and $M + 2$ slots. Assume $\theta_i = 1$ for all slots. The bid of advertisers 1 and 2 is equal to 1. Advertisers $3, \ldots, M + 2$ bid $\frac{1}{M}$. The budget of all advertisers except advertiser 2 is infinite. Since the budget of advertiser 2 is limited, she will eventually run out of budget. After that, the maximum revenue any strict algorithm can obtains is 1 (by allocating the query to set $\{1, 2\}$ or set $\{1\} \cup \{3, \cdots, M + 2\}$). However, the optimal non-strict algorithm keeps 2 in the auction, and obtains revenue $2 - \frac{1}{M}$ from each query. In this case advertiser 2 receives queries for free.

Corollary 2 *The ratio between the optimal revenue of the non-strict and the strict models can be made as large as* 2.

In the rest of this section, we provide an upper bound on this ratio. For keyword j and set $S \in \mathcal{S}_j$, we call advertiser $i \in S$ an *active* advertiser if her remaining budget is greater than $p(i, j, S)$. We define set $S \in \mathcal{S}_j$ a *proper set* if all advertiser $i \in S$ are active. Also, let $\pi^{(A)}(j, S)$ be the revenue from active advertisers in S, i.e., the sum of the prices of active advertisers.

Lemma 3 *For every keyword j and set S, there exists a proper subset $S' \subset S$ such that*

$$\pi^{(A)}(j, S') \geq \frac{1}{2}\pi(j, S).$$

Proof : Without loss of generality, assume $S = \{1, 2, 3, \ldots, \ell\}$, and $b_1 \geq b_2 \geq \ldots \geq b_\ell$. Let T be the set of all active advertisers in S. If advertiser 1 is inactive, then removing that advertiser from S can only increase

$\pi(j, S)$; hence, assume that advertiser 1 is active. If S is proper the claim trivially holds. Otherwise, let q be the inactive advertiser in S with the largest bid.

Now consider two subsets $T_1 = \{1, 2, \ldots, q\}$ and $T_2 = \{i \in T | i > q\}$. Note that the price for q is zero in T_1; therefore T_1 is proper. Also, because any subset of a proper set is proper, T_2 is proper. We have

$$\pi^{(A)}(j, S) = \sum_{i \in T} p(i, j, S) = \sum_{i < q} \theta_i b_{i+1} + \sum_{i \in T_2} \theta_i b_{i+1}$$
$$= \pi(j, T_1) + \sum_{i \in T_2} \theta_i b_{i+1} \quad (2)$$

For simplicity $b_{\ell+1}$ is defined to be zero. We now show

$$\sum_{i \in T_2} \theta_i b_{i+1} \leq \pi(j, T) \quad (3)$$

Because $b_{i+1} \leq b_i$, we get:

$$\sum_{i \in T_2} b_{i+1} \theta_i \leq \sum_{i \in T_2} b_i \theta_i. \quad (4)$$

Let d_i denote the number of inactive advertisers before advertiser i in S. For any advertiser $i \in T$, let $n(i)$ denote the next advertiser after i in T. The position of i in the set T will be $i - d_i$, and hence we have:

$$\sum_{i \in T} p(i, j, T) = \sum_{i \in T} b_{n(i)} \theta_{i-d_i}$$

Since θ is decreasing, we have $\theta_{i-d_i} \geq \theta_i \geq \theta_{n(i)}$. Therefore, we get:

$$\sum_{i \in T} p(i, j, T) \geq \sum_{i \in T} b_{n(i)} \theta_{n(i)} \geq \sum_{i \in T_2} b_i \theta_i \quad (5)$$

The last inequality is followed from $T_2 \subset T$. Combining inequalities 4 and 5 gives us inequality 3. Plugging into 2 we get

$$\pi^{(A)}(j, S) = \pi(j, T_1) + \pi(j, T),$$

which completes the proof. □

Theorem 4 *Let δ be the maximum ratio between the bid and the budget of an advertisers. The ratio of the revenues of optimal offline algorithms in the non-strict and the strict models is at most $2 + \frac{\delta}{1-\delta}$. Also, this ratio can be as large as 2.*

Proof :   The lower bound is immediate from Corollary 2. To prove the upper bound, consider an optimal non-strict offline algorithm. We construct a strict algorithm based on this non-strict algorithm called $\mathcal{P}$ (we refer to both algorithm and allocation as $\mathcal{P}$). Suppose the optimal strict algorithm allocates query j to set S. Algorithm $\mathcal{P}$ allocates j to a proper subset of S with the maximum revenue. Active advertisers, and hence proper sets, are determined with reference to remaining budgets in $\mathcal{P}$.

