An Adaptive Algorithm for Selecting Profitable Keywords for Search-Based Advertising Services

Paat Rusmevichientong* David P. Williamson†

July 16, 2007

Abstract

Increases in online searches have spurred the growth of search-based advertising services offered by search engines, enabling companies to promote their products to consumers based on search queries. With millions of available keywords whose clickthru rates and profits are highly uncertain, identifying the most profitable set of keywords becomes challenging. We formulate a stylized model of keyword selection in search-based advertising services. Assuming known profits and unknown clickthru rates, we develop an approximate adaptive algorithm that prioritizes keywords based on a *prefix ordering* – sorting of keywords in a descending order of expected-profit-to-cost ratio (or "bang-per-buck"). We show that the average expected profit generated by our algorithm converges to near-optimal profits, with the convergence rate that is independent of the number of keywords and scales gracefully with the problem's parameters. By leveraging the special structure of our problem, our algorithm trades off bias with faster convergence rate, converging very quickly but with only near-optimal profit in the limit. Extensive numerical simulations show that when the number of keywords is large, our algorithm outperforms existing methods, increasing profits by about 20% in as little as 40 periods. We also extend our algorithm to the setting when both the clickthru rates and the expected profits are unknown.

1 Introduction

Search-based advertising services offered by search engines enable companies to promote their products to targeted groups of consumers based on their search queries. Examples of these services include Google Adwords (`http://www.google.com/adwords`), Yahoo! Sponsored Search (`http://searchmarketing.yahoo.com/srch/index.php`), and Microsoft's MSN AdCenter (`http://advertising.msn.com/searchadv/`). A company wishing to advertise through these services typically sets a daily budget, selects a set of keywords, and determines the bid price for each selected keyword. When

*School of Operations Research and Information Engineering, Cornell University, Ithaca, NY 14853, USA. E-mail: paatrus@cornell.edu

†School of Operations Research and Information Engineering and Faculty of Computing and Information Science, Cornell University, Ithaca, NY 14853, USA. E-mail: dpw@cs.cornell.edu

consumers search for one of the selected keywords, search engines then display the ads for the keyword on the search result page based on the bids submitted by the advertisers. A company whose ad is displayed pays the search engine only when the consumer clicks on the ad. The ad will not be displayed if the company's spending that day has exceeded its budget. Figure 1 shows examples of the ads shown to a user under the search-based advertising programs offered by Google and Yahoo!.

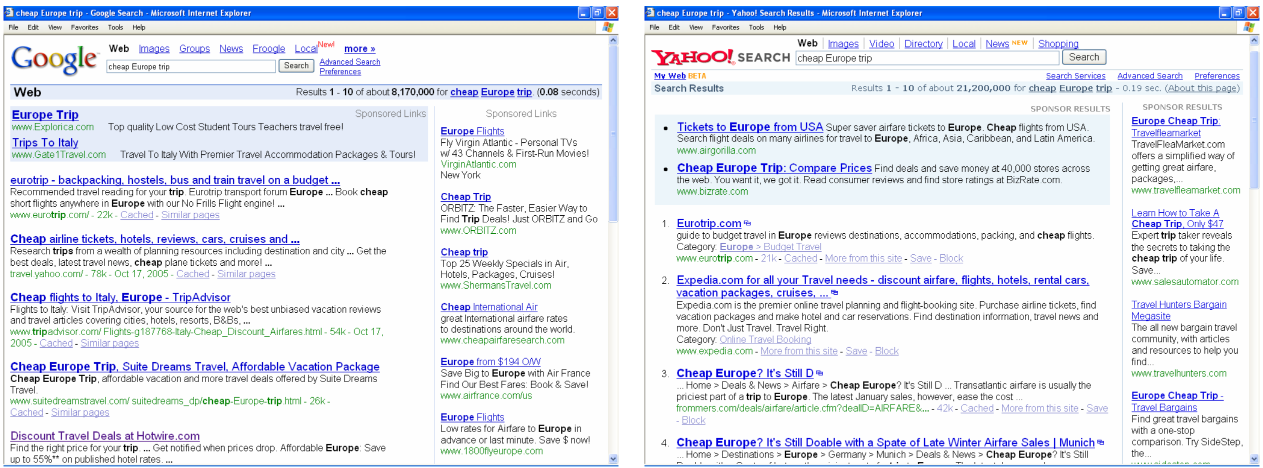


Figure 1: The left and right figures show the result of searching for "cheap Europe trip" (on 10/18/05) on Google and Yahoo!, respectively. The links on the top (shaded) area and on the right hand column of the search result page correspond to ads by companies participating in the search-based advertising programs.

Search engines conduct auctions based on the bids submitted by advertisers, which determine the placements and positions of the ads on the search result page and the cost paid by each advertiser once the consumer clicks on the ad. Details of the actual cost of each keyword, of how a search query matches to a keyword, and of how different bid amounts translate to the position on the search result page vary from one search engine to another. Under the Google Adwords program, for instance, each click will cost the ad's owner an amount equal to the next highest bid plus one cent, and the ad's placement depends not only on the bid amount, but also on the ad's popularity and clickthrus.

With millions of available keywords, a user of search-based advertising services faces challenging decisions. She not only has to identify an effective set of keywords, she also needs to develop ads that will draw consumers and to determine a bid price for each keyword that balances the tradeoffs between the costs and profits that may result from each click on the ad. Furthermore, much of the information that is required to compute these tradeoffs is not known in advance and is difficult to estimate *a priori*. For instance, the clickthru probability associated with the ad for each keyword – the probability that the consumer will click on the ad once it appears on the search result page – varies dramatically depending on the keyword, the ad itself, and the position of the ad on the search result page. In addition, the expected profit associated with each ad is often not known in advance.

We consider a stylized model of search-based advertising services and focus on one of the challenges faced by users of of these programs: given a fixed daily budget and unknown profits and clickthru probabilities, develop a policy that adaptively selects a set of keywords in order to maximize total expected profits. Our model ignores the competitive reactions among advertisers, assuming that the cost-per-click associated with each keyword is fixed and known in advance. We focus on identifying effective sets of keywords in light of uncertain clickthru rates and profitabilities associated with the ads for the keywords. Given a daily budget, we want to spend the money on the most profitable keywords. There is, however, a tradeoff between selecting too few profitable keywords and not spending the entire budget versus selecting too many keywords and depleting the budget too soon, and thus, losing opportunities to receive clicks from more profitable keywords that may arrive later.

Since the total cost depends on the number of clicks, maximizing expected profits given the budget constraint requires knowledge of the clickthru probability and the expected profit of the ad associated with each keyword. Unfortunately, the clickthru rate and the expected profit are generally *not* known in advance and we can obtain estimates of their values *only by* selecting the keyword and observing the displays of the ad (commonly referred to as its impressions), the number of clicks, and the resulting sales. Since we pay for each click on the ad, this process can result in significant costs, yet may offer an opportunity to discover potentially profitable keywords. We thus must balance the tradeoffs between selecting keywords that seem to yield high average profits based on past performance and selecting previously unused keywords in order to learn about their clickthru probabilities and expected profits. This is usually cast in the machine learning literature as balancing "exploitation" (of known good options) and "exploration" (of unknown options that might be better than known options).

We model the problem as follows: we assume that in every time period, some number of queries arrive, where the number is independent and identically distributed. At the beginning of each time period, we must specify a subset of keywords. Then queries arrive sequentially; keyword i is queried with probability λ_i. If we have enough money remaining in our budget, our ad is displayed. With some probability p_i, the user clicks on our ad; we then pay the cost-per-click of c_i to the search engine and receive a random profit whose expected value is denoted by π_i, representing the net income after excluding the cost c_i. We ignore the competitive aspect of the bidding process and assume that the probabilities λ_i and costs c_i, are known; justification of our ability to know these quantities from publicly available sources is given in Section 2.1. The probabilities p_i and the expected profits π_i are *unknown*, and we learn them over time. Our goal is to give an algorithm that converges to near-optimal profits. We consider three different settings in order of increasing complexities.

1) Static Case: When the clickthru rates p_i and expected profits π_i are fixed and known in advance.

2) Unknown Clickthru Rates: When p_i are unknown but π_i are assumed to be known in advance.

3) Unknown Clickthru Rates and Expected Profits: When both p_i and π_i are unknown.

We begin by considering the static case – when the probabilities p_i and the expected profits π_i are known. This problem is related to the well-known stochastic knapsack problem and is NP-complete. We show that we can obtain a near-optimal approximation algorithm by considering *prefix-orderings*. We order the keywords in a decreasing order of expected-profit-to-cost ratio (or "bang-per-buck"); that is, $\pi_1/c_1 \geq \pi_2/c_2 \geq \cdots \geq \pi_N/c_N$. Then our algorithm chooses a set of the form $\{1, \ldots, \ell\}$ for some integer ℓ; we call such a set a *prefix* of the keywords. Since the keywords are sorted in a decreasing order of "bang-per-buck", our algorithm chooses the largest prefix whose expected cost is close to the budget. We show that if the cost of each item is small compared to our budget, if the expected number of arrivals of any given keyword is not too large, and if the distribution of the number of queries is concentrated around its mean, then in expectation our algorithm returns a near-optimal solution, where the closeness to optimality depends on how well these assumptions are satisfied (Theorem 1). In Section 3.1, we give evidence that in practice these assumptions are quite likely to be true.

Our approach based on the prefix-ordering of keywords has been used as an approximation algorithm for the standard knapsack problem by Sahni [27] and in the nonadaptive algorithm of Dean, Goemans, and Vondrak [8] for the stochastic knapsack problem. However, our problem is somewhat different than the stochastic knapsack problem where profits are known, but sizes are drawn from arbitrary known distributions and items can be placed in the knapsack in a specified order. Here costs (corresponding to item sizes) are deterministic and known, but query arrival (corresponding to item placement) is random and not under our control.

Our result in the static case guides the development of an adaptive algorithm for the case when the clickthru probabilities p_i are unknown but the expected profits π_i are assumed to be fixed and known in advance. In each time period, we either choose a random prefix of keywords, or a prefix of keywords using our algorithm from the static case applied to our estimates of the p_i from the last time period. We show in Theorem 2 that, averaged over the number of time periods T, in expectation our algorithm converges to near-optimal profits.

Our proposed adaptive algorithm should be compared to traditional multi-armed bandit algorithms [3, 4, 9, 17, 18, 19]. These algorithms must choose one of N slot machines (or "bandits") to play in each time step. Each play of a bandit will yield a reward, whose distribution is not known in advance. We want to find a sequence of bandits whose average expected reward converges to the optimal reward.

If we associate each bandit with a prefix subset of keywords, we can show that the average expected profit generated from these algorithms converges to the profit of the best prefix subset, which provides a good approximation to the optimal profit. These algorithms focus on finding the best prefix subset and treat the prefix subsets as N independent decisions, ignoring any special structure that exists among the prefix sets. Thus, the convergence rates of these algorithms depend on the number of possible keywords N, which might be very large in our case.

In contrast, by leveraging the special structure of our problem, we focus *not* on finding the best prefix subset, an objective that will require us to try all N prefix subsets, leading to a very slow convergence rate when the number of keywords N is large. Instead, our adaptive algorithm focuses on a special prefix set $\{1, \ldots, \mathcal{I}_U\}$, where $\mathcal{I}_U$ is roughly the largest index such that the expected cost of the prefix $\{1, \ldots, \mathcal{I}_U\}$ does not exceed our budget. As shown in Section 5, we believe that the value of $\mathcal{I}_U$ will in practice be significantly smaller than the total number of keywords N, leading to fast convergence. Although the profit from the prefix set $\{1, \ldots, \mathcal{I}_U\}$ is not the best that can be achieved among all prefix sets, we show that it is quite close to the optimal profit, and thus, close to that of the best prefix set.

By focusing on a specific prefix subset, the convergence rate of our algorithm is independent of the number of keywords N and scales gracefully with the problem's parameters. *Our algorithm thus trades off biases – from selecting only near-optimal prefix sets – with a significantly faster convergence rate.* To our knowledge, this is one of the first online algorithm that allows decision makers to tradeoff between biases of the resulting decisions and improved convergence rates, an important consideration when faced with an extremely large number of decisions. We believe that this represents one of the main contributions of our paper. When compared to two standard multi-armed bandit algorithms, extensive numerical simulations show that in as little as 40 time periods our algorithm outperforms these two algorithms, increasing the profit by about 20%.

We also consider the case when both the clickthru rates p_i and the expected profits π_i are unknown. In this case, we can no longer sort the keywords in a prefix ordering *a priori*. However, it turns out that under certain assumptions, the prefix ordering can be learned very quickly by estimating the average profits and ordering the keywords based on the *average-profit-to-cost* ratio. We show that, with high probability, the ordering based on average profits coincide with the true prefix ordering. Given the ordering of the keywords, we can thus apply our existing adaptive algorithm. In Section 6, we provide an analysis of the proposed method and establish its convergence rate.

To the best of our knowledge, this work is the first publicly available study that addresses the problem of identifying profitable sets of keywords, realistically taking into account the uncertainty of the clickthru probabilities and expected profits and the need to estimate their values based on historical performance. While there has been a great deal of interest from researchers in the area of search-based advertising services, much of it has focused on the design of auctions and payment mechanisms [1, 5, 7, 11, 13, 21, 20, 24]. Our paper is one of only a few that focuses on the *users* of search-based advertising. Kitts et al. [14] consider the problem of finding optimal bidding strategies under a variety of objectives, but only in our static setting in which the clickthru probabilities are known in advance. We believe that removing this assumption is a crucial step towards a usable algorithm. Furthermore, Kitts et al. do not offer an algorithm for finding an optimal set of keywords when the objective is to maximize expected profit subject to a budget constraint, which again is needed given the requirements of using a service such as Google Adwords.

We have attempted to construct a model of the problem that is simultaneously tractable and realistic, justifying our choices with real data whenever possible. Our resulting algorithm is simple to state and code, and provides good results in simulations. Our paper is structured as follows. We present our model in Section 2. We analyze the static case in Section 3. We discuss the connection of our problem with multi-armed bandits in Section 4.1, then give our algorithm in Section 4.2 in the case when the clickthru rates are unknown but the expected profits are fixed and known in advance. We give the results of an experimental study comparing our algorithm with multi-armed bandit algorithms in Section 5. Section 6 extends the algorithm to the case when both the expected profits and the clickthru rates are unknown and we have to learn both quantities simultaneously.

2 Problem Formulation and Model Description

In this section, we develop a stylized model of search-based advertising services and formulate the problem of identifying profitable sets of keywords. Since the details of the auction process for determining ads placements vary by search engine, we will ignore the auction mechanism and any competitive responses by advertisers and assume that the bids for all keywords are constant, and thus, the cost-per-click of each keyword is fixed and known in advance. In addition, we will assume that once the company selects a set of keywords, the ads for the selected keywords will appear on the search result page if and only if there is a search query matching one of the selected keywords and if the company's current balance is sufficient to cover the cost associated with a click on the ad. Sections 2.1 provides additional justification of the model's assumptions and discussion of the publicly available data sources for estimating some of the model's parameters. Assume that we have a probability space $(\Omega, \mathcal{F}, \mathcal{P})$ and N keywords indexed by $1, 2, \ldots, N$. For any $t \geq 1$, let S^t denote the total number of search queries that arrive in period t. We assume that $S^1, S^2, \ldots$ are independent and identically distributed random variables with mean $1 < \mu < \infty$ and the distribution of S^t is known in advance.

We assume that, in each time period, the search queries arrive sequentially. For any $t \geq 1$ and $r \geq 1$, let the random variable Q_r^t denote the keyword associated with the r^{th} search query in period t. We assume that $\left(Q_r^t : t \geq 1, r \geq 1\right)$ are independent and identically distributed random variables and $\mathcal{P}\left\{Q_r^t = i\right\} = \lambda_i$ for $i = 1, \ldots, N$, with $\sum_{i=1}^N \lambda_i \leq 1$. In this case, $1 - \sum_{i=1}^N \lambda_i$ represents the probability that the query corresponds to an unknown keyword. We assume that λ_i's are known in advance (see Section 2.1 for more details).

At the beginning of each time period, we start with a balance of \$$B$ and must select a set of keywords, which will determine the set of ads that will be shown to consumers. As a search query arrives, if the query matches with one of the selected keywords and we have enough money remaining in the account, the ad associated with the keyword appears on the search result page. Each display of an ad is called an *impression*. If the consumer then clicks on the ad, we receive a profit, pay the cost associated with the keyword to the search engine, update our remaining balance, and wait for an arrival of another search query. The process terminates once all search queries have arrived for the time period.

By appropriate scaling, we may assume without loss of generality that the beginning balance in each time period is \$1. For each keyword $1 \leq i \leq N$, let $0 \leq p_i \leq 1$, $c_i > 0$, and $\pi_i \geq 0$ denote the clickthru probability, the ad cost or *cost-per-click* (CPC), and the expected profit per click (after excluding c_i) of keyword i, respectively. We emphasize that since we scale the budget B to \$1, the value c_i represents the *cost-per-click relative to the budget*, that is, the nominal cost associated with clicking on the ad of keyword i divided by the nominal budget. We will assume that the CPC c_i are known in advance and remain constant. Although the CPC of each keyword depends on the bidding behavior of other users, we believe this requirement is a reasonable model in the short-term (see Section 2.1 for more details).

Let $\mathbf{1}(\cdot)$ denote the indicator function and let the random variable $B_r^{A_t}$ indicate the remaining balance when the r^{th} search query appears and we have decided to select the set of keywords $A_t \subseteq \{1, \ldots, N\}$ in period t. Let X_{ri}^t be a Bernoulli random variable with parameter p_i indicating whether or not the consumer clicks on the ad associated with keyword i during the arrival of the r^{th} search query in period t. We assume that, for any i, the random variables $\left(X_{ri}^t : t \geq 1, r \geq 1\right)$ are independent and identically distributed. Also, let Π_{ri}^t denote the profit associated with the click on the ad for keyword i during the arrival of the r^{th} search query in period t. We assume that for any i, the random variables $\left(\Pi_{ri}^t : t \geq 1, r \geq 1\right)$ are independent and identically distributed with $E\left[\Pi_{ri}^t\right] = \pi_i$. We assume that $\left(S^t : t \geq 1\right)$, $\left(Q_r^t : t \geq 1, r \geq 1\right)$, $\left(\Pi_{ri}^t : t \geq 1, r \geq 1, 1 \leq i \leq N\right)$, and $\left(X_{ri}^t : t \geq 1, r \geq 1, 1 \leq i \leq N\right)$ are independent random variables. If we select the set of keywords $A_t \subseteq \{1, \ldots, N\}$ in period t, the expected profit $E\left[Z_{A_t}\right]$ is given by

$$E\left[Z_{A_t}\right] = E\left[\sum_{r=1}^{S^t}\sum_{i=1}^{N} \Pi_{ri}^t \mathbf{1}\left(i \in A_t, Q_r^t = i, B_r^{A_t} \geq c_i, X_{ri}^t = 1\right)\right] . \tag{1}$$

The above expression for the expected profit reflects our stylized model of the search-based advertising dynamics. Under this model, we receive a profit Π_{ri}^t from keyword i during the arrival of the r^{th} search query in period t *if and only if* 1) we select keyword i at the beginning of period t, that is, $i \in A_t$; 2) the r^{th} query corresponds to keyword i, that is, $Q_r^t = i$; 3) we have enough balance to pay for the cost of the keyword, that is, $B_r^{A_t} \geq c_i$; and 4) the consumer actually clicks on the ad, that is, $X_{ri}^t = 1$.

Let $(\mathcal{F}_t : t \geq 1)$ denote a filtration on the probability space $(\Omega, \mathcal{F}, \mathcal{P})$, where $\mathcal{F}_t$ corresponds to events that have happened up to time t. A policy $\phi = (A_1, A_2, \ldots)$ is a sequence of random variables, where A_t corresponds to the set of keywords selected in period t. We will focus exclusively on a class of *non-anticipatory policies* where A_t is $\mathcal{F}_t$-measurable for all $t \geq 1$. For any $T \geq 1$, we aim to find a non-anticipatory policy that maximizes the average expected profit over T periods, that is, $\max_\phi \frac{1}{T}\sum_{t=1}^T E\left[Z_{A_t}\right]$. In general, the optimization problem is intractable and we aim to find an approximation algorithm whose average expected profit converges to a near optimal expected profit.

In addition to the magnitude of the profit relative to the optimal, we will also focus on the rate in which the average expected profit approaches its limit. In particular, we are interested in how this rate scales with the problem's parameters such as the number of keywords N, the cost-per-click c_i, or the expected profit π_i. To formalize this concept, we introduce the following definition. Let $\Theta = \{(N, \mu, (\lambda_i, c_i, p_i, \pi_i : 1 \leq i \leq N)) : N \in \mathcal{Z}_+, \mu > 0, c_i > 0, p_i \in [0,1], \pi_i > 0 \; \forall i\}$ denote the set of all instances of the problem's parameters. For each $\theta \in \Theta$, let $Z^*(\theta)$ denote the maximum expected profit that can be obtained if we know the parameter θ in advance.

Definition 1 *A policy ϕ has a* convergence rate $f : \Theta \times \mathcal{Z} \to \Re_+$ *if for any $\theta \in \Theta$, there exists $0 \leq \zeta(\theta) \leq 1$ such that for any $T \geq 1$,*

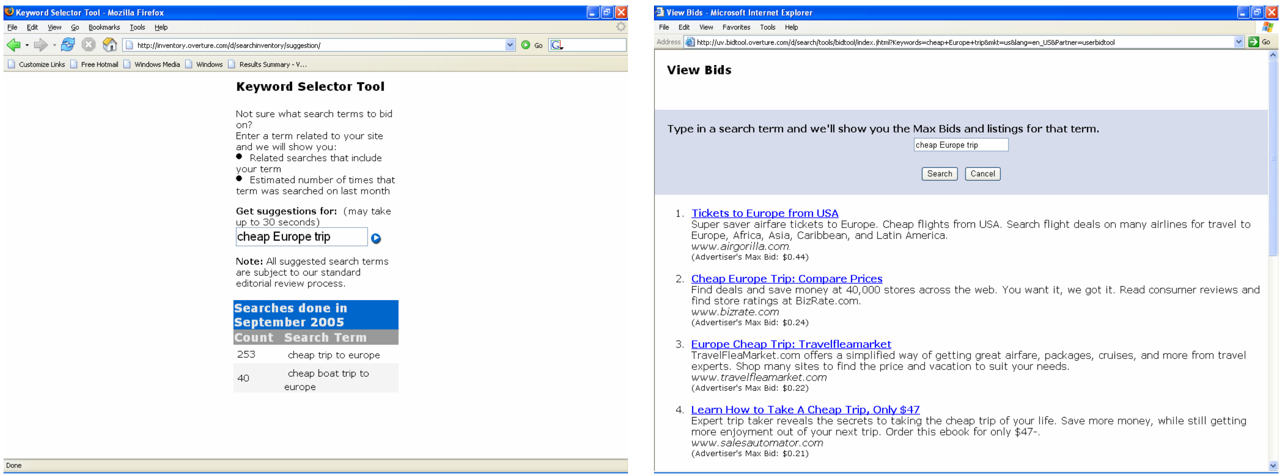
$$\frac{\sum_{t=1}^T E\left[Z_{A_t(\theta)}\right]}{TZ^*(\theta)} \geq \zeta(\theta) - f(\theta, T),$$

where $(A_1(\theta), A_2(\theta), \ldots)$ denotes the sequence of subsets generated by ϕ for the problem instance θ.

Throughout this paper, when we write about the convergence rate of an algorithm and its dependence on a certain parameter, we always refer to how the function f varies as the parameter changes. Of particular interest will be the dependence of the convergence rate on the number of keywords N and the time horizon T, which reflect how well an algorithm scales as the problem size increases. Also, to facilitate our exposition, we will suppress the notation θ when it is obvious from the context.

2.1 Model Calibration

Although we aim to estimate adaptively the clickthru probability and the expected profit associated with the ad of each keyword based on past performance, our formulation assumes that we can estimate *a priori* the arrival process, the query distributions, and the current CPC of each keyword. In this section, we identify publicly available data sources that might be used to estimate these parameters.



(a) (b)

Figure 2: Figure (a) is a screen shot showing the estimated number of search queries in Yahoo! in September 2005 related to the keyword "cheap Europe trip". Figure (b) shows the current bids (as of 10/18/05) for the same keyword under the Yahoo! Sponsored Marketing program.

Figure 2(a) shows the estimated number of search queries in Yahoo! in September 2005 related to the keyword "cheap Europe trip". This data is obtained through the publicly available Keyword Selector tool provided by the Yahoo! Sponsored Search program (`http://inventory.overture.com/d/searchinventory/suggestion/`). Using this information, it might be possible to estimate the probability that each query will correspond to a particular keyword.

Our model also assumes that the cost-per-click (CPC) associated with each keyword is known in advance. Although the actual CPC depends on the outcome of the bidding behaviors of other users, as indicated in Section 2, we believe that this requirement is a reasonable assumption in the short-term. Whereas the clickthru probability and the profit associated with the ad of each keyword can be estimated *only by* selecting the keywords and observing the resulting impressions, clicks, and sales, it is possible to estimate the CPC associated with each keyword using data provided by search engines. For instance, Figure 2(b) shows an example of the current outstanding bids (as of 10/18/05) for the keyword "cheap Europe trip" under the Yahoo! Sponsored Search Marketing program. This information is publicly available through the Bid Tools offered by Yahoo! at `http://uv.bidtool.`

overture.com/d/search/tools/bidtool/index.jhtml.[1] Given the outstanding bids for a keyword, we might approximate the CPC of each keyword as the median bid price, the k^{th} maximum bid price, or the average bid price.

3 A Static Setting: When Clickthru Rates and Expected Profits Are Known

We begin by assuming that the clickthru rates and the expected profits are known. The structure of the policy found here will guide the development of the adaptive policy in the next sections. Since we know the clickthru probability and the expected profit associated with the ad for each keyword, the problem reduces to the following static optimization, which we will call the **STATIC BIDDING** problem.

$$Z^* \equiv \max_{A \subseteq \{1,\ldots,N\}} E\left[Z_A\right] = \max_{A \subseteq \{1,\ldots N\}} E\left[\sum_{r=1}^{S}\sum_{i=1}^{N} \Pi_{ri}\mathbf{1}\left(i \in A, Q_r = i, B_r^A \geq c_i, X_{ri} = 1\right)\right],$$

where $E\left[Z_A\right]$ denotes the expected profit from selecting keywords in A. The above expression of the objective function follows directly from the general expression given in Equation (1) in Section 2; we simply drop the superscript t denoting the time period. The following lemma allows us to simplify the objective function of the **STATIC BIDDING** problem; its proof appears in Appendix A.1.

Lemma 1 *For any $A \subseteq \{1,2,\ldots,N\}$, $r \geq 1$, and keyword $i \in A$, $\mathcal{P}\left\{Q_r = i, B_r^A \geq c_i, X_{ri} = 1 \middle| S \geq r\right\} = \lambda_i p_i \mathcal{P}\left\{B_r^A \geq c_i \middle| S \geq r\right\}$ and $E\left[Z_A\right] = \sum_{i \in A} \pi_i \lambda_i p_i \sum_{r=1}^{\infty} E\left[\mathbf{1}\left(B_r^A \geq c_i, S \geq r\right)\right]$.*

Note that if $p_i = 1$ for all i, then the consumer will always click on the ad. Thus, the problem becomes a variation of the stochastic knapsack problem, which is known to be NP-complete [8, 12]. Unless $P = NP$, it is thus unlikely that there exists a polynomial-time algorithm for solving the **STATIC BIDDING** problem. The following assumption identifies the special structure of our problem that will enable the development of an efficient approximation algorithm in Section 3.2.

Assumption 1

(a) The number of available keywords N is sufficiently large so that $\mu \sum_{i=1}^{N} c_i p_i \lambda_i > 1$.

(b) The keywords are indexed so that

$$\frac{\pi_1}{c_1} \geq \frac{\pi_2}{c_2} \geq \cdots \geq \frac{\pi_{\mathcal{I}^*}}{c_{\mathcal{I}^*}} \geq \max\left\{\frac{\pi_\ell}{c_\ell} : \ell \geq \mathcal{I}^* + 1\right\} \quad \text{and} \quad \mu \sum_{\ell=1}^{\mathcal{I}^*} c_i p_i \lambda_i \leq 1 < \mu \sum_{\ell=1}^{\mathcal{I}^*+1} c_i p_i \lambda_i.$$

(c) There exists $k \geq 1$ and $0 \leq \alpha < 1$ such that $c_i \leq 1/k$ and $\mu \lambda_i \leq k^\alpha$ for $1 \leq i \leq \mathcal{I}^* + 1$.

[1] Since January 2007, Yahoo! has significantly revised its Sponsored Search Marketing program, and the bid information for each keyword is now only available to registered users of the Sponsored Search program.

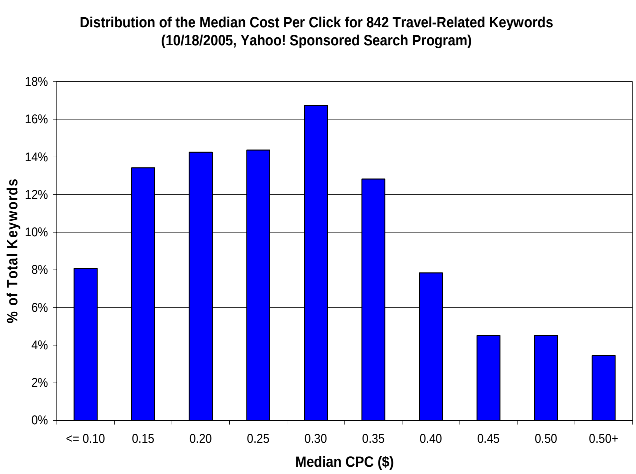
3.1 Discussion of Assumption 1 and the Need for Fast Adaptive Algorithms

Assumption 1(a) requires that the total expected cost from all keywords exceeds our daily budget. We believe that this assumption should always hold in most applications because the number of available keywords N is very large. In most cases, $\mu \sum_{i=1}^{N} c_i p_i \lambda_i$ should be *significantly* larger than 1. Assumption 1(b) requires that the first $\mathcal{I}^*$ keywords are sorted in a *prefix ordering* – in a descending order of expected-profit-to-cost ratio (or "bang-per-buck"), where the index $\mathcal{I}^*$ represents the largest index whose total expected cost $\mu \sum_{i=1}^{\mathcal{I}^*} c_i p_i \lambda_i$ is less than the daily budget of \$1. Since we assume that the expected profit π_i are fixed and known in advance, Assumption 1(b) can be satisfied by sorting *all* keywords. However, we note that the ordering after the first $\mathcal{I}^*$ keywords can be arbitrary.

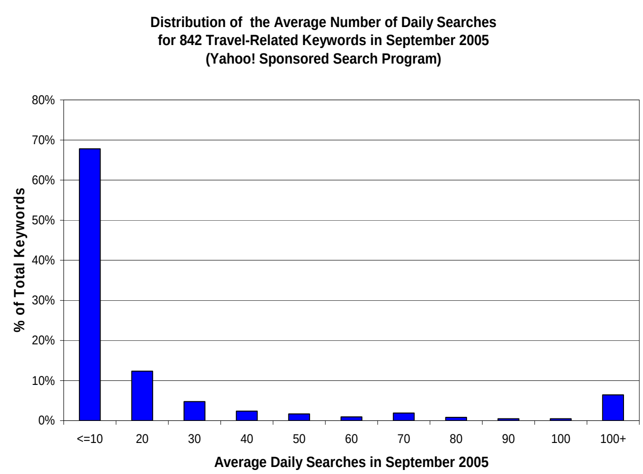
Our key assumption is thus Assumption 1(c), which places constraints on the magnitude of the cost and the expected number of arrivals associated with the first $\mathcal{I}^*$ keywords. In practice, we believe that $\mathcal{I}^*$ should be significantly smaller than the total number of available keywords N, and thus, these constraints apply to only a small fraction of the keywords. The following lemma shows the relationship between the expect number of queries and the CPC of the keywords under Assumption 1(c).

Lemma 2 *Under Assumption 1(c),* $\left(\max_{1 \le i \le \mathcal{I}^*} \mu \lambda_i\right)^{1/\alpha} \le k \le 1 / \max_{1 \le i \le \mathcal{I}^*} c_i$.

For Assumption 1(c) to hold, the expected number of queries among the first $\mathcal{I}^*$ keywords must be less than $1/\max_{1 \le i \le \mathcal{I}^*} c_i$. By definition, c_i represents the cost of keyword i *relative to the daily budget*. Thus, as our budget increases, c_i will become smaller and $1/\max_{1 \le i \le \mathcal{I}^*} c_i$ will increase.



(a)



(b)

Figure 3: Figure (a) shows the distribution of the median bid price under the Yahoo! Sponsored Search program for 842 travel-related keywords collected during a 2-day period from 10/17/05 - 10/18/05. Figure (b) shows the distribution of average daily searches on Yahoo! for the same set of keywords in September 2005.

As shown in Figures 3(a) and 3(b), we believe that in most applications the reciprocal of the CPC relative to the budget should be significantly larger than the the expected number of arrivals for any keyword. Figure 3(a) and 3(b) show the distribution of the median bid price and the average daily searches for each of the 842 travel-related keywords. These keywords were chosen based on the authors' experience in working with an online seller of travel packages. Examples of the keywords include "hawaii vacation package", "africa safari", and "london discount airfare". A complete list of the 842 keywords used in this study is available at `http://legacy.orie.cornell.edu/~paatrus/keywordlist.txt`. The data in Figure 3(a) are collected over a 2-day period (10/17/05-10/18/05) from the publicly available Bid Tool offered by the Yahoo! Sponsored Search program at `http://uv.bidtool.overture.com/d/search/tools/bidtool/index.jhtml`, while the data in Figure 3(b) are obtained through the publicly available Keyword Selection tool offered by the Yahoo! Sponsored Search program at `http://inventory.overture.com/d/searchinventory/suggestion/`.

As seen from Figure 3(a), the average median bid price for a keyword is \$0.30-\$0.35. A daily budget of \$300-\$350/day implies that the reciprocal of the cost-per-click relative to budget is around $1000 = (\$300/\$0.3)$, that is, $k \approx 1000$. From Figure 3(b), for over 80% of the keywords, the average number of daily searches is at most 20 searches per day. With $k = 1000$, this translates to the value of α of around 0.43 ($20 \approx 1000^{0.43}$) Of course, as the daily budget increases, so will the value of k and the value of α will get smaller. As shown in Theorem 1 and 2, the performance of our proposed algorithms will improve as k increases and α decreases.

Although Figure 3(a) and 3(b) suggest that Assumption 1(c) is likely to hold in a short term, advertisers will change their bids over time and update their ads associated with each keyword, and this will change the cost-per-click, the arrival rates, and the clickthru probabilities. Such changes can lead to changes in the performance of our algorithm. Since it is very difficult to incorporate long-term behaviors of advertisers into a model, it is useful to have adaptive algorithms that identify profitable keywords quickly, and we believe this is one of the main contributions of our work.

3.2 An Approximation Algorithm and Discussions

Before we proceed to the main result whose proof is given in Section A.2, let us introduce the following notation which will be used throughout the rest of the paper. Let $\mathcal{I}_U$ be defined by

$$\mathcal{I}_U = \max\left\{\ell : \mu \sum_{i=1}^{\ell} c_i p_i \lambda_i \leq 1 - \frac{1}{k} - \frac{1}{k^{(1-\alpha)/3}}\right\}. \tag{2}$$

Note that by definition $\mathcal{I}_U \leq \mathcal{I}^*$ where $\mathcal{I}^*$ is defined in Assumption 1.