As proved in Lemma 3, the revenue of the non-strict algorithm from the active advertisers for each query is at most twice the revenue of $\mathcal{P}$. In the following we prove that the total revenue of the non-strict algorithm from inactive advertisers is at most a $\frac{\delta}{1-\delta}$ fraction of the revenue of $\mathcal{P}$, which completes the proof.

Note that the set of advertisers that receive a query in $\mathcal{P}$ is a subset of advertisers allocated the query by the non-strict algorithm. Hence, if an advertiser is inactive, it has spent at least $1 - \delta$ fraction of her

budget, both in $\mathcal{P}$ and the non-strict algorithm. Therefore, the total revenue of the non-strict algorithm from inactive advertisers is at most a δ times $\frac{1}{1-\delta}$ fraction of the revenue of $\mathcal{P}$. □

Given the assumption that bids are small compared to budgets, we expect that δ to be small. Therefore, the theorem above essentially gives a tight bound on the maximum ratio between the revenue of the two throttling models.

4   An online strict algorithm

In this section we present an online algorithm for the strict model under the GSP scheme. Note that the greedy algorithm (for the pay-as-you-bid pricing scheme) is the same as the non-throttling algorithm, so by Corollary 1, it fails to get a good competitive ratio in the strict model. In the next section we also show that the algorithm of Mehta et al is not competitive in this model. Our strict algorithm makes greedy choices at each step and achieves competitive ratio $\frac{1}{3}$, even when compared to the offline *non-strict* optimal algorithm. Further, we will show that the analysis of the strict greedy algorithm is tight, even against a strict offline optimal.

The strict greedy algorithm:

Upon the arrival of a new query for keyword j:

> Allocate the query to a *proper* set with maximum revenue,
> i.e., proper set S which maximizes $\pi(j, S)$.

Lemma 5 *Proper set S with maximum revenue can be found using dynamic programming.*

Proof :   Without loss of generality, assume for $i \geq 1$, $b_i \geq b_{i+1}$. Define $r_{u,l}$ to be the maximum revenue can be obtained from slots 1 to l when u is in the slot $l+1$. We can compute $r_{u,l}$ using the dynamic program below:

$$r_{u,l} = \begin{cases} \max_{\{v < u:\ b_u \theta_l >\text{ remaining budget of } v\}} \{r_{v,l-1} + b_u \theta_l\} & l < u \\ 0 & l \geq u \text{ or } l = 1 \end{cases} \tag{6}$$

The time complexity of this dynamic program is $O(|\mathcal{A}|^2 k)$. □

Theorem 6 *Let δ be the maximum ratio between the bid and the budget of an advertisers. The strict algorithm is $\left(\frac{1-\delta}{3-2\delta}\right)$-competitive with respect to the optimal offline solution for the non-strict model which is allowed to use (but not charge) advertisers whose budget have been exhausted.*[3]

Proof :   Consider an optimal non-strict algorithm denoted by OPT. Let $\pi_G(t)$ and $\pi_{\text{OPT}}(t)$ denote the total revenue obtained by the strict greedy algorithm and OPT at time t. Also, let $\pi_{\text{OPT}}^{(A)}(t)$ (resp. $\pi_{\text{OPT}}^{(I)}(t)$) denote the total revenue obtained by OPT from advertisers who are active (resp. inactive) at time t. In these definitions and in the rest of the proof, when we call an advertiser active/inactive at time t, it is always with

---

[3] Note that the optimal revenue in the non-strict model is greater than or equal to the revenue in the strict model.

reference to the strict greedy algorithm; the same is true for proper sets. We will relate $\pi^{(I)}_{\text{OPT}}(t)$ and $\pi^{(A)}_{\text{OPT}}(t)$ separately to the revenue of the strict greedy algorithm, $\pi_G(t)$.