Theorem 1 *Let k and α be defined as in Assumption 1. Suppose the probabilities p_i and the expected profits π_i are known and $1/k + 1/k^{1-\alpha} + 1/k^{(1-\alpha)/3} \leq 1$. Let Z^* denote the optimal profit, $P =$*

$\{1, 2, \ldots, \mathcal{I}_U\}$, *and* $\rho = E[\min\{S, \mu\}]/\mu$. *Then,*

$$\rho\left(1 - \frac{1}{k^{(1+2\alpha)/3}} - \frac{2}{k^{(1-\alpha)/3}}\right) Z^* \leq E\left[Z_P\right] \leq Z^*.$$

Theorem 1 shows that the quality of the prefix-based approximation algorithm depends on two factors. The first factor, $E\left[\min\{S, \mu\}\right]/\mu$, measures how tightly the distribution of S concentrates around its mean. For a Poisson distribution, this parameter can be computed explicitly and it is given in the following lemma whose proof follows from a standard inequality in [10] and we omit the details.

Lemma 3 *If S is a Poisson random variable with mean μ, then* $1 - \frac{1}{2\sqrt{\mu}} \leq E\left[\min\{S, \mu\}\right]/\mu \leq 1$.

In most applications, the expected number of search queries μ will be very large, ranging from hundreds of thousands to over millions of queries per day. If the arrival follows a Poisson distribution, then the ratio $E\left[\min\{S, \mu\}\right]/\mu$ will be very close to one.

The second factor that influences the quality of our approximation algorithm is the value of k and α. Since $0 \leq \alpha < 1$, as k – the relative value between the budget and the CPC for each keyword – increases, the expected profit will be close to the optimal. Figure 4 shows the value of the performance bound given in Theorem 1 (with $\rho = 1.0$) for various values of α and k, with k ranging from 10^3 to 10^6. We observe that the bound improves as k increase and as α decreases. We believe that in most applications the our prefix-based approximation should perform well. As shown in Figure 3(a) and 3(b), the values of k is generally quite large and α fairly small. With $k = 1000$ and $\alpha = 0.2$, we observe from Figure 4 that our profit is within 68% of the optimal.

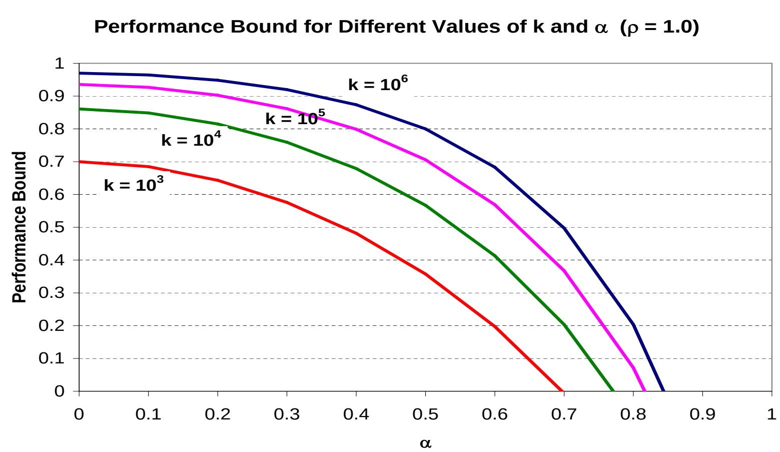


Figure 4: Plot of the performance bound given in Theorem 1 for various values of k and α (with $\rho = 1.0$).

From Assumption 1(a), it is interesting to note that there exists at least one keyword i such that $1/N < \mu c_i \lambda_i p_i$. If this keyword corresponds to one of the first $\mathcal{I}^*$ keywords with the highest expected-profit-to-cost ratio, it follows from Assumption 1(c) that $1/N \leq p_i/k^{1-\alpha} \leq 1/k^{1-\alpha}$, and therefore,

$k^{1-\alpha} \leq N$. In this case, we have the following upper bound on the maximum value of the performance guarantee: $1 - \frac{1}{k^{(1+2\alpha)/3}} - \frac{2}{k^{(1-\alpha)/3}} \leq 1 - \frac{2}{k^{(1-\alpha)/3}} \leq 1 - \frac{2}{N^{1/3}}$. As we are interested in the setting when the number of keywords N is very large (with hundreds of thousands of keyword), the above upper bound should impose negligible losses in performance.

4 Unknown Clickthru Probabilities with Known Expected Profits

In this section, we consider the case when the clickthru probabilities are *not* known in advance, but the expected profits π_i are fixed and assumed to be known. To estimate the clickthru probabilities, we need to select the keywords and observe the resulting impressions and clicks, a process that can potentially result in significant costs. In Section 4.1, we will show how this problem can be formulated as a multi-armed bandit problem, where each bandit corresponds to a prefix subset of keywords. By formulating the problem as an instance of the multi-armed bandit, we can apply existing adaptive algorithms developed for this class of problems. Unfortunately, existing multi-armed bandit algorithms have performance guarantees that deteriorate as the number of keywords increases, motivating us to develop an alternative adaptive approximation algorithm that leverages the special structure of our problem. In Section 4.2, we describe our algorithm and show that the algorithm exhibits a better performance bound compared with traditional multi-armed bandit algorithms. We will show that the quality of the solution generated by our algorithm is independent of the number of keywords and scales gracefully with the problem's other parameters.

4.1 A Traditional Multi-Armed Bandit Approach

Theorem 1 shows that by considering only subsets of keywords that follow the prefix ordering – a descending order of expected-profit-to-cost ratio – we can achieve near-optimal profits. This result enables us to reduce the size of the decision space from 2^N (all possible subsets of N keywords) to just N. Assume that the keywords are indexed as in Assumption 1. For any $1 \leq i \leq N$, let $D_i = \{1, 2, \ldots i\}$. We can view each D_i as a decision whose payoff corresponds to the expected profit $E\left[Z_{D_i}\right]$ that results from selecting the set of keywords D_i. Let Z^{prefix} denote the maximum profit among decisions $D_1, \ldots D_N$, that is, $Z^{prefix} = \max_{1 \leq i \leq N} E\left[Z_{D_i}\right]$.

Viewing each D_i as a bandit, we can then formulate our problem as the multi-armed bandit problem for which many algorithms exist. Two such algorithms are the UCB1 and the ε_n-Greedy algorithms whose brief descriptions are given below (see [3] for more details). Both algorithms maintain the average profit obtained from each of the N bandits, and for any i, we denote by $\bar{x}_i$ and n_i the current average profit from selecting the decision D_i and the number of times that D_i has been selected. Both of these parameters are initialized to zero. Also, we denote by L the maximum profit that can be obtained in any period and for any $1 \leq i \leq N$, let $\Delta_i = Z^{prefix} - E\left[Z_{D_i}\right]$.

Algorithms	Deterministic Policy UCB1	Randomized Policy ε_n-Greedy
Initialization	Try each of the N prefix sets once, update the average profits $\bar{x}_i$ and set $n_i = 1$ for all i.	Choose $c > 0$ and $0 < d < \min\{\Delta_i : \Delta_i > 0\}$, and define a sequence $\{\epsilon_t \in (0,1] : t \geq 1\}$ where $\varepsilon_t \equiv \min\left\{1,\ cNL^2/\left(d^2 t\right)\right\}$.
Description	For each period $t \geq 1$, choose the subset that maximizes $\bar{x}_j + L\sqrt{2\left(\ln t\right)/n_j}$.	For each period $t \geq 1$, with probability $1 - \varepsilon_t$, choose the prefix set that has the highest current average profit and, with probability ε_t, choose a random prefix set.

Table 1: Description of the UCB1 and ε_n-Greedy algorithms.

As seen from Table 1, UCB1 is a deterministic adaptive algorithm that requires the decision-maker to try each of the N prefix subsets at least once. Although the ε_n-Greedy algorithm does not try every subset, we need to have prior knowledge of d which represents that difference between the best and the second best prefix sets. Let $(Z_t(\text{UCB1}) : t \geq 1)$ and $(Z_t(\text{Greedy}) : t \geq 1)$ denote the sequence of profits obtained by the algorithms UCB1 and ε_n-Greedy, respectively. The average expected profit generated by these algorithms are given by (see [3]):

$$1 \geq \frac{\sum_{t=1}^{T} E\left[Z_t(\text{UCB1})\right]}{TZ^{prefix}} \geq 1 - \frac{L\left(\beta_N + \alpha_N \log T\right)}{TZ^{prefix}},$$

where $\alpha_N = \sum_{1 \leq i \leq N : \Delta_i > 0} 8/\Delta_i$ and $\beta_N = \left(1 + \pi^2/3\right)\sum_{i=1}^{N} \Delta_i$, and if $c > 5d^2/L^2$,

$$1 \geq \frac{\sum_{t=1}^{T} E\left[Z_t(\text{Greedy})\right]}{TZ^{prefix}} \geq 1 - \frac{\left(cL^2/d^2\right)\left(dN + L\log T + (cN)^{cL^2/5d^2} o\left(\log T\right)\right)}{TZ^{prefix}}.$$

The above results show that the average profit earned by the UCB1 and ε_n-Greedy algorithms converge to the maximum expected profit Z^{prefix} among prefix sets. We know from Theorem 1 that Z^{prefix} provides a good approximation to Z^*, suggesting that both algorithms may also provide viable algorithms for solving our problem. This result is stated in the following proposition whose proof follows immediately from the relationship between Z^* and Z^{prefix} implied by Theorem 1.

Proposition 1 *Let* $\eta = \rho\left(1 - 1/k^{(1+2\alpha)/3} - 2/k^{(1-\alpha)/3}\right)$ *. Then, for any* $T \geq 1$,

$$\frac{\sum_{t=1}^{T} E\left[Z_t(\text{UCB1})\right]}{TZ^*} \geq \eta - \frac{L\left(\beta_N + \alpha_N \log T\right)}{TZ^*}$$

*and if the parameters c and d in the ε_n-*Greedy *algorithm are chosen so that* $c > 5d^2/L^2$,

$$\frac{\sum_{t=1}^{T} E\left[Z_t(\text{Greedy})\right]}{TZ^*} \geq \eta - \frac{\left(cL^2/d^2\right)\left(dN + L\log T + (cN)^{cL^2/5d^2} o\left(\log T\right)\right)}{TZ^*}$$

4.1.1 Discussion of the Performance Bounds for UCB1 and ε_n-Greedy Algorithms

The discussion here will help us contrast the bound for our proposed algorithm in Section 4.2.

Choice of Objectives: The result of Proposition 1 is stated in terms of the maximum profit Z^* among *all* subsets of keywords, instead of Z^{prefix}. We believe that Z^* represents a reasonable benchmark because our goal is to determine a policy that achieve maximum profits. Of course, as shown in Theorem 1, in most applications Z^{prefix} and Z^* should be close together.

Dependence on the Maximum Profit L: The running average profit of the UCB1 algorithm depends linearly on the maximum profit L, while the ε_n-GREEDY algorithm depends on L^3. As shown in the following lemma, when the profit Π_i from each keyword i is deterministic with $\Pi_i = \pi_i$ for all i, we can upper bound L in terms of the ratio of the expected profit and the cost of the first keyword. The proof of this result follows immediately from the definition of our problem.

Lemma 4 *Under Assumption 1, if the profit from each keyword i is deterministic and equal to π_i, then* $L \leq \max\left\{\sum_{i=1}^{N} \pi_i x_i \;\middle|\; \sum_{i=1}^{N} c_i x_i \leq 1, \quad x_i \in \mathcal{N} \text{ for all } i\right\} \leq \pi_1/c_1$.

Recall that c_1 represents the cost per click of keyword 1 *relative* to the daily budget. As the budget increases, so will the value of L. Thus, L depends primarily on the daily budget and should be *independent* of the number of keywords.

Dependence on the Number of Keywords N: Since both the UCB1 and ε_n-GREEDY algorithms treat each of the N prefixes $D_1, \ldots, D_N$ are N independent bandits and ignore any special structure among these sets, the rates of convergence scale linearly with the number of keywords N. When the number of keywords is extremely large, the dependence on N represents an undesirable feature. In fact, the $UCB1$ algorithm requires us to try each of the N possible decisions once during the first N time periods. Clearly, for large values of N, this might not be feasible. Although the ε_n-GREEDY algorithm does not require testing of each decision once, it requires a prior knowledge of the difference between the best and the second best decision (see the definition of the parameter d), which can be difficult to estimate when the number of decisions is large.

4.2 An Improved Adaptive Approximation Algorithm

By exploiting the special structure of our problem, we will develop an alternative algorithm whose convergence rate is *independent of the number of keywords*. Our algorithm, which we will refer to as the ADAPTIVE BIDDING algorithm, is shown in Figure 5. In the algorithm, $\hat{p}_i^t$ represents our *estimate*, at the end of t periods, of the clickthru probability of the ad associated with keyword i. As long as the ad associated with keyword i has not received any impression (that is, when we have no data on the clickthru probability), we will set $\hat{p}_i^t$ to 1. However, when the ad receives at least one impression, we set $\hat{p}_i^t$ to the average clickthru probability. The main result of this section is stated in the following theorem whose proof appears in Appendix B. Recall that $\rho = E\left[\min\left\{S, \mu\right\}\right]/\mu$ and L denote the maximum profit that can be obtained in any one period. Also, let $\lambda^* = \max_{1 \le i \le \mathcal{I}_U} \lambda_i$. Note that $\lambda^* \mu$ corresponds to the maximum expected number of queries among the first $\mathcal{I}_U$ keywords.

ADAPTIVE BIDDING

- INITIALIZATION: For any i, let y_i and x_i denote the *cumulative* number of impressions and clicks, respectively, that the ad associated with keyword i has received. Initialize $y_i = x_i = 0$ and $\hat{p}_i^0 = 1$ for all i. Choose a sequence $(\gamma_t \in [0,1] : t \ge 1)$, where γ_t denotes the probability that we choose a random decision at time t. (Section 5.3 will explore good choices for the γ_t).

- DESCRIPTION: For each period $t \ge 1$,

  1) Let ℓ_t be the index such that $\mu \sum_{u=1}^{\ell_t} c_u \lambda_u \hat{p}_u^{t-1} \le 1 - 1/k - 2/k^{(1-\alpha)/3} < \mu \sum_{u=1}^{\ell_t + 1} c_u \lambda_u \hat{p}_u^{t-1}$

  2) Let F_t be an independent binary random variable such that $\mathcal{P}\left\{F_t = 1\right\} = 1 - \gamma_t$ and $\mathcal{P}\left\{F_t = 0\right\} = \gamma_t$, and define g_t as follows: if $F_t = 1$, set $g_t = \ell_t$, and when $F_t = 0$, set g_t to an integer chosen uniformly at random from $\{1, \ldots, N\}$.

  3) Select the set of keywords $G_t = \{1, \ldots, g_t\}$ and observe the resulting impressions and clicks.

  4) Updates: For any keyword i, let V_i^t and W_i^t denote the number of impressions and clicks that keyword i receives in this period, respectively. Then, for all i,

$$y_i \leftarrow y_i + V_i^t, \qquad x_i \leftarrow x_i + W_i^t, \quad \text{and} \quad \hat{p}_i^t = \begin{cases} 1, & \text{if } y_i = 0 \\ \frac{x_i}{y_i}, & \text{if } y_i > 0 \end{cases}$$

- OUTPUT: A sequence $(G_t : t \ge 1)$ and $(g_t : t \ge 1)$.

Figure 5: ADAPTIVE BIDDING algorithm.

Theorem 2 *Let $(G_t : t \ge 1)$ denote the sequence of decisions generated by the* ADAPTIVE BIDDING *algorithm. Under Assumption 1, if $1/k + 1/k^{1-\alpha} + 2/k^{(1-\alpha)/3} \le 1$, then for any $T \ge 1$ and $0 < \epsilon < 1$,*

$$\frac{\sum_{t=1}^{T} E\left[Z_{G_t}\right]}{T Z^*} \ge \rho \left(1 - \frac{1}{k^{(1+2\alpha)/3}} - \frac{3}{k^{(1-\alpha)/3}}\right) - \frac{\sum_{t=1}^{T} \gamma_t}{T} - M(\epsilon, T),$$

where

$$M(\epsilon, T) = \frac{\epsilon \rho \mathcal{I}_U}{k^{1-\alpha}} + \frac{(128/\rho^2)\ L}{\left(\min_{1 \le i \le \mathcal{I}_U} \pi_i \lambda_i p_i\right)\ kT} + \frac{12\ \mathcal{I}_U \lambda^* \mu / \left(1 - e^{-\lambda^* \mu}\right)}{\epsilon^2 k T}.$$

The expression $M(\epsilon, T)$ reflects the loss in performance due to estimation errors (from using $\hat{p}_i^t$ in place of p_i). The following corollary provides a simplified expression for $M(\epsilon, T)$ under the additional assumption that the maximum expected number of search queries among the keywords $1, 2, \ldots, \mathcal{I}_U$ is at least one. As shown in Figure 3(b), we believe that this is a reasonable assumption in many applications. The proof of Corollary 1 follows immediately from Assumption 1 that $\lambda^* \mu \leq k^\alpha$ and from the fact that $1/\left(1 - e^{-\lambda^* \mu}\right) \leq 1/(1 - e^{-1}) \leq 2$.

Corollary 1 *Under the hypothesis of Theorem 2, if the maximum expected number of search queries among the keywords* $1, 2, \ldots, \mathcal{I}_U$ *is at least one, that is,* $\max_{1 \leq i \leq \mathcal{I}_U} \lambda_i \mu \geq 1$, *then*

$$M(\epsilon, T) \leq \frac{(128/\rho^2)\ L}{(\min_{1 \leq i \leq \mathcal{I}_U} \pi_i \lambda_i p_i)\ kT} + \frac{\mathcal{I}_U}{k^{1-\alpha}} \left(\rho\epsilon + \frac{24}{\epsilon^2 T}\right),$$

and by choosing $\epsilon = 1/\sqrt[3]{T}$, *we obtain*

$$M(\epsilon, T) \quad \leq \quad \frac{(128/\rho^2)\ L}{(\min_{1 \leq i \leq \mathcal{I}_U} \pi_i \lambda_i p_i)\ kT} + \frac{25\mathcal{I}_U}{k^{1-\alpha} \sqrt[3]{T}}$$

4.3 Discussion of Theorem 2 and Corollary 1

We will now discuss and contrast different aspects of the performance bound of our Adaptive Bidding with those of the UCB1 and the ε_n-Greedy algorithms.

4.3.1 Approximation Guarantee

From Proposition 1, we have following performance bounds for the UCB1 and the ε_n-Greedy algorithms: $\eta = \rho\left(1 - 1/k^{(1+2\alpha)/3} - 2/k^{(1-\alpha)/3}\right)$,

$$\frac{\sum_{t=1}^T E\left[Z_t(\text{UCB1})\right]}{TZ^*} \geq \eta - O\left(\frac{LN \log T}{T}\right) \quad \text{and} \quad \frac{\sum_{t=1}^T E\left[Z_t(\text{Greedy})\right]}{TZ^*} \geq \eta - O\left(\frac{L^2 N \log T}{T}\right).$$

We observe that according to Theorem 2, there is a slight decrease in the approximation guarantee of our algorithm, from $2/k^{(1-\alpha)/3}$ to $3/k^{(1-\alpha)/3}$. In most applications, we believe that the value of k is so large that the resulting loss in performance guarantee is negligible.

4.3.2 Dependence on L

The convergence rate of our Adaptive Bidding algorithm depends on the maximum profit L that can be earned in any period, and on $\min_{1 \leq i \leq \mathcal{I}_U} \pi_i \lambda_i p_i$. The expression $\pi_i \lambda_i p_i$ represents the expected profit *per search query* associated with keyword i, which we believe to be independent of the number of keywords N. Thus, the convergence rate of our Adaptive Bidding algorithm should exhibit a similar dependence on L as in both the the UCB1 and ε_n-Greedy algorithms. As shown in the next section, however, our algorithm differs from traditional multi-armed bandit algorithms in our tradeoffs between the time horizon T and the number of keywords N.

4.3.3 Tradeoffs Between Time Horizon T and the Number of Keywords N

As the number of time horizon T increases, the average running expected profits after T periods under the UCB1 and the ε_n-Greedy algorithms converge at the rate of $O(\log T/T)$. However, the performance bounds of the the UCB1 and the ε_n-Greedy algorithms scale linearly with the number of keywords N, which can be up to hundreds of thousands of keywords in a typical marketing campaign. In fact, we believe that in most applications, the number of keywords N is significantly larger (by many orders of magnitude) than the number of time horizon T available for the managers to try different decisions. When the length of the horizon T is small, it might be more desirable to have algorithms whose convergence rate does not scale linearly with the total number of keywords N.

By exploiting the special structure of our problem, our Adaptive Bidding algorithm offers an alternative to traditional multi-armed bandit algorithms. Although the average profit after T periods under our proposed algorithm converges to the optimal Z^* at the rate of $O\left(1/\sqrt[3]{T}\right)$ (see Corollary 1), the convergence rate of algorithm depends linearly only on $\mathcal{I}_U$, independent of the number of keywords N. The following lemma provides upper and lower bounds on $\mathcal{I}_U$.

Lemma 5 *Under Assumption 1, if* $1/k + 1/k^{1-\alpha} + 2/k^{(1-\alpha)/3} \leq 1$*, then* $k^{2(1-\alpha)/3} < \mathcal{I}_U \leq \mathcal{I}^*$*.*

Proof: By definition, $\mathcal{I}_U \leq \mathcal{I}^*$. We will establish a lower bound on $\mathcal{I}_U$ by contradiction. If $\mathcal{I}_U \leq k^{2(1-\alpha)/3}$, it follows from Assumption 1(c) that $\mu c_i \lambda_i p_i \leq 1/k^{1-\alpha}$ for $i \leq \mathcal{I}^* + 1$, which implies that $\mu \sum_{i=1}^{\mathcal{I}_U+1} c_i \lambda_i p_i \leq \frac{k^{2(1-\alpha)/3}+1}{k^{1-\alpha}} = \frac{1}{k^{(1-\alpha)/3}} + \frac{1}{k^{1-\alpha}} \leq 1 - \frac{1}{k} - \frac{1}{k^{(1-\alpha)/3}}$ where the last inequality follows from our assumption that $1/k + 1/k^{1-\alpha} + 2/k^{(1-\alpha)/3} \leq 1$. This contradicts the definition of $\mathcal{I}_U$. ■

The above lemma shows that the number of keywords $\mathcal{I}_U$ in our prefix-based approximation algorithm is less than $\mathcal{I}^*$, the maximum number of keywords whose total expected cost does not exceed our daily budget. The actual value of $\mathcal{I}^*$ will depend on the cost-per-click, the expected number of queries, and the clickthru rate for each keyword, along with the daily budget. In the case when these parameters are the same, we can establish an upper bound on $\mathcal{I}^*$ as shown in the following lemma whose proof follows directly from the definition.

Lemma 6 *Under Assumption 1, if* $c_i = 1/k$*,* $\lambda_i \mu \geq 1$*, and* $p_i = p$ *for all* i*, then* $\mathcal{I}^* \leq k/p$*.*

When the costs and the clickthru rates of the keywords are roughly the same, the parameter $\mathcal{I}_U$ will depend primarily on k which represents *the size of the budget relative to the cost-per-click of each keyword*, and is independent of the number of keywords N. We believe that the above result is representative of the general phenomenon underlying our problem. The parameter $\mathcal{I}_U$ reflects the maximum number of keywords whose expected cost does not exceed $1 - 1/k - 1/k^{(1-\alpha)/3}$ fraction of the budget; and thus, its magnitude should be determined primary by the budget constraint, and in most cases, should be significantly smaller than N.

4.3.4 Tradeoffs Between Time Horizon T and the Approximation Guarantee

By choosing different values of ϵ in Theorem 2, we can improve the dependence on T in the convergence rate of ADAPTIVE BIDDING algorithm, at the expense of a reduction in the approximation guarantee. This result is stated in the following lemma. We state the result under the assumption that $\max_{1 \le i \le \mathcal{I}_U} \lambda_i \mu \ge 1$, but a similar statement can be obtained in the general case. Note that by Lemma 5, $k^{2(1-\alpha)/3}/\mathcal{I}_U < 1$.

Lemma 7 *Under the hypothesis of Theorem 2, if the maximum expected number of search queries among the keywords* $1, 2, \ldots, \mathcal{I}_U$ *is at least one, that is,* $\max_{1 \le i \le \mathcal{I}_U} \lambda_i \mu \ge 1$*, then by choosing* $\epsilon = k^{2(1-\alpha)/3}/\mathcal{I}_U$*, we obtain*

$$M(\epsilon, T) = \frac{\rho}{k^{(1-\alpha)/3}} + \frac{(128/\rho^2)\ L}{(\min_{1 \le i \le \mathcal{I}_U} \pi_i \lambda_i p_i)\ kT} + \frac{24\mathcal{I}_U^3}{k^{7(1-\alpha)/3} T}, \quad \text{and}$$

$$\frac{\sum_{t=1}^{T} E\left[Z_{G_t}\right]}{T Z^*} \ge \rho \left(1 - \frac{1}{k^{(1+2\alpha)/3}} - \frac{4}{k^{(1-\alpha)/3}}\right) - \frac{\sum_{t=1}^{T} \gamma_t}{T} - \frac{(128/\rho^2)\ L}{(\min_{1 \le i \le \mathcal{I}_U} \pi_i \lambda_i p_i)\ kT} - \frac{24\mathcal{I}_U^3}{k^{7(1-\alpha)/3} T}.$$

As seen from the above lemma, the approximation guarantee in this case decreases from $3/k^{(1-\alpha)/3}$ to $4/k^{(1-\alpha)/3}$, however, the convergence rate in terms of T improve from $O\left(1/\sqrt[3]{T}\right)$ to $O\left(1/T\right)$, which makes it comparable to the convergence rate under both the UCB1 and the ε_n-GREEDY algorithms. Of course, we pay the price by having our convergence rate scale with $\mathcal{I}_U^3$ instead of $\mathcal{I}_U$. Whether this leads to a faster convergence rates will depend on how $\mathcal{I}_U^3$ compare with the number of time periods T that the manager has available for making decisions.

4.3.5 Tradeoffs Between Biases and Convergence Rates

As one of the main contributions, we emphasize that our proposed algorithm allows the decision maker to trade off biases with faster convergence rates. The limiting profit under our ADAPTIVE BIDDING algorithm is *less than* the maximum expected profit among all prefix sets, which is achieved by traditional multi-armed bandit algorithms. The convergence rate of our algorithm, however, does *not* increase with the number of keywords N as in other methods. Our algorithm is thus suited to a setting when the number of keywords N is significantly larger than the time horizon T and when the decision-maker is willing to sacrifice small biases in performance for faster convergence. To our knowledge, this is the first algorithm that allow trade-offs between biases of the limiting decisions and convergence rates.

5 Experiments

We compare our proposed Adaptive Bidding algorithm with both the UCB1 and the ε_n-Greedy algorithms. In Section 5.1, we compare the performance of all three algorithms on a representative large-scale problem involving 50,000 keywords. To better understand the tradeoffs between biases and convergence rates, we consider in Section 5.2 a setting with a much smaller number of keywords and focus on understanding the impact on the performance as we increase the number of keywords. In Section 5.3 considers the impact of initialization and randomization.

5.1 Performance on a Representative Large-Scaled Problem

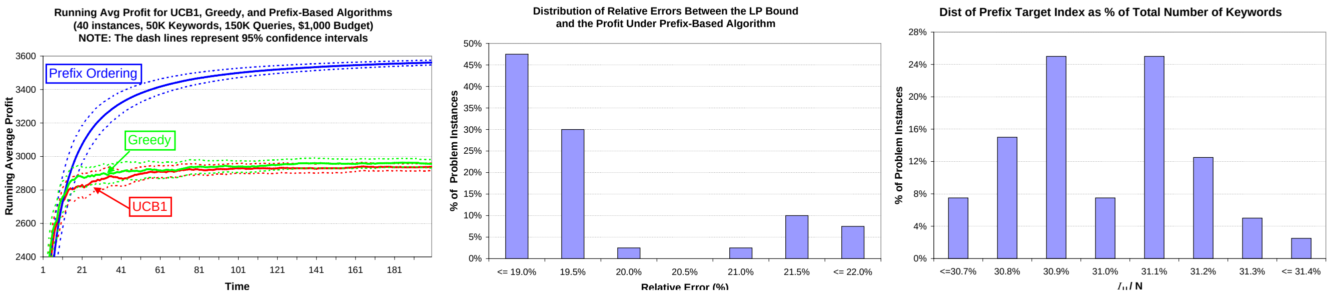
Our experiment consists of 40 randomly generated problem instances. Each problem instance has 50,000 keywords, 200 time periods, $1,000 budget per period, and the number of search queries in each time period has a Poisson distribution with a mean of 150,000. For each problem instance, the cost-per-click of each keyword is chosen uniformly at random from the interval [0.1,0.3) and the profit-per-click of each keyword is chosen uniformly at random from the interval [0,1). The clickthru rate associated with each keyword is a random number from [0.0,0.2). To generate the probability that a search query will correspond to each keyword, we assign random numbers (chosen independently and uniformly from [0,1)) to all keywords and normalize them.

For each problem instance, we compare our Adaptive Bidding with the UCB1[2] and the ε_n-Greedy algorithms. For our prefix-based algorithm, we set the randomization probability γ_t to be $\gamma_t = 1/t^2$ for all $t \geq 1$. The parameter d for the ε_n-Greedy algorithm is set to one (see Table 1 for more details). As noted in [3], the performance of the greedy algorithm depends crucially on the parameter c (see Table 1 for more details). Through a simple line search, we choose the parameter c that works best for our problem and we set $c = 0.5/L^2$, where L denotes the largest profit in any one period.

Figure 6(a) shows the running average profit over time for all three algorithms when averaged over all problem instances. The dashed lines above and below the solid lines correspond to the 95% confidence interval. As seen from the figure, our Adaptive Bidding outperforms both UCB1 and ε_n-Greedy algorithms, increasing the profit by about 20% in as little as 40 periods.

Figure 6(b) shows the distribution over all problem instances of the relative difference between the linear programming upper bound (Lemma 8) and the profit under our algorithm. This is an upper bound on the difference from optimality. As seen from the figure, the average profit differs from the optimal by at most 22%. For this experiment, the value of k and α are around 3,333 and 0.22,

[2] The UCB1 initializes by trying each of the prefix sets once. We choose these sets by random sampling from among those that have not been tried.



(a) (b) (c)

Figure 6: (a) compares the running average profit over time under the UCB1, ε_n-Greedy, and ADAPTIVE BIDDING algorithms. (b) shows the distribution of the relative difference between the linear programming bound and the average profit generated by our algorithm. (c) shows the distribution of $\mathcal{I}_U/N$.

respectively. For these values, the bound of Theorem 2 implies that our algorithm should converge to at least 61% of the optimal, but our simulation results show that our algorithm performs much better than this.

$\mathcal{I}_U$ versus N: Figure 6(c) shows the distribution of $\mathcal{I}_U/N$. Recall that $\mathcal{I}_U$ represents the target index under our ADAPTIVE BIDDING algorithm. The average value of $\mathcal{I}_U$ across problem instances is around 15,500, or about 31% of the total number of keywords N. The small value of $\mathcal{I}_U$ contributes to the fast convergence of our ADAPTIVE BIDDING algorithm.

5.2 Tradeoffs Between Biases and Convergence Rates

To help us understand the tradeoffs between biases and convergence rates, we compare the three algorithms on problem instances with a smaller number of keywords. Figure 7 shows the performance of the three algorithms as the number of keywords increases from 750 to 850 keywords. For each value of N, we consider 40 problem instances with each instance having \$200 budget per period and the expected number of searches per period is $4,000$. To facilitate our comparison, we set the CPC and the clickthru probability of each keyword to \$0.10 and 1.0, respectively.

The figures on the left-hand column correspond to the running average profit over time, while the figures on the right-hand column correspond to the average profit in each period. Since our prefix-based algorithm focuses on a specific prefix index $\mathcal{I}_U$, the running average profit converges very quickly, yielding significantly higher profits during the first 500 periods. In fact, the average profit in each period under our algorithm is virtually constant during the entire horizon, indicating that our algorithm quickly identifies the target prefix index $\mathcal{I}_U$.

The limiting decision under our algorithm is, of course, biased and it is not the best possible prefix set. After exploring most of the prefix sets, the running average profit under both the UCB1 and ε_n-GREEDY algorithms eventually catch up with and surpass that of our algorithm. It is interesting that the time when the running average profits of both algorithms exceed our profit is roughly the same as the number of keywords. We note that if each period in our experiment corresponds to a day, then it would take both the UCB1 and ε_n-Greedy algorithms over one year to catch up with the running average of our algorithm. In our application, it is thus equally important to focus on the rate of convergence, as well as the quality of the limiting solution. The results from Figure 7 provides additional numerical evidence supporting our claim that our proposed Adaptive Bidding algorithm trades off faster convergence rate with biases in the limiting decision, representing one of the main contributions of our paper.

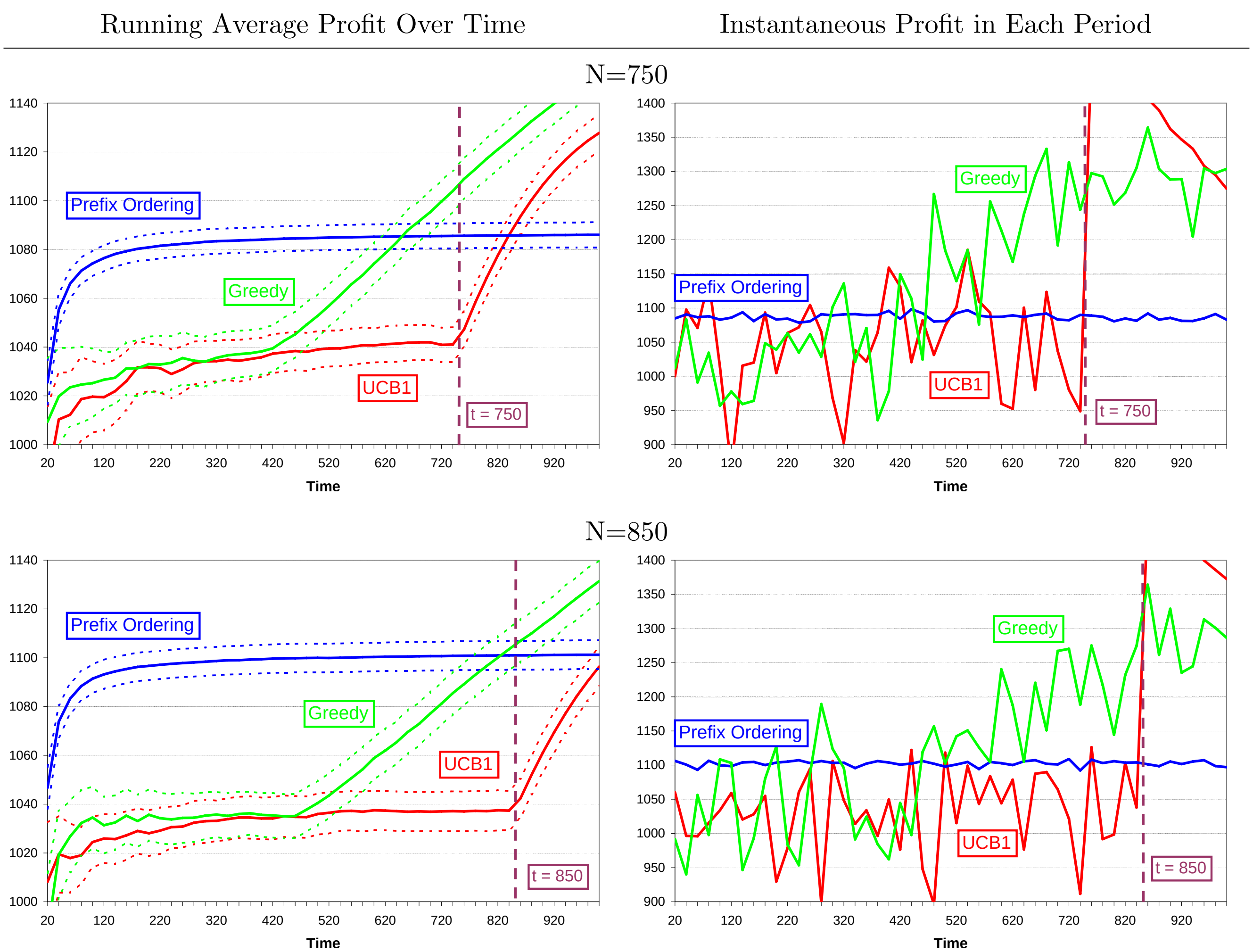


Figure 7: Comparison of the three algorithms as the number of keywords increases from 750 to 850.