First, observe that any advertiser who contributes to $\pi^{(I)}_{\text{OPT}}(t)$ must have already used up a fraction $(1-\delta)$ of her budget in the strict greedy algorithm at time t. Hence, we have:

$$\sum_t \pi^{(I)}_{\text{OPT}}(t) \leq \sum_t \pi_G(t)/(1-\delta) \tag{7}$$

Also, by Lemma 3, and because the greedy algorithm allocated the query to a proper set with maximum revenue, we have:

$$\pi^{(A)}_{\text{OPT}}(t) \leq 2\pi_G(t) \tag{8}$$

The claim immediately follows from inequalities 7 and 8. $\qquad\square$

4.1 A tight example

We will now prove that our analysis for the strict greedy algorithm is tight, even against a strict offline optimal algorithm. Consider a scenario with k slots, and $3 + 2k$ advertisers $A_1; A_2; A_3;$ $B_1, B_2, \ldots, B_k;$ $C_1, C_2, \ldots, C_k$. Advertiser A_1 has budget N; A_2, A_3 have budgets N/k each; $B_1, B_2, \ldots, B_k$ have budgets $N/(k-1)$ each; $C_1, C_2, \ldots, C_k$ have budgets N/k each. The queries arrive in three phases; each phase has N identical queries. We summarize the queries, the bids, and the actions taken/profit obtained by the strict greedy algorithm as well as an offline algorithm in table 1. The total revenue earned by the online algorithm is $N + O(N/k)$ while the total revenue of the offline algorithm is $3N - O(N/k)$, giving a lower bound of $\frac{1}{3}$ on the competitive ratio.

Phase	Bids	Greedy ranking	Greedy revenue	Offline ranking	Offline revenue
1	$A_2 : 2/K; A_3 : 1/K$	A_2, A_3	N/k	Passes	0
2	$A_1 : 2; A_2 : 1$ $B_1 \ldots B_k : 1/(k-1)$ $C_1 \ldots C_k : 1/k$	$A_1, B_1 \ldots B_k$	$Nk/(k-1)$	$A_1, A_2, C_1 \ldots C_{k-1}$	$N(1 + \frac{k-1}{k})$
3	$B_1 \ldots B_k : 1/(k-1)$	$B_1 \ldots B_k$ (if budget left)	$N/(k-1)$	$B_1 \ldots B_k$	N

Table 1: An example showing that the analysis of greedy is tight. Each phase has N queries. When we do not specify the bid of an advertiser for a query, it is assumed to be 0.

5 Online algorithms for the non-strict model

In this section, we show that the greedy algorithm is $\frac{1}{2}$-competitive in the non-strict model. But first, we present the modified algorithm of Mehta et al. [11] for the GSP scheme. Define function $\Phi : [0, 1] \to [0, \frac{e-1}{e}]$ to be $\Phi(x) = 1 - e^{x-1}$. Also, let f_i be the fraction of the *spent* budget of advertiser i.

The (modified) algorithm of Mehta et al. for the GSP scheme:

Upon the arrival of a new query for keyword j:

Allocate the query to a set $S \in \text{argmax}_{S \subset \mathcal{S}_j} \sum_{i \in S} \Phi(f_i)p(i, j, S)$.

We develop a primal-dual scheme to analyze this algorithm which is similar to the approach proposed by Buchbinder et al. [3]. But, before the analysis, we first observe that this algorithm cannot be used under the strict model. The reason is that set S which maximize $\sum_{i \in S} \Phi(f_i) p(i, j, S)$ may contain advertisers with no remaining budget. For instance, consider the first example in Section 3. In the strict model, the algorithm presented above allocates each query to the same set of advertisers as the non-throttling algorithm. Hence, by Corollary 1, it fails to be competitive. However, we prove that the algorithm is $(1 - \frac{1}{e})$-competitive in the non-strict model.