5.3 Impact of Randomization and Initialization

We explore the impact of different choices of randomization and initialization by running our algorithm on 100 problem instances under four settings: (a) $\gamma_t = 0$ and $\hat{p}_i^0 = 0$, (b) $\gamma_t = 1/t$ and $\hat{p}_i^0 = 1$, (c)

$\gamma_t = 1/t^2$ and $\hat{p}_i^0 = 1$, and (d) $\gamma_t = 0$ and $\hat{p}_i^0 = 1$. Figure 8 shows the average profit over time in each setting. Recall that our Adaptive Bidding algorithm sets the initial estimate of the clickthru probability to 1, corresponding to cases (b), (c), and (d). In this case, we clearly see the benefits of randomization; allowing for random decisions with small probability (either $\gamma_t = 1/t$ or $1/t^2$) yields significantly higher average profits. It is interesting to note that although setting $\gamma_t = 1/t$ yields slightly higher profits over $\gamma_t = 1/t^2$, the difference is quite small. We also note that the benefit of randomization is quite the opposite of the theoretical result suggested in Theorem 2 and Corollary 1. Reconciling these differences remains an open research question.

When there is no randomization (case (a)), our proposed Adaptive Bidding algorithm continues to converge to the same limit as in the case of randomization (albeit at the expense of slower convergence rate). We believe this happens because we leverage the special structure of our underlying problem by focusing on estimating the parameter $\mathcal{I}_U$.

By setting the initial clickthru probability for each keyword to 1, we limit the number of keywords chosen in each time period. Initially, we will select a small subset of keywords and gradually enlarge the set of keywords as we observe more data on impressions and clicks. The opposite extreme of this approach is to set the initial clickthru probability for each keyword to 0, implying that we will select all keywords in the first time period. Although this setting yields the highest average profit, as seen from the top line in Figure 8, we do not have a formal proof of the convergence under this setting.

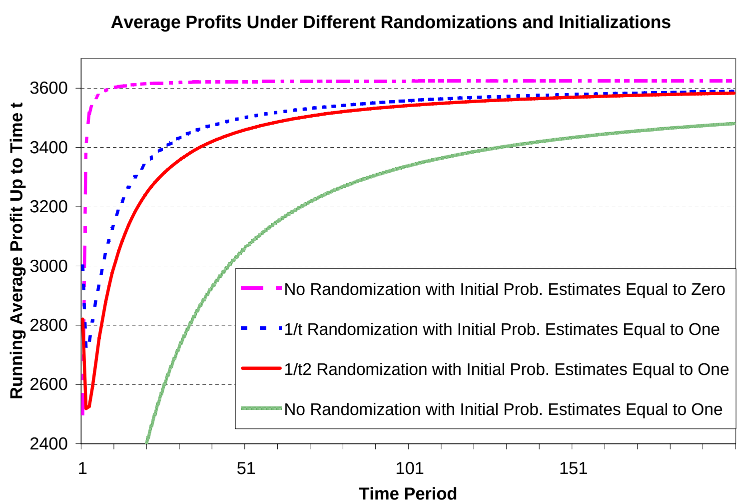


Figure 8: Impact of different randomization and initialization.

6 An Extension to the Case of Unknown Expected Profits

We extend the Adaptive Bidding algorithm to the case when *both* the expected profit π_i and the clickthru probability p_i are *unknown*. With unknown expected profits, we cannot sort the keywords in a descending order of expected-profit-to-cost ratio, and thus, Assumption 1(b) may not hold. Learning the ordering of the expected-profit-to-cost ratios will typically require estimating the profit for each of the N keywords, which is impractical when N is large.

In many applications, however, similar keywords often exhibit similar expected profits. For instance, the expected profit of the keyword "cheap Europe trip" might be closely related to the expected profit of other keywords describing inexpensive travel packages to Europe. Correlations among products have been documented by researches in many industries (see, for example, [6, 23, 28]), and these correlations often arise because the products share common characteristics.

We can express the relationship among the expected profits by grouping the keywords based on the underlying products they aim to promote [25]. A more general approach is to consider a collection of feature vectors that describe characteristics of each keyword, such as the sales, price, and cost of the products targeted by each keyword, or the popularity of the keyword and its corresponding product. We will assume that the expected profit can be written as a a linear combination of the feature vectors, and this is stated in the following assumption. As a convention, all vectors are assumed to be column vectors, and for any vector b, b' denotes its transpose.

Assumption 2 *There exists Q linearly independent vectors $u_1, \ldots, u_Q$ in $\Re^N$ such that for any t and r, the profit random variables $\Pi_r^t = (\Pi_{r1}^t, \ldots \Pi_{rN}^t)'$ associated with each keyword during the arrival of the r^{th} query in period t is given by $\Pi_r^t = \sum_{\ell=1}^Q \gamma_\ell u_\ell + E_r^t$, where $E_r^t = (\epsilon_{r1}^t, \ldots, \epsilon_{rN}^t)'$, and the random variables $\{\epsilon_{ri}^t : t \geq 1, r \geq 1, 1 \leq i \leq N\}$ are independent with ϵ_{ri}^t having a normal distribution with mean 0 and variance σ_i^2.*

Using a linear combination of a tractable number of feature vectors as an approximation has found successes in addressing many large-scale dynamic programming problems. Under Assumption 2, the feature vectors $u_1, \ldots, u_Q$ are known in advance, while the coefficients $(\gamma_1, \ldots, \gamma_Q)$ are *unknown* and must be learned by selecting different keywords and observing the resulting profits. We will present an algorithm that adaptively learns the coefficients γ_ℓ over time. From the above assumption $\pi_r = E\left[\Pi_r^t\right] = \sum_{\ell=1}^Q \gamma_\ell u_\ell$, for any $1 \leq r \leq N$, implying that the vector of expected profit $\Pi = (\pi_1, \ldots, \pi_N)'$ lies in an Q-dimension subspace spanned by the Q feature vectors $u_1, \ldots, u_Q$.

Let us introduce the following notation. Let $U \in \Re^{N \times Q}$ denote an N-by-Q matrix whose columns consist of vectors $u_1, \ldots, u_Q$. Since $u_1, \ldots, u_Q$ are linearly independent by Assumption 2, the matrix U has rank Q. We can thus identify the set of rows $\{i_1, \ldots, i_Q\} \in \{1, 2, \ldots, N\}$ of the matrix U such that the corresponding Q-by-Q sub-matrix is non-singular, and will be denoted by $A_Q \in \Re^{Q \times Q}$. Note that by Assumption 2, the vector of coefficients $(\gamma_1, \ldots, \gamma_Q)$ satisfy the following relationship:

$$\left(\pi_{i_1}, \ldots, \pi_{i_Q}\right)' = A_Q \left(\gamma_1, \ldots, \gamma_Q\right)' \quad \Leftrightarrow \quad \left(\gamma_1, \ldots, \gamma_Q\right)' = A_Q^{-1} \left(\pi_{i_1}, \ldots, \pi_{i_Q}\right)' \tag{3}$$

We now present an algorithm that adaptively learns the ordering of the expected-profit-to-cost ratio and simultaneously optimizes the average expected profit over time. To facilitate our exposition, we will assume that the length of the horizon T has been given in advance. Our result easily generalizes to the case when T is unknown. Our proposed algorithm – which we refer to as Adaptive Bidding With Unknown Profits – consists of two phases. In the first phase, we will focus on estimating the expected profits associated with the keywords $i_1, \ldots, i_Q$ by trying each of the Q keywords for $O(\log(NT))$ periods. After a total of $O(Q \log(NT))$ periods in Phase 1, we then use the average profit of these keywords to estimate the coefficients γ_ℓ's using Equation (3). Once we have estimates of the coefficients, we can infer the estimated expected profit of all keywords, which will allow us to rank the keywords in descending order of estimated-expected-profit-to-cost ratio. Once we have the ranking of keywords, we then apply our Adaptive Bidding from Section 4.2.

Before we proceed to proceed to the formal description of the algorithm, let us introduce the following notation. Let $\tau = (\tau_1, \tau_2, \ldots, \tau_N)$ denote the *unknown* ordering of the expected-profit-to-cost ratio, that is, $\frac{\pi_{\tau_1}}{c_{\tau_1}} \geq \frac{\pi_{\tau_2}}{c_{\tau_2}} \geq \cdots \geq \frac{\pi_{\tau_N}}{c_{\tau_N}}$, and let $\mathcal{I}_\tau^*$ denote the index such that $\mu \sum_{\ell=1}^{\mathcal{I}_\tau^*} c_{\tau_\ell} p_{\tau_\ell} \lambda_{\tau_\ell} \leq 1 < \mu \sum_{\ell=1}^{\mathcal{I}_\tau^*+1} c_{\tau_\ell} p_{\tau_\ell} \lambda_{\tau_\ell}$. As in Assumption 1(c), we will assume that $c_i \leq 1/k$ and $\mu \lambda_i \leq k^\alpha$ for $1 \leq i \leq \mathcal{I}_\tau^* + 1$. Also, let Δ denote the smallest non-zero gap between two consecutive values of the expected-profit-to-cost ratio among the first $\mathcal{I}_\tau^*$ keywords and between the $\mathcal{I}_\tau^{*th}$ and the remaining keywords, that is,

$$\Delta = \min \left\{ \min_{1 \leq i \leq \mathcal{I}_\tau^*} \left\{ \frac{\pi_{\tau_i}}{c_{\tau_i}} - \frac{\pi_{\tau_{i+1}}}{c_{\tau_{i+1}}} : \frac{\pi_{\tau_i}}{c_{\tau_i}} > \frac{\pi_{\tau_{i+1}}}{c_{\tau_{i+1}}} \right\}, \quad \min_{\ell > \mathcal{I}_\tau^*} \left\{ \frac{\pi_{\mathcal{I}_\tau^*}}{c_{\mathcal{I}_\tau^*}} - \frac{\pi_{\tau_\ell}}{c_{\tau_\ell}} : \frac{\pi_{\mathcal{I}_\tau^*}}{c_{\mathcal{I}_\tau^*}} > \frac{\pi_{\tau_\ell}}{c_{\tau_\ell}} \right\} \right\}.$$

Let $\underline{c} = \min_{1 \leq \ell \leq N} \{c_\ell\}$ denote the minimum cost-per-click and let $\sigma^2 = \max_{1 \leq \ell \leq N} \{\sigma_\ell^2\}$ be maximum variance associated with the profit of each keyword. The algorithm is given formally in Figure 9.

The main result of this section is stated in the following theorem whose proof is given in Appendix C. We believe that the requirement of Theorem 1 that each keyword $i_1, \ldots, i_Q$ receives at least T_0 clicks is reasonable because each of these keywords will be selected for T_0 periods during Phase 1.

Theorem 3 *Suppose that each keyword $i_1, \ldots, i_Q$ receives at least T_0 clicks during Phase 1 of the* Adaptive Bidding With Unknown Profits. *Then, under Assumption 2,*

- *With probability at least $1 - 1/T$, the estimated-profit-to-cost ratio $\widehat{\Pi}_i / c_i$ satisfies:*

$$\frac{\widehat{\Pi}_{\tau_1}}{c_{\tau_1}} \geq \frac{\widehat{\Pi}_{\tau_2}}{c_{\tau_2}} \geq \cdots \geq \frac{\widehat{\Pi}_{\tau_{\mathcal{I}_\tau^*}}}{c_{\tau_{\mathcal{I}_\tau^*}}} \geq \max \left\{ \frac{\widehat{\Pi}_{\tau_\ell}}{c_{\tau_\ell}} : \ell \geq \mathcal{I}_\tau^* + 1 \right\}$$

Adaptive Bidding With Unknown Profits

INITIALIZATION: Let the horizon length T be given and let $M = U \cdot A_Q^{-1}$. Also, let $T_0 \equiv \left\lceil \left(8\sigma^2 \max\left\{\sum_{\ell=1}^{Q} M_{i\ell}^2 : 1 \le i \le N\right\} \cdot \ln(2NT)\right) \Big/ \underline{c}^2\Delta^2 \right\rceil$

DESCRIPTION: The algorithm consists of the following two phases.

- Phase 1: This phase consists of $Q \cdot T_0$ periods.

  1) Each of the Q keywords $\{i_1, \ldots, i_Q\}$ will be selected for T_0 periods, that is, for $1 \le \ell \le Q$, we will select the one-keyword subset $\{i_\ell\}$ for T_0 periods. At the end of Phase 1, for $1 \le \ell \le Q$, let $\widehat{\Pi}_{i_\ell}$ denote the average profit associated with clicks on the ad for keyword i_ℓ, that is, $\widehat{\Pi}_{i_\ell}$ equals to the total profits earned from all clicks on the ad for keyword i_ℓ divided by the total number of clicks on the ad. We set $\widehat{\Pi}_{i_\ell}$ to zero if there is no click on the ad for keyword i_ℓ.

  2) Let the estimated coefficients $(\hat{\gamma}_1, \ldots, \hat{\gamma}_Q)'$ and the estimated expected profits $\left(\widehat{\Pi}_1, \ldots, \widehat{\Pi}_N\right)'$ be defined by:

$$(\hat{\gamma}_1, \ldots, \hat{\gamma}_Q)' = A_Q^{-1}\left(\hat{\Pi}_{i_1}, \ldots, \hat{\Pi}_{i_Q}\right)' \quad \text{and} \quad \left(\widehat{\Pi}_1, \ldots \widehat{\Pi}_N\right)' = U(\hat{\gamma}_1, \ldots, \hat{\gamma}_Q)'.$$

- Phase 2: Sort all keywords in a descending of estimated-profit-to-cost ratio $\widehat{\Pi}_i / c_i$ and apply the Adaptive Bidding algorithm for the remaining $T - (Q \cdot T_0)$ periods.

Figure 9: Adaptive Bidding With Unknown Profits algorithm .

- *Let* $\xi = \rho\left(1 - 1/k^{(1+2\alpha)/3} - 3/k^{(1-\alpha)/3}\right)$. *For any* $0 < \epsilon < 1$,

$$\frac{E\left[\sum_{t=1}^{T} Z_{G_t}\right]}{Z^* T} \quad \ge \quad \left(1 - \frac{1}{T}\right)\left(1 - \frac{QT_0}{T}\right)\left(\xi - \frac{\sum_{t=(QT_0)+1}^{T} \gamma_t}{T - QT_0} - M(\epsilon, T - QT_0)\right),$$

where $M(\epsilon, T)$ *is the* same *as the one given in Theorem 2 for the* Adaptive Bidding *algorithm.*

The first part of Theorem 3 shows that we can learn the ordering of the expected-profit-to-cost among the first $\mathcal{I}_\tau^*$ keywords with only $O(QT_0) = O(Q \ln(NT))$ periods, a relatively short time horizon compared with T. We emphasize that our algorithm depends on the number of keywords *only through its logarithm*, and we believe that this should be tractable in most applications. The performance bound given in Theorem 3 consists of three terms. The first term $1 - \frac{1}{T}$ reflects the probability that the ordering computed in Phase 1 of the algorithm based the *estimated-profit-to-cost* ratios coincides with the true ordering. The second term $1 - \frac{QT_0}{T}$ reflects the amount of time in Phase 2 as a fraction of the total time horizon T, representing the amount of time that we do exploitation. And finally, the last term $\xi - \frac{\sum_{t=(QT_0)+1}^{T} \gamma_t}{T-QT_0} - M(\epsilon, T-QT_0)$ reflects the performance bound of the Adaptive Bidding algorithm from Section 4.2 (see Theorem 2), when we apply it for $T - QT_0$ time periods.

Acknowledgements

We are grateful to the Associate Editor and two anonymous referees for their thorough and insightful comments, which lead to significant improvements in the paper. The authors would like to thank Mark Eisner for contacts with companies in search-based advertising services, Bobby Kleinberg for helpful discussions about online optimization with large strategy sets, and Ashish Goel for many helpful comments and suggestions that greatly improve the presentation of our paper. We also thank Shane Henderson for pointers to simulation software packages. To generate random variables in our experiments, we use the *Stochastic Simulation in Java* package developed by Pierre L'Ecuyer at the University of Montreal.

References

[1] G. Aggarwal and J. D. Hartline, "Knapsack Auctions," Workshop on Sponsored Search Auctions, EC 2005.

[2] R. Agrawal, "The continuum-armed bandit problem," *SIAM J. Control and Optimization*, vol. 33, pp. 1926–1951, 1995.

[3] P. Auer, N. Cesa-Bianchi, P. Fischer, "Finite-time Analysis of the Multiarmed Bandit Problem," *Machine Learning*, pp. 235–256, 2002.

[4] P. Auer, N. Cesa-Bianchi, Y. Freund, R. E. Schapire, "Gambling in a rigged casino: The adversarial multi-armed bandit problem," SODA 1995.

[5] M-F. Balcan, A. Blum, J. D. Hartline, Y. Mansour "Sponsored Search Auction Design via Machine Learning," Workshop on Sponsored Search Auctions, EC 2005.

[6] E. Brynjolfsson, Y. (Jeffrey) Hu, M. D. Smith, "Consumer Surplus in the Digital Economy: Estimating the Value of Increased Product Variety at Online Booksellers," *Management Science*, vol. 49, no. 11, pp. 1580-1596, November 2003.

[7] J. Chen, D. Liu, A. B. Whinston, "Designing Share Structure in Auctions of Divisible Goods," Workshop on Sponsored Search Auctions, EC 2005.

[8] B. Dean, M.X. Goemans and J. Vondrak, "Approximating the Stochastic Knapsack Problem: The Benefit of Adaptivity," FOCS 2004, pp. 208–217.

[9] E. Even-Dar, S. Mannor, and Y. Mansour, "PAC bounds for multi-armed bandit and Markov decision processes," *Fifteenth Annual Conference on Computational Learning Theory*, pp. 255-270, 2002.

[10] G. Gallego, "A Minmax Distribution Free Procedure for the (Q, R) Inventory Model," *Operations Research Letters*, vol. 11, pp. 55–60, 1992.

[11] J. Goodman, "Pay-Per-Percentage of Impressions: An Advertising Method that is Highly Robust to Fraud," Workshop on Sponsored Search Auctions, EC 2005.

[12] R.L. Graham, E.L. Lawler, J.K. Lenstra, and A.H.G. Rinnooy Kan, "Optimization and approximation in deterministic sequencing and scheduling: A survey," *Annals of Discrete Mathematics*, 5:287–326, 1979.

[13] B. J. Jansen and M. Resnick, "Examining Searcher Perceptions of and Interactions with Sponsored Results," Workshop on Sponsored Search Auctions, EC 2005.

[14] B. Kitts, P. Laxminarayan, B. LeBlanc, R. Meech, "A Formal Analysis of Search Auctions Including Predictions on Click Fraud and Bidding Tactics," Workshop on Sponsored Search Auctions, EC 2005.

[15] R. Kleinberg, "Online Decision Problems with Large Strategy Sets," Ph.D. disseration, MIT 2005.

[16] R. Kleinberg, "Nearly tight bounds for the continuum-armed bandit problem," in *NIPS* 2005.

[17] S. R. Kulkarni and G. Lugosi, "Finite-time lower bounds for the two-armed bandit problem," *IEEE Trans. Aut. Control*, pp. 711-714, 2000.

[18] T. L. Lai and H. Robbins, "Asymptotically efficient adaptive allocation rules," *Advances in Applied Mathematics*, vol. 6, pp. 4-22, 1985.

[19] S. Mannor and J. N. Tsitsiklis, "Lower bounds on the sample complexity of exploration in the multiarmed bandit problem," *Sixteenth Annual Conference on Computational Learning Theory*, pp. 418-432. Springer, 2003.

[20] C. Meek, D. M. Chickering, D. B. Wilson, "Stochastic and Contingent-Payment Auctions," Workshop on Sponsored Search Auctions, EC 2005.

[21] A. Mehta, A. Saberi, U. Vazirani, and V. Vazirani, "Adwords and Generalized Online Matching," FOCS 2005.

[22] M. Mitzenmacher and E. Upfal, *Probability and Computing: Randomized Algorithms and Probabilistic Analysis*, Cambridge, 2005.

[23] S. Oren, S. Smith, and R. Wilson, "Multi-product Pricing for Electric Power," *Energy Economics*, pp. 104–114, April 1987.

[24] D. C. Parkes and T. Sandholm,"Optimize-and-Dispatch Architecture for Expressive Ad Auctions," Workshop on Sponsored Search, EC 2005.

[25] S. Pandey, D. Chakrabarti, and D. Agarwal, "Multi-armed Bandit Problems with Dependent Arms," *Proceedings of the 24th International Conference on Machine Learning*, Corvallis, OR, 2007.

[26] D. Pollard, *Convergence of Stochastic Processes*, Springer Verlang, 1994.

[27] S. Sahni, "Approximate Algorithms for the 0/1 Knapsack Problem," *Journal of the ACM*, 22:115–124, 1975.

[28] C. A. Yano and G. Dobson, "Profit Optimizing Product Line Design, Selection and Pricing with Manufacturing Cost Considerations: A Survey," *Product Variety Management: Research and Advances* (T.-H. Ho and C.S. Tang, eds.), Kluwer Academic Publisher, November 1998, pp. 145-176.

[29] M. Zinkevich, "Online convex programming and generalized infinitisimal gradient ascent," in *Proceeding of the 20th International Conference on Machine Learning*, pp. 928–936, 2003.

Appendix

To facilitate the reading, we will provide restatements of the results that we wish to prove.

A Proofs of Results in Section 3

A.1 Proof of Lemma 1

Restatement of Lemma 1: *For any* $A \subseteq \{1, 2, \ldots, N\}$, $r \geq 1$, *and* $i \in A$, $\lambda_i p_i \mathcal{P}\left\{B_r^A \geq c_i \middle| S \geq r\right\} = \mathcal{P}\left\{Q_r = i, B_r^A \geq c_i, X_{ri} = 1 \middle| S \geq r\right\}$, *and* $E[Z_A] = \sum_{i \in A} \pi_i \lambda_i p_i \sum_{r=1}^{\infty} E\left[\mathbf{1}\left(B_r^A \geq c_i, S \geq r\right)\right]$.

Proof: Recall that B_r^A denotes the remaining account balance *when the* r^{th} *search query arrives* and we select the keywords in A. This implies that the event $\{B_r^A \geq c_i\}$ depends only on the previous $r-1$ search queries, and does *not* depend on whether the consumer clicks on the ad associated with keyword i when the r^{th} query arrives *nor* does it depend on what the r^{th} query corresponds to. Thus,

$$\begin{aligned} \mathcal{P}\left\{Q_r = i, B_r^A \geq c_i, X_{ri} = 1 \middle| S \geq r\right\} &= \mathcal{P}\left\{B_r^A \geq c_i \middle| S \geq r\right\} \mathcal{P}\left\{Q_r = i, X_{ri} = 1 \middle| S \geq r\right\} \\ &= \lambda_i p_i \mathcal{P}\left\{B_r^A \geq c_i \middle| S \geq r\right\} \end{aligned}$$

where the last equality follows from our assumption that the clickthrus of the ads are independent of the keyword corresponding to the r^{th} search query, and both of these random variables are independent of the total number of arrivals S. The next result of the lemma follows directly from the above result and the fact that Π_{ri} is independent of all the other random variables and $E[\Pi_{ri}] = \pi_i$. ■

A.2 Proof of Theorem 1

Restatement of Theorem 1: *Let k and α be defined as in Assumption 1. Suppose the probabilities p_i and the expected profits π_i are known and $1/k + 1/k^{1-\alpha} + 1/k^{(1-\alpha)/3} \leq 1$. Let Z^* denote the optimal profit, $P = \{1, 2, \ldots, \mathcal{I}_U\}$, and $\rho = E[\min\{S, \mu\}]/\mu$. Then,*

$$\rho\left(1 - \frac{1}{k^{(1+2\alpha)/3}} - \frac{2}{k^{(1-\alpha)/3}}\right) Z^* \leq E[Z_P] \leq Z^*.$$

The proof of Theorem 1 relies on the following lemmas. The first lemma establishes an upper bound on the optimal expected profit in terms of the optimal value of a linear program.

Lemma 8 *Let* $Z^{LP} \equiv \max\left\{\sum_{i=1}^N \pi_i \lambda_i p_i z_i \,\middle|\, \sum_{i=1}^N c_i \lambda_i p_i z_i \leq 1,\ 0 \leq z_i \leq \mu,\ \forall i\right\}$. *Then,* $Z^* \leq Z^{LP}$.

Proof: For any A, let the random variable C_A denote the total cost associated with bidding on keywords in the set A. By definition, we know that $C_A \leq 1$ with probability 1, which implies that

$$1 \geq E[C_A] = E\left[\sum_{r=1}^{S} \sum_{i=1}^{N} c_i \mathbf{1}\left(i \in A, Q_r = i, B_r^A \geq c_i, X_{ri} = 1\right)\right] = \sum_{i \in A} c_i \lambda_i p_i \sum_{r=1}^{\infty} E\left[\mathbf{1}\left(B_r^A \geq c_i, S \geq r\right)\right]$$

where the last equality follows from Lemma 1. Note that for any i, $\sum_{r=1}^{\infty} E\left[\mathbf{1}\left(B_r^A \geq c_i, S \geq r\right)\right] \leq \sum_{r=1}^{\infty} E\left[\mathbf{1}\left(S \geq r\right)\right] = E[S] = \mu$. For any $1 \leq i \leq N$, let $0 \leq z_i \leq \mu$ be defined by $z_i = \mathbf{1}(i \in A) \cdot \sum_{r=1}^{\infty} E\left[\mathbf{1}\left(B_r^A \geq c_i, S \geq r\right)\right]$. Then, we have that $\sum_{i=1}^{N} c_i p_i \lambda_i z_i \leq 1$ and also that

$$\begin{aligned} E\left[Z_A\right] &= E\left[\sum_{r=1}^{S}\sum_{i=1}^{N} \Pi_i \mathbf{1}\left(i \in A, Q_r = i, B_r^A \geq c_i, X_{ri} = 1\right)\right] \\ &= \sum_{i \in A} \pi_i \lambda_i p_i \sum_{r=1}^{\infty} E\left[\mathbf{1}\left(B_r^A \geq c_i, S \geq r\right)\right] = \sum_{i=1}^{N} \pi_i \lambda_i p_i z_i \end{aligned}$$

Since A is arbitrary, the desired result follows. ■

The next lemma relates the optimal value Z^{LP} of the linear program defined in Lemma 8 to the expected profits obtained when we select keywords according to the prefix ordering.

Lemma 9 *Let Z^{LP} denote the optimal value of the linear program defined in Lemma 8. For any $u \leq \mathcal{I}_U$, $\sum_{i=1}^{u} \pi_i \lambda_i p_i \geq \left(\sum_{i=1}^{u} c_i \lambda_i p_i\right) Z^{LP} \geq \left(\sum_{i=1}^{u} c_i \lambda_i p_i\right) Z^*$.*

Proof: Since the second inequality follows directly from Lemma 8, it suffices to prove only the first inequality. From Assumption 1(b), $\mathcal{I}_U \leq \mathcal{I}^*$ where $\mathcal{I}^*$ is the index such that $\mu \sum_{i=1}^{\mathcal{I}^*} c_i \lambda_i p_i \leq 1 < \mu \sum_{i=1}^{\mathcal{I}^*+1} c_i \lambda_i p_i$. Since $\pi_1/c_1 \geq \pi_2/c_2 \geq \cdots \geq \pi_{\mathcal{I}^*}/c_{\mathcal{I}^*} \geq \max\{\pi_\ell/c_\ell : \ell > \mathcal{I}^*\}$, it follows from the definition of Z^{LP} that

$$Z^{LP} = \mu \sum_{i=1}^{\mathcal{I}^*} \pi_i \lambda_i p_i + \pi_m \lambda_m p_m \left(\frac{1 - \mu \sum_{i=1}^{\mathcal{I}^*} c_i \lambda_i p_i}{c_m \lambda_m p_m}\right),$$

where $m > \mathcal{I}^*$ is the index such that $\pi_m/c_m = \max\{\pi_\ell/c_\ell : \ell > \mathcal{I}^*\}$. Since $u \leq \mathcal{I}_U \leq \mathcal{I}^*$, we have $\frac{\sum_{i=1}^{u} \pi_i \lambda_i p_i}{\sum_{i=1}^{u} c_i \lambda_i p_i} \geq \frac{\sum_{i=u+1}^{\mathcal{I}^*} \pi_i \lambda_i p_i}{\sum_{i=u+1}^{\mathcal{I}^*} c_i p_i \lambda_i}$ and $\frac{\sum_{i=1}^{u} \pi_i \lambda_i p_i}{\sum_{i=1}^{u} c_i \lambda_i p_i} \geq \frac{\pi_m \lambda_m p_m}{c_m \lambda_m p_m}$, where we define $0/0 = 1$. The above result implies that $\sum_{i=u+1}^{\mathcal{I}^*} \pi_i \lambda_i p_i \leq \left(\sum_{i=1}^{u} \pi_i \lambda_i p_i\right)\left(\frac{\sum_{i=u+1}^{\mathcal{I}^*} c_i \lambda_i p_i}{\sum_{i=1}^{u} c_i \lambda_i p_i}\right)$ and $\pi_m \lambda_m p_m \leq \left(\sum_{i=1}^{u} \pi_i \lambda_i p_i\right)\left(\frac{c_m \lambda_m p_m}{\sum_{i=1}^{u} c_i \lambda_i p_i}\right)$. Putting everything together, we have

$$\begin{aligned} Z^{LP} &= \mu \sum_{i=1}^{u} \pi_i \lambda_i p_i + \mu \sum_{i=u+1}^{\mathcal{I}^*} \pi_i \lambda_i p_i + \pi_m \lambda_m p_m \left(\frac{1 - \mu \sum_{i=1}^{\mathcal{I}^*} c_i \lambda_i p_i}{c_m \lambda_m p_m}\right) \\ &\leq \left(\mu \sum_{i=1}^{u} \pi_i \lambda_i p_i\right)\left(1 + \frac{\sum_{i=u+1}^{\mathcal{I}^*} c_i \lambda_i p_i}{\sum_{i=1}^{u} c_i \lambda_i p_i} + \frac{1}{\mu}\left(\frac{c_m \lambda_m p_m}{\sum_{i=1}^{u} c_i \lambda_i p_i} \cdot \frac{1 - \mu \sum_{i=1}^{\mathcal{I}^*} c_i \lambda_i p_i}{c_{m+1} \lambda_m p_m}\right)\right) \\ &\leq \left(\mu \sum_{i=1}^{u} \pi_i \lambda_i p_i\right)\left(1 + \frac{\sum_{i=u+1}^{\mathcal{I}^*} c_i \lambda_i p_i}{\sum_{i=1}^{u} c_i \lambda_i p_i} + \frac{1 - \mu \sum_{i=1}^{\mathcal{I}^*} c_i \lambda_i p_i}{\mu \sum_{i=1}^{u} c_i \lambda_i p_i}\right) = \left(\mu \sum_{i=1}^{u} \pi_i \lambda_i p_i\right)\left(\frac{1}{\mu \sum_{i=1}^{u} c_i \lambda_i p_i}\right). \end{aligned}$$

■

The final lemma provides a lower bound on the probability $\mathcal{P}\left\{B_r^H \geq c_i \middle| S \geq r\right\}$ that we have enough money remaining if the r^{th} search query corresponds to keyword i, given that we bid on a set of keywords H and $r \leq \mu$. Since the proof is fairly technical, the details are given in Appendix A.2.1.

Lemma 10 *Let $H = \{1, \ldots, u\}$ where u is any positive integer such that $u \le \mathcal{I}_U$. Then, for any keyword $i \in H$ and $1 \le r \le \mu$, $\mathcal{P}\{B_r^H \ge c_i \mid S \ge r\} \ge 1 - 1/k^{(1+2\alpha)/3}$.*

Here is the proof of Theorem 1.

Proof: By the hypothesis $1/k^{1-\alpha} \le 1 - 1/k - 1/k^{(1-\alpha)/3}$, $\mathcal{I}_U$ is thus well-defined. Therefore,

$$\begin{aligned} E\left[Z_P\right] &= \sum_{i\in P}\pi_i\lambda_i p_i \sum_{r=1}^{\infty}\mathcal{P}\left\{B_r^P \ge c_i, S \ge r\right\} \ge \sum_{i\in P}\pi_i\lambda_i p_i \sum_{r=1}^{\lfloor\mu\rfloor}\mathcal{P}\left\{B_r^P \ge c_i, S \ge r\right\} \\ &\ge \left(1 - \frac{1}{k^{(1+2\alpha)/3}}\right)\left(\sum_{r=1}^{\lfloor\mu\rfloor}\mathcal{P}\left\{S \ge r\right\}\right)\sum_{i\in P}\pi_i\lambda_i p_i = \rho\left(1 - \frac{1}{k^{(1+2\alpha)/3}}\right)\mu\sum_{i\in P}\pi_i\lambda_i p_i \end{aligned}$$

where the second inequality follows from Lemma 10 which shows that $\mathcal{P}\left\{B_r^P \ge c_i \middle| S \ge r\right\} \ge 1 - 1/k^{(1+2\alpha)/3}$. The second equality follows from the fact that $\sum_{r=1}^{\lfloor\mu\rfloor}\mathcal{P}\left\{S \ge r\right\} = E\left[\min\left\{S, \mu\right\}\right] = \rho\mu$ by definition of ρ. Using Lemma 9, we conclude that

$$E\left[Z_P\right] \ge \rho\left(1 - \frac{1}{k^{(1+2\alpha)/3}}\right)Z^*\mu\sum_{i\in P}c_i\lambda_i p_i \ge \rho\left(1 - \frac{1}{k^{(1+2\alpha)/3}} - \frac{2}{k^{(1-\alpha)/3}}\right)Z^*,$$

which is the desired result. The last inequality follows from Assumption 1(c), which implies that, for any keyword $i \le \mathcal{I}^*$, $\mu c_i\lambda_i p_i \le 1/k^{1-\alpha}$; it follows from the definition of $\mathcal{I}_U$ that $\mu\sum_{i=1}^{\mathcal{I}_U}c_i\lambda_i p_i \ge 1 - 1/k - 1/k^{(1-\alpha)/3} - 1/k^{1-\alpha}$. Since $k \ge 1$, $0 < \alpha < 1$, and $1/k^{(1-\alpha)/3} \ge 1/k^{1-\alpha}$, we have that

$$\begin{aligned} \left(1 - \frac{1}{k^{(1+2\alpha)/3}}\right)\left(1 - \frac{1}{k} - \frac{1}{k^{(1-\alpha)/3}} - \frac{1}{k^{1-\alpha}}\right) &\ge \left(1 - \frac{1}{k^{(1+2\alpha)/3}}\right)\left(1 - \frac{1}{k} - \frac{2}{k^{(1-\alpha)/3}}\right) \\ &\ge 1 - \frac{1}{k^{(1+2\alpha)/3}} - \frac{2}{k^{(1-\alpha)/3}} + \frac{2}{k^{(2+\alpha)/3}} - \frac{1}{k} \\ &\ge 1 - \frac{1}{k^{(1+2\alpha)/3}} - \frac{2}{k^{(1-\alpha)/3}}. \end{aligned}$$

■

A.2.1 Proof of Lemma 10

The proof of Lemma 10 makes use of the following result.