Theorem 7 *The non-strict algorithm is $(1 - \frac{1}{e})$-competitive with respect to the optimal offline solution of the non-strict model.*

Proof : In the online setting, without loss of generality, we assume that there is one query from each keyword, i.e., $n_j = 1$ for every keyword j. We construct a feasible solution for the primal and the dual. We describe how to update the primal and dual variables, after arrival of each new query, such that it maintains the feasibility; also, the ratio of the values of the primal and the dual be greater than or equal to $1 - \frac{1}{e}$. Therefore, by the end of the algorithm, we have a solution which is within a ratio of $1 - \frac{1}{e}$ of the optimal solution of the primal linear program.

We initialize all of the variables to zero, and update them once a query shows up. In particular we let $\beta_i = \frac{e^{f_i} - 1}{e - 1}$, which implies $1 - \beta_i = \frac{e}{e-1} \Phi(f_i)$.

Consider query j, and assume that the algorithm allocates it to set S which maximizes:

$$\sum_{i \in S} (1 - \beta_i) p(i, j, S)$$

Let α_j equal to this maximum. Therefore, the dual remains feasible.

Recall that $\beta_i = \frac{e^{f_i} - 1}{e - 1}$; hence, β_i is updated multiplicatively as long as $\beta_i < 1$. By the assumption that bids are small compared to the budgets, the increase in β is equal to:

$$\frac{c_i}{B_i} (\beta_i + \frac{1}{e - 1})$$

where $c_i = \min\{p(i, j, S), (1 - f_i) B_i\}$ is the amount i is charged. Also, we increase y_i by $p(i, j, S) - c_i$. Therefore, the primal remain feasible; and its value is increased by $\sum_{i \in S} \pi(j, S) - y_i = \sum_{i \in S} c_i$.

The increase in the value of the dual is equal to:

$$\alpha_j + \sum_{i \in S} (\beta_i + \frac{1}{e - 1}) \frac{c_i}{B_i} B_i = \sum_{i \in S} (1 - \beta_i) c_i + \sum_{i \in S} (\beta_i + \frac{1}{e - 1}) c_i$$
$$= \frac{e}{e - 1} \sum_{i \in S} c_i$$

Therefore, the value of the primal remains within a $\frac{e-1}{e}$ ratio of the value of the dual, which completes the proof. $\qquad \square$

Note that the above proof works for any pricing scheme, and is not specific to GSP. Therefore, as long as we can find a set $S \in \text{argmax}_{S \in \mathcal{S}_j} \{\sum_{i \in S} \Phi(f_i) p(i, j, S)\}$ in polynomial time, the above theorem guarantees a competitive ratio of $1 - 1/e$. For GSP, such a set can be find in polynomial time using dynamic programming similar to Lemma 5. The proof of the following lemma is omitted.

Lemma 8 *Set $S \in \text{argmax}_{S \subset \mathcal{S}_j} \sum_{i \in S} \Phi(f_i) p(i, j, S)$ can be found in polynomial time using dynamic programming. Moreover, the same dynamic program can be used as the separation oracle to solve LP 2.*

Also observer that if one can find a set S such that

$$\sum_{i \in S} \Phi(f_i) p(i, j, S) \geq \lambda \max_{S'} \{ \sum_{i \in S'} \Phi(f_i) p(i, j, S') \},$$

for a constant $\lambda \leq 1$, then the approximation ratio of the algorithm is at least $\lambda(1 - \frac{1}{e})$.

Now we show that the greedy algorithm is essentially $\frac{1}{2}$-competitive in the non-strict model.

Proposition 9 *The non-strict greedy algorithm that upon the arrival of a new query allocates it to a set with the maximum revenue is $\left(\frac{1-\delta}{2-\delta} \right)$-competitive with respect to the optimal non-strict offline solution.*

Proof :   It is easy to see that $\sum_t \pi_{\mathrm{OPT}}^{(I)}(t) \leq \sum_t \pi_G(t)/(1 - \delta)$, where all the variables are defined with respect to the non-strict greedy algorithm. Because the greedy non-strict algorithm can choose the same set that is chosen by the optimal offline algorithm at this step, we have $\pi_{\mathrm{OPT}}^{(A)}(t) \leq \pi_G(t)$. The claim follows immediately by combining these two inequalities. □

6   Conclusion

In this paper we studied the problem of online competitive ad space allocation when the pricing scheme is GSP. We observe a difference in the performance of the allocation algorithm between pay-as-you-bid and the GSP pricing schemes. For example, under the GSP scheme, if the search engine rapidly exhausts the budget of an advertiser with high value, the total revenue may significantly decrease in the future. This is the reason that the algorithm proposed for the pay-as-you-bid model fail to be competitive in the online strict model.