Lemma 11 *For any subset A, $Var\left(\sum_{s=1}^{r-1}\sum_{i\in A}c_i\mathbf{1}\left(Q_s = i, X_{si} = 1\right)\middle| S \ge r\right) \le \frac{r-1}{k}\sum_{i\in A}c_i\lambda_i p_i$.*

Proof: Since Q_s's are independent and for any i, X_{si}'s are also independent, it suffices to show that for any $s < r$, $Var\left(\sum_{i\in A}c_i\mathbf{1}\left(Q_s = i, X_{si} = 1\right)\middle| S \ge r\right) \le \frac{1}{k}\sum_{i\in A}c_i\lambda_i p_i$. Note that since Q_s and X_{si} are independent, we have that $E\left[\mathbf{1}\left(Q_s = i, X_{si} = 1\right)|S \ge r\right] = \lambda_i p_i$, which implies that

$$Var\left(\sum_{i\in A}c_i\mathbf{1}\left(Q_s = i, X_{si} = 1\right)\middle| S \ge r\right)$$

$$
\begin{aligned}
&= E\left[\left(\sum_{i\in A} c_i\left(\mathbf{1}\left(Q_s=i,X_{si}=1\right)-p_i\lambda_i\right)\right)^2 \middle| S\geq r\right]\\
&= \sum_{i\in A} c_i^2 E\left[\left(\mathbf{1}\left(Q_s=i,X_{si}=1\right)-p_i\lambda_i\right)^2 \middle| S\geq r\right]\\
&\quad + \sum_{i,j\in A: i\neq j} c_i c_j E\left[\left(\mathbf{1}\left(Q_s=i,X_{si}=1\right)-\lambda_i p_i\right)\cdot\left(\mathbf{1}\left(Q_s=j,X_{sj}=1\right)-\lambda_j p_j\right) \middle| S\geq r\right]\\
&= \sum_{i\in A} c_i^2\lambda_i p_i\left(1-\lambda_i p_i\right) - \sum_{i,j\in A:i\neq j} c_i c_j \lambda_i p_i \lambda_j p_j \;\leq\; \frac{1}{k}\sum_{i\in A} c_i\lambda_i p_i,
\end{aligned}
$$

where the third equality follows from the fact that $\mathbf{1}(Q_s=i)\mathbf{1}(Q_s=j)=0$ for all $i\neq j$. The final inequality follows from Assumption 1(c) that $c_i\leq 1/k$ for all i. ■

Here is the proof of Lemma 10.

Proof: For any keyword $i\in H$, let T_r^H denote the total amount of money that we have already spent when r^{th} search query arrives. Then, $\mathcal{P}\left\{B_r^H\geq c_i\middle| S\geq r\right\} = 1-\mathcal{P}\left\{T_r^H>1-c_i\middle| S\geq r\right\}$. We will focus on developing an upper bound for the expression $\mathcal{P}\left\{T_r^H>1-c_i\middle| S\geq r\right\}$. By definition of T_r^H, we know that, with probability 1, $T_r^H=\sum_{s=1}^{r-1}\sum_{i\in H} c_i\mathbf{1}\left(Q_s=i, B_r^H\geq c_i, X_{si}=1\right)\leq \sum_{s=1}^{r-1}\sum_{i\in H} c_i\mathbf{1}\left(Q_s=i,X_{si}=1\right)$. Moreover, by the hypothesis of the lemma,

$$
E\left[\sum_{s=1}^{r-1}\sum_{i\in H} c_i\mathbf{1}\left(Q_s=i,X_{si}=1\right)\middle| S\geq r\right] = (r-1)\sum_{i\in H} c_i\lambda_i p_i \leq \mu\sum_{i\in H} c_i\lambda_i p_i \leq 1-\frac{1}{k}-\frac{1}{k^{(1-\alpha)/3}},
$$

and it follows from Chebyshev's Inequality and the previous lemma that

$$
\begin{aligned}
\mathcal{P}\left\{T_r^H>1-c_i\middle| S\geq r\right\} &\leq \mathcal{P}\left\{\sum_{s=1}^{r-1}\sum_{i\in H} c_i\mathbf{1}\left(Q_s=i,X_{si}=1\right)>1-c_i\middle| S\geq r\right\}\\
&\leq \frac{Var\left(\sum_{s=1}^{r-1}\sum_{i\in H} c_i\mathbf{1}\left(Q_s=i,X_{si}=1\right)\middle| S\geq r\right)}{\left(1-c_i-(r-1)\sum_{i\in H} c_i p_i\lambda_i\right)^2}\\
&\leq \frac{\frac{r-1}{k}\sum_{i\in H} c_i\lambda_i p_i}{\left(1-c_i-(r-1)\sum_{i\in H} c_i p_i\lambda_i\right)^2} \;\leq\; \frac{\frac{\mu}{k}\sum_{i\in H} c_i\lambda_i p_i}{\left(1-\frac{1}{k}-\mu\sum_{i\in H} c_i p_i\lambda_i\right)^2},
\end{aligned}
$$

where the last inequality follows from the fact that $r\leq\mu$ and $c_i\leq 1/k$ (Assumption 1(c)). Therefore,

$$
\mathcal{P}\left\{T_r^H>1-c_i\middle| S\geq r\right\} \;\leq\; \frac{\frac{\mu}{k}\sum_{i\in H} c_i\lambda_i p_i}{\left(1-\frac{1}{k}-\mu\sum_{i\in H} c_i p_i\lambda_i\right)^2} \leq \frac{k^{(2-2\alpha)/3}}{k}\mu\sum_{i\in H} c_i\lambda_i p_i \;\leq\; \frac{1}{k^{(1+2\alpha)/3}},
$$

where the second inequality follows from the fact that $H=\{1,2,\ldots,u\}$ where $u\leq\mathcal{I}_U$, which implies that $\mu\sum_{i\in H} c_i\lambda_i p_i\leq 1-1/k-1/k^{(1-\alpha)/3}$. Therefore, $1/k^{(1-\alpha)/3}\leq 1-1/k-\mu\sum_{i\in H} c_i\lambda_i p_i$. The final inequality follows from the fact that $\mu\sum_{i\in H} c_i\lambda_i p_i\leq 1$. ■

B Proof of Theorem 2

Restatement of Theorem 2: Let $(G_t : t \geq 1)$ denote the sequence of decisions generated by the Adaptive Bidding algorithm. Under Assumption 1, if $1/k + 1/k^{1-\alpha} + 2/k^{(1-\alpha)/3} \leq 1$, then for any $T \geq 1$ and $0 < \epsilon < 1$,

$$\frac{\sum_{t=1}^{T} E\left[Z_{G_t}\right]}{TZ^*} \geq \rho\left(1 - \frac{1}{k^{(1+2\alpha)/3}} - \frac{3}{k^{(1-\alpha)/3}}\right) - \frac{\sum_{t=1}^{T}\gamma_t}{T} - M(\epsilon, T),$$

where

$$M(\epsilon, T) = \frac{\epsilon\rho\mathcal{I}_U}{k^{1-\alpha}} + \frac{(128/\rho^2)\ L}{\left(\min_{1\leq i\leq \mathcal{I}_U} \pi_i\lambda_i p_i\right)\ kT} + \frac{12\ \mathcal{I}_U\lambda^*\mu/\left(1 - e^{-\lambda^*\mu}\right)}{\epsilon^2 kT}.$$

The proof of Theorem 2 relies on the following three lemmas. The first lemma establishes an upper bound on the probability that $\ell_t > \mathcal{I}_U$. The proof appears in Appendix B.1.

Lemma 12 *For any $t \geq 1$, $\mathcal{P}\left\{\ell_t > \mathcal{I}_U\right\} \leq 1/k^{(1-\alpha)/3}$.*

To facilitate our exposition, we will use the following notation throughout Section B. For any $T \geq 1$, let the random variable $\mathcal{J}_T \subseteq \{1, 2, \ldots T\}$ denote the set of time periods when the index of the last keyword chosen based on the prefix ordering is less than or equal to $\mathcal{I}_U$, that is, $\mathcal{J}_T = \{t \leq T : F_t = 1, \ell_t \leq \mathcal{I}_U\}$. Note that when $F_t = 1$, the algorithm chooses the decision based on the prefix ordering, that is, $g_t = \ell_t$. The result from Lemma 12 shows that $\mathcal{J}_T$ will contain a large fraction of time periods, enabling us to restrict our attention to only time periods in $\mathcal{J}_T$. Also, for any keyword i, let $\mathcal{E}_{i,T} = E\left[\sum_{t\in\mathcal{J}_T : g_t \geq i}\left|p_i - \hat{p}_i^{t-1}\right|\right]$ denote the expected cumulative errors during time periods in $\mathcal{J}_T$ between our estimate of the clickthru probability for keyword i and its true value. The following lemma relates the performance of our algorithm to the accuracy of our estimates of the clickthru probabilities associated with keywords $1, \ldots, \mathcal{I}_U$ during time periods in $\mathcal{J}_T$. The proof of this result appears in Appendix B.2.

Lemma 13 *For any $T \geq 1$,*

$$\frac{\sum_{t=1}^{T} E\left[Z_{G_t}\right]}{TZ^*} \quad \geq \quad \rho\left(1 - \frac{1}{k^{(1+2\alpha)/3}} - \frac{3}{k^{(1-\alpha)/3}}\right) - \frac{\sum_{t=1}^{T}\gamma_t}{T} - \frac{\rho}{T}\left(1 - \frac{1}{k^{(1+2\alpha)/3}}\right)\mu\sum_{i=1}^{\mathcal{I}_U} c_i\lambda_i\mathcal{E}_{i,T}$$

The next result establishes an upper bound on the accumulated difference between the empirical estimate of the clickthru probability $\hat{p}_i^{t-1}$ and its true value p_i, corresponding to the second term in the above lemma. The proof appears in Appendix B.3.

Lemma 14 *For any $T \geq 1$ and $0 < \epsilon < 1$,*

$$\frac{\rho}{T}\left(1 - \frac{1}{k^{(1+2\alpha)/3}}\right)\mu\sum_{i=1}^{\mathcal{I}_U} c_i\lambda_i\mathcal{E}_{i,T} \leq \frac{\epsilon\rho\mathcal{I}_U}{k^{1-\alpha}} + \frac{(128/\rho^2)\,L}{\left(\min_{1\leq i\leq \mathcal{I}_U}\pi_i\lambda_i p_i\right) kT} + \frac{12\mathcal{I}_U\lambda^*\mu/\left(1 - e^{-\lambda^*\mu}\right)}{\epsilon^2 kT}.$$

The proof of Theorem 2 follows immediately from Lemma 13 and 14.

B.1 Proof of Lemma 12

Proof: For any keyword u, let $Y_u(t)$ denote the *cumulative* number of impressions that the ad associated with keyword u has received *up to the end of period* t. It follows from the definition of ℓ_t that

$$
\begin{aligned}
\mathcal{P}\{\ell_t > \mathcal{I}_U\} &= \mathcal{P}\left\{\mu \sum_{u=1}^{\mathcal{I}_u+1} c_u \lambda_u \hat{p}_u^{t-1} \le 1 - \frac{1}{k} - \frac{2}{k^{(1-\alpha)/3}}\right\} \\
&= \mathcal{P}\left\{\mu \sum_{u=1}^{\mathcal{I}_u+1} c_u \lambda_u \left(p_u - \hat{p}_u^{t-1}\right) \ge \mu \sum_{u=1}^{\mathcal{I}_U+1} c_u \lambda_u p_u - \left(1 - \frac{1}{k} - \frac{2}{k^{(1-\alpha)/3}}\right)\right\} \\
&\le \mathcal{P}\left\{\mu \sum_{u=1}^{\mathcal{I}_u+1} c_u \lambda_u \left(p_u - \hat{p}_u^{t-1}\right) > \frac{1}{k^{(1-\alpha)/3}}\right\} \\
&\le \mathcal{P}\left\{\mu \sum_{u=1}^{\mathcal{I}_U+1} c_u \lambda_u \left(p_u - \hat{p}_u^{t-1}\right) \mathbf{1}\left(Y_u(t-1) \ge 1\right) > \frac{1}{k^{(1-\alpha)/3}}\right\},
\end{aligned}
$$

where the first inequality follows from the definition of $\mathcal{I}_U$, which implies that $1 - 1/k - 1/k^{(1-\alpha)/3} < \mu \sum_{u=1}^{\mathcal{I}_U+1} c_u \lambda_u p_u$. Therefore, we have that $\mu \sum_{u=1}^{\mathcal{I}_U+1} c_u \lambda_u p_u - \left(1 - \frac{1}{k} - \frac{2}{k^{(1-\alpha)/3}}\right) > \frac{1}{k^{(1-\alpha)/3}}$. The final inequality follows from the fact that for any keyword u such that $Y_u(t-1) = 0$, we have $\hat{p}_u^{t-1} = 1$, which implies that $p_u - \hat{p}_u^t \le 0$. By Markov's Inequality, we have

$$
\begin{aligned}
\mathcal{P}\{\ell_t > \mathcal{I}_U\} &\le \mathcal{P}\left\{\mu \sum_{u=1}^{\mathcal{I}_U+1} c_u \lambda_u \left(p_u - \hat{p}_u^{t-1}\right) \mathbf{1}\left(Y_u(t-1) \ge 1\right) > \frac{1}{k^{(1-\alpha)/3}}\right\} \\
&\le \frac{E\left[\left\{\mu \sum_{u=1}^{\mathcal{I}_U+1} c_u \lambda_u \left(p_u - \hat{p}_u^{t-1}\right) \mathbf{1}\left(Y_u(t-1) \ge 1\right)\right\}^2\right]}{\left(1/k^{(1-\alpha)/3}\right)^2} \\
&= k^{(2-2\alpha)/3} \sum_{u=1}^{\mathcal{I}_U+1} \left(\mu c_u \lambda_u\right)^2 E\left[\left(p_u - \hat{p}_u^{t-1}\right)^2 \mathbf{1}\left(Y_u(t-1) \ge 1\right)\right] \\
&\quad + 2k^{(2-2\alpha)/3} \sum_{u,v:u<v} \mu^2 c_u c_v \lambda_u \lambda_v E\left[\left(p_u - \hat{p}_u^{t-1}\right)\left(p_v - \hat{p}_v^{t-1}\right) \mathbf{1}\left(Y_u(t-1) \ge 1, Y_v(t-1) \ge 1\right)\right]
\end{aligned}
$$