We studied different models of throttling and their effects on the revenue. From algorithmic point of view, the main open problem is to close the gap for the competitive ratio in the strict model. We provided a lower bound of $\frac{1}{3}$, and an upper bound of $\frac{1}{2}$ against the optimal non-strict algorithm. With respect to the optimal strict algorithm, by a reduction from the single slot model, we have an upper bound of $(1 - \frac{1}{e})$ [11].

Another big open question in this area is to combine the algorithmic problem of online ad allocation with a game-theoretic analysis of the auction mechanism. Neither our work nor any of the predecessors of this work [11, 3, 9, 6, 7] consider the strategic behavior of the bidders. This is mainly due to difficulties surrounding repeated auctions and the lack of a satisfactory game theoretic analysis of such auctions. In particular, the folk theorem proves that repeated games have a large set of equilibria, showing that the usual equilibrium concepts do not have good predictive power for such games. The challenge is to come up with a reasonable game theoretic model which limits the range of strategic options of the bidders to avoid running into the folk theorem, while at the same time does not ignore the strategic behavior of the agents altogether. Recently, Feldman et al. [5] made an interesting attempt to get around these difficulties. They proposed a truthful mechanism for the offline setting, assuming the utilities of agents is the number of clicks (not click times value) they receive.

References

[1] Zoë Abrams, Ofer Mendelevitch, and John Tomlin. Optimal delivery of sponsored search advertisements subject to budget constraints. In *Proceedings of the 8th ACM Conference on Electronic Commerce (EC)*, pages 272–278, 2007.

[2] Ian Ayres and Peter Cramton. Deficit reduction through diversity: How affirmative action at the fcc increased auction competition. *Stanford Law Review*, 48:761–815, 1996.

[3] Niv Buchbinder, Kamal Jain, and Seffi Naor. Online primal-dual algorithms for maximizing ad-auctions revenue. In *Proceedings of the 15th Annual European Symposium on Algorithms (ESA)*, 2007.

[4] Benjamin Edelman, Michael Ostrovsky, and Michael Schwarz. Internet advertising and the generalized second price auction: Selling billions of dollars worth of keywords. *American Economic Review*, 97(1):242–259, March 2007.

[5] Jon Feldman, S. Muthukrishnan, Evdokia Nikolova, and Martin Pal. A truthful mechanism for offline ad slot scheduling. *Proceedings of First International Symposium on Algorithmic Game Theory*, 2008.

[6] Gagan Goel and Aranyak Mehta. Adwords auctions with decreasing valuation bids. In Xiaotie Deng and Fan Chung Graham, editors, *WINE*, volume 4858 of *Lecture Notes in Computer Science*, pages 335–340. Springer, 2007.

[7] Gagan Goel and Aranyak Mehta. Online budgeted matching in random input models with applications to adwords. In *Proceedings of Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2008.

[8] Sébastien Lahaie, David M. Pennock, Amin Saberi, and Rakesh Vohra. Sponsored search auctions. In N. Nisan, T. Roughgarden, É. Tardos, and V. V. Vazirani, editors, *Algorithmic Game Theory*, chapter 28. Cambridge University Press, 2007.

[9] Mohammad Mahdian, Hamid Nazerzadeh, and Amin Saberi. Allocating online advertisement space with unreliable estimates. In *Proceedings of the 8th ACM Conference on Electronic Commerce (EC)*, pages 288–294, 2007.

[10] Preston McAfee and John McMillan. Auctions and bidding. *J. ECON. LITERATURE*, pages 699–703, 1987.

[11] Aranyak Mehta, Amin Saberi, Umesh Vazirani, and Vijay Vazirani. Adwords and generalized on-line matching. In *Proceedings of the 46th Annual IEEE Symposium on Foundations of Computer Science*, pages 264–273, 2005.

[12] Hal R. Varian. Position auctions. *International Journal of Industrial Organization*, 26(6):1163–1178, December 2006.