Now, consider any $1 \le u < v \le \mathcal{I}_U + 1$,

$$
\begin{aligned}
& E\left[\left(p_u - \hat{p}_u^{t-1}\right)\left(p_v - \hat{p}_v^{t-1}\right) \mathbf{1}\left(Y_u(t-1) \ge 1, Y_v(t-1) \ge 1\right)\right] \\
&= \sum_{y_u \ge 1} \sum_{y_v \ge 1} \mathcal{P}\left\{Y_u(t-1) = y_u, Y_v(t-1) = y_v\right\} \\
&\qquad \times E\left[\left(p_u - \hat{p}_u^{t-1}\right)\left(p_v - \hat{p}_v^{t-1}\right) \middle| Y_u(t-1) = y_u, Y_v(t-1) = y_v\right] \\
&= \sum_{y_u \ge 1} \sum_{y_v \ge 1} \mathcal{P}\left\{Y_u(t-1) = y_u, Y_v(t-1) = y_v\right\} \\
&\qquad \times E\left[\left(\frac{1}{y_u} \sum_{s=1}^{y_u} \left(X_u^s - p_u\right)\right)\left(\frac{1}{y_v} \sum_{s=1}^{y_v} \left(X_v^s - p_v\right)\right) \middle| Y_u(t-1) = y_u, Y_v(t-1) = y_v\right] \\
&= 0,
\end{aligned}
$$

where X_u^s is an indicator random variable indicating whether or not the consumer clicks on the ad associated with keyword u during the s^{th} time when the ad for keyword u receives an impression. The final equality follows from the fact X_u^s and X_v^s are independent of $Y_u(t)$ and $Y_v(t)$. Moreover, X_u^s and X_v^t are also independent for any s and t and $E\left[X_u^s - p_u\right] = E\left[X_v^s - p_v\right] = 0$ for all s. In addition,

$$\begin{aligned}
&E\left[\left(p_u - \hat{p}_u^{t-1}\right)^2 \mathbf{1}\left(Y_u(t-1) \geq 1\right)\right] \\
&= \sum_{y_u \geq 1} \mathcal{P}\left\{Y_u(t-1) = y_u\right\} E\left[\left(p_u - \hat{p}_u^{t-1}\right)^2 \middle| Y_u(t-1) = y_u\right] \\
&= \sum_{y_u \geq 1} \mathcal{P}\left\{Y_u(t-1) = y_u\right\} E\left[\left(\frac{1}{y_u}\sum_{s=1}^{y_u}\left(X_u^s - p_u\right)\right)\left(\frac{1}{y_u}\sum_{s=1}^{y_u}\left(X_u^s - p_u\right)\right) \middle| Y_u(t-1) = y_u\right] \\
&= \sum_{y_u \geq 1} \mathcal{P}\left\{Y_u(t-1) = y_u\right\} E\left[\frac{1}{y_u^2}\sum_{s=1}^{y_u}\left(X_u^s - p_u\right)^2 \middle| Y_u(t-1) = y_u\right] \\
&= \sum_{y_u \geq 1} \mathcal{P}\left\{Y_u(t-1) = y_u\right\} \frac{p_u(1-p_u)}{y_u} \;\leq\; p_u
\end{aligned}$$

where the third equality follows from the fact that X_u^s's are independent and identically distributed Bernoulli random variables with parameter p_u and they are independent of $Y_u(t)$. Moreover, $E\left[X_u^s - p_u\right] = 0$ for all s. The fourth equality follows from the fact that X_u^s are Bernoulli random variable with parameter p_u. Putting everything together, we have that

$$\begin{aligned}
\mathcal{P}\left\{\ell_t > \mathcal{I}_U\right\} &\leq k^{(2-2\alpha)/3} \sum_{u=1}^{\mathcal{I}_U+1} \left(\mu c_u \lambda_u\right)^2 p_u \;\leq\; k^{(2-2\alpha)/3} \frac{1}{k^{1-\alpha}} \sum_{u=1}^{\mathcal{I}_U+1} \mu c_u \lambda_u p_u \\
&= \frac{1}{k^{(1-\alpha)/3}} \mu \sum_{u=1}^{\mathcal{I}_U+1} c_u \lambda_u p_u \;\leq\; \frac{1}{k^{(1-\alpha)/3}}
\end{aligned}$$

where the third inequality follows from Assumption 1(c), which implies that, for any keyword $i \leq \mathcal{I}^* + 1$, $\mu c_i \lambda_i \leq 1/k^{1-\alpha}$. The final inequality follows from the definition of $\mathcal{I}_U$, which implies that $\mu \sum_{u=1}^{\mathcal{I}_U+1} c_u \lambda_u p_u \leq 1 - \frac{1}{k} - \frac{1}{k^{(1-\alpha)/3}} + \frac{1}{k^{1-\alpha}} \leq 1$ ■

B.2 Proof of Lemma 13

Proof: Recall that, for any $t \geq 1$, we have $G_t = \{1, \ldots, g_t\}$. Also, let S^t denote the total number of search queries in time t, and let $B_r^{g_t}$ denote the remaining account balance in period t just before the arrival of the r^{th} search query associated with keyword i (in period t), assuming that we have bid on the set of keywords in $\{1, 2, \ldots, g_t\}$ in period t. Also, let the random variable Q_r^t denote the keyword associated with the r^{th} search query in period t. Also, let X_{ri}^t denote a Bernoulli random variable indicating whether or not the consumer clicks on the ad associated with keyword i during the arrival of the r^{th} search query in period t. Then, it follows from Lemma 1 that $\sum_{t=1}^T E\left[Z_{G_t}\right] = \sum_{t=1}^T E\left[\sum_{i=1}^{g_t} \pi_i \lambda_i p_i \sum_{r=1}^{\infty} \mathbf{1}\left(B_r^{g_t} \geq c_i, S^t \geq r\right)\right]$. Recall that $J_T = \{t \leq T : F_t = 1 \,\text{and}\, \ell_t \leq \mathcal{I}_U\}$, where the binary random variable $F_t \in \{0,1\}$ indicates whether or not we choose a decision based on the prefix ordering in period t, and thus,

$$\begin{aligned}
\sum_{t=1}^{T} E\left[Z_{G_t}\right] &\geq \sum_{t=1}^{T} E\left[\mathbf{1}\left(t \in \mathcal{J}_T\right) \cdot E\left[\sum_{i=1}^{g_t} \pi_i \lambda_i p_i \sum_{r=1}^{\lfloor \mu \rfloor} \mathbf{1}\left(B_r^{g_t} \geq c_i, S^t \geq r\right) \;\middle|\; \ell_t, F_t\right]\right] \\
&\geq \rho\left(1 - \frac{1}{k^{(1+2\alpha)/3}}\right) \sum_{t=1}^{T} E\left[\mathbf{1}\left(t \in \mathcal{J}_T\right) \cdot \mu \sum_{i=1}^{g_t} \pi_i \lambda_i p_i\right] \qquad (4)
\end{aligned}$$

where the final inequality follows because $t \in \mathcal{J}_T$ implies that $g_t = \ell_t \leq \mathcal{I}_U$, and therefore,

$$\begin{aligned}
E\left[\sum_{i=1}^{g_t} \pi_i \lambda_i p_i \sum_{r=1}^{\lfloor \mu \rfloor} \mathbf{1}\left(B_r^{g_t} \geq c_i, S^t \geq r\right) \;\middle|\; \ell_t, F_t\right] &= \sum_{i=1}^{\ell_t} \pi_i \lambda_i p_i \sum_{r=1}^{\lfloor \mu \rfloor} \mathcal{P}\left\{B_r^{\ell_t} \geq c_i, S^t \geq r \;\middle|\; \ell_t, F_t\right\} \\
&\geq \left(1 - \frac{1}{k^{(1+2\alpha)/3}}\right) \sum_{i=1}^{\ell_t} \pi_i \lambda_i p_i \sum_{r=1}^{\lfloor \mu \rfloor} \mathcal{P}\left\{S^t \geq r \middle| \ell_t, F_t\right\} \\
&= \rho\left(1 - \frac{1}{k^{(1+2\alpha)/3}}\right) \mu \sum_{i=1}^{g_t} \pi_i \lambda_i p_i,
\end{aligned}$$

where the inequality follows from Lemma 10 and the fact that, conditioned on ℓ_t and $S^t \geq r$, $B_r^{\ell_t}$ is independent of F_t. The final equality follows from the fact that S^t is independent of ℓ_t and F_t and $\sum_{r=1}^{\lfloor \mu \rfloor} \mathcal{P}\left\{S^t \geq r\right\} = E\left[\min\left\{S^t, \mu\right\}\right] = \rho\mu$.

Note that when $\ell_t \leq \mathcal{I}_U$, it follows from the definition of $\mathcal{I}_U$ that $\mu \sum_{i=1}^{g_t} c_i \lambda_i p_i = \mu \sum_{i=1}^{\ell_t} c_i \lambda_i p_i \leq 1 - \frac{1}{k} - \frac{1}{k^{(1-\alpha)/3}}$, and we have that $\sum_{i=1}^{\ell_t} \pi_i \lambda_i p_i \geq Z^* \sum_{i=1}^{\ell_t} c_i \lambda_i p_i$ by Lemma 9. Thus, it follows from Equation (4) that

$$\begin{aligned}
\sum_{t=1}^{T} E\left[Z_{G_t}\right] &\geq \rho\left(1 - \frac{1}{k^{(1+2\alpha)/3}}\right) Z^* \sum_{t=1}^{T} E\left[\mathbf{1}\left(t \in \mathcal{J}_T\right) \cdot \mu \sum_{i=1}^{g_t} c_i \lambda_i p_i\right] \\
&= \rho\left(1 - \frac{1}{k^{(1+2\alpha)/3}}\right) Z^* \sum_{t=1}^{T} E\left[\mathbf{1}\left(t \in \mathcal{J}_T\right)\left\{\mu \sum_{i=1}^{g_t} c_i \lambda_i \hat{p}_i^{t-1} + \mu \sum_{i=1}^{g_t} c_i \lambda_i \left(p_i - \hat{p}_i^{t-1}\right)\right\}\right] \\
&\geq \rho\left(1 - \frac{1}{k^{(1+2\alpha)/3}}\right) Z^* \sum_{t=1}^{T} E\left[\mathbf{1}\left(t \in \mathcal{J}_T\right)\left\{C^* + \mu \sum_{i=1}^{g_t} c_i \lambda_i \left(p_i - \hat{p}_i^{t-1}\right)\right\}\right], \qquad (5)
\end{aligned}$$

where $C^* \equiv 1 - 1/k - 2/k^{(1-\alpha)/3} - 1/k^{1-\alpha}$. The final inequality follows from the definition of ℓ_t, which implies that $\mu \sum_{i=1}^{\ell_t} c_i \lambda_i \hat{p}_i^{t-1} \leq 1 - \frac{1}{k} - \frac{2}{k^{(1-\alpha)/3}} < \mu \sum_{i=1}^{\ell_t+1} c_i \lambda_i \hat{p}_i^{t-1}$. Thus, when $\ell_t \leq \mathcal{I}_U$, it follows from Assumption 1(c) that $c_i \lambda_i \mu \leq 1/k^{1-\alpha}$ for all $i \leq \ell_t + 1$, and we must have $\mu \sum_{i=1}^{\ell_t} c_i \lambda_i \hat{p}_i^{t-1} \geq C^*$.

Since F_t and ℓ_t are independent, it follows from the definition of F_t and Lemma 12 that for any t, $E\left[\mathbf{1}\left(t \in \mathcal{J}_T\right)\right] \geq \left(1 - \frac{1}{k^{(1-\alpha)/3}}\right)\left(1 - \gamma_t\right)$. Moreover, by interchanging the order of the sum, we have

$$\begin{aligned}
E\left[\sum_{t=1}^{T} \mathbf{1}\left(t \in \mathcal{J}_T\right) \sum_{i=1}^{g_t} c_i \lambda_i \mu \left(p_i - \hat{p}_i^{t-1}\right)\right] &= E\left[\sum_{i=1}^{\max\{g_t \,:\, t \in \mathcal{J}_T\}} \sum_{t \in \mathcal{J}_T : g_t \geq i} c_i \lambda_i \mu \left(p_i - \hat{p}_i^{t-1}\right)\right] \\
&\geq -E\left[\sum_{i=1}^{\max\{g_t \,:\, t \in \mathcal{J}_T\}} \sum_{t \in \mathcal{J}_T : g_t \geq i} c_i \lambda_i \mu \left|p_i - \hat{p}_i^{t-1}\right|\right] \geq -\mu \sum_{i=1}^{\mathcal{I}_U} c_i \lambda_i \mathcal{E}_{i,T},
\end{aligned}$$

where the final inequality follows from the definition of $\mathcal{E}_{i,T}$ and the fact that for any $t \in \mathcal{J}_T$, $g_t = \ell_t \leq \mathcal{I}_U$. Putting the above results with Equation (5), we obtain

$$\begin{aligned} \frac{\sum_{t=1}^T E\left[Z_{G_t}\right]}{T Z^*} &\geq \rho C^* \left(1 - \frac{1}{k^{(1+2\alpha)/3}}\right)\left(1 - \frac{1}{k^{(1-\alpha)/3}}\right) \frac{1}{T}\sum_{t=1}^T (1-\gamma_t) - \frac{\rho}{T}\left(1 - \frac{1}{k^{(1+2\alpha)/3}}\right)\mu \sum_{i=1}^{\mathcal{I}_U} c_i \lambda_i \mathcal{E}_{i,T} \\ &\geq \frac{\rho}{T}\left(1 - \frac{1}{k^{(1+2\alpha)/3}} - \frac{3}{k^{(1-\alpha)/3}}\right) - \frac{\sum_{t=1}^T \gamma_t}{T} - \frac{\rho}{T}\left(1 - \frac{1}{k^{(1+2\alpha)/3}}\right)\mu \sum_{i=1}^{\mathcal{I}_U} c_i \lambda_i \mathcal{E}_{i,T} \end{aligned}$$

which is the desired result. Note that the final inequality follows from the observation that $C^*\left(1 - \frac{1}{k^{(1+2\alpha)/3}}\right)\left(1 - \frac{1}{k^{(1-\alpha)/3}}\right) \geq 1 - \frac{1}{k^{(1+2\alpha)/3}} - \frac{3}{k^{(1-\alpha)/3}}$ whose proof follows from standard algebra and we omit the details. ■

B.3 Proof of Lemma 14

The proof of Lemma 14 makes use of the following results. The first result provides an upper bound on the sample average for a Bernoulli random variable. Since the result follows from a standard application of Chernoff bound (see, for example, [22]), and we omit the proof.

Lemma 15 *Let $X_1, \ldots, X_n$ be independent and identically distributed Bernoulli random variables with $\mathcal{P}\{X_i = 1\} = p$. Then, for any $0 < \epsilon < 1$, $E\left|p - \frac{1}{n}\sum_{i=1}^n X_i\right| \leq \epsilon + 2e^{-n\epsilon^2/3}$.*

The next result is a direct corollary of Berstein's Inequality for real-valued random variables (see, for example, Appendix B in [26]) and we omit the details.

Theorem 4 *Let $Y_1, \ldots, Y_n$ be independent nonnegative random variables and $0 \leq Y_i \leq M$ for all i. Let $Y = \sum_{i=1}^n Y_i$. Then, for any $0 < \delta < 1$ and $0 \leq \eta \leq E[Y]$,*

$$\mathcal{P}\left\{Y \leq (1-\delta)\eta\right\} \leq \exp\left\{\frac{-\frac{1}{2}\delta^2\eta^2}{Var(Y) + \frac{1}{3}M\delta\eta}\right\}$$

The next lemma establishes an upper bound on the cumulative error $\mathcal{E}_{i,T}$ of keyword i.

Lemma 16 *For any keyword i, $0 < \epsilon, \delta < 1$, and $T \geq 1$,*

$$\mathcal{E}_{i,T} \leq \epsilon T + \frac{1}{1 - \exp\left\{-\delta^2\lambda_i\rho^2\left(1 - \frac{1}{k^{(1+2\alpha)/3}}\right)^2 \Big/ 4\right\}} + \frac{2}{1 - \exp\left\{-\epsilon^2(1-\delta)\lambda_i\mu\rho\left(1 - \frac{1}{k^{(1+2\alpha)/3}}\right) \Big/ 3\right\}}$$

Proof: Let $0 < \epsilon, \delta < 1$ be given. Recall that $\mathcal{E}_{i,T} = E\left[\sum_{t \in \mathcal{J}_T : g_t \geq i} \left|p_i - \hat{p}^{t-1}\right|\right]$ where $\mathcal{J}_T = \{t \leq T : F_t = 1 \,\text{and}\, \ell_t \leq \mathcal{I}_U\}$. Note that when $F_t = 1$, we have that $g_t = \ell_t$. Therefore, it suffices to consider only keywords i such that $i \leq \mathcal{I}_U$. So, let any $i \leq \mathcal{I}_U$ be given. To prove the desired result, it suffices to show that for *any arbitrary* set of periods $1 \leq t_1 < t_2 < \cdots < t_w \leq T$ such that $t_h \in \mathcal{J}_T$ and $g_{t_h} \geq i$ for all $1 \leq h \leq w$, the conditional expectation

$$E\left[\sum_{t \in \mathcal{J}_T : g_t \geq i} \left|p_i - \hat{p}_i^{t-1}\right| \;\middle|\; (t_1, \ell_{t_1}, F_{t_1}), \ldots, (t_w, \ell_{t_w}, F_{t_w}),\right]$$

is bounded above by the expression in Lemma 16. For ease of exposition, we will drop the reference to the conditioning information $\mathcal{F}_T \equiv \{(t_1, \ell_{t_1}, F_{t_1}), \ldots, (t_w, \ell_{t_w}, F_{t_w})\}$ when it is clear from the context. Note that during time periods $t_1, \ldots, t_w$, we selected keywords based on the prefix ordering ($F_{t_h} = 1$ and $g_{t_h} = \ell_{t_h}$ for all h) whose largest index g_{t_h} includes i and is less than or equal to $\mathcal{I}_U$.

To facilitate our exposition, let $\xi = \rho\left(1 - \frac{1}{k^{(1+2\alpha)/3}}\right)$. Recall that $Y_i(t)$ denotes the total cumulative impressions that the ad associated with keyword i has received at the end of period t. Conditioned on $(t_1, \ell_{t_1}, F_{t_1}), \ldots, (t_w, \ell_{t_w}, F_{t_w})$, we have that

$$\begin{aligned}
&E\left[\sum_{t \in \mathcal{J}_T : g_t \geq i} \left|p_i - \hat{p}_i^{t-1}\right| \;\middle|\; (t_1, \ell_{t_1}, F_{t_1}), \ldots, (t_w, \ell_{t_w}, F_{t_w})\right] \\
&= \sum_{h=1}^{w} E\left|\hat{p}_i^{t_h-1} - p_i\right| \\
&= \sum_{h=1}^{w} E\left[\left|\hat{p}_i^{t_h-1} - p_i\right| \cdot \mathbf{1}\left(Y_i(t_h - 1) \leq (1-\delta)(h-1)\lambda_i \mu \xi\right)\right] \\
&\quad + \sum_{h=1}^{w} E\left[\left|\hat{p}_i^{t_h-1} - p_i\right| \cdot \mathbf{1}\left(Y_i(t_h - 1) > (1-\delta)(h-1)\lambda_i \mu \xi\right)\right] \\
&\leq \sum_{h=1}^{w} \mathcal{P}\left\{Y_i(t_h - 1) \leq (1-\delta)(h-1)\lambda_i \mu \xi\right\} + \sum_{h=1}^{w}\left(\epsilon + 2e^{-\epsilon^2(1-\delta)(h-1)\lambda_i \mu \xi/3}\right) \\
&\leq \sum_{h=1}^{w} \mathcal{P}\left\{Y_i(t_h - 1) \leq (1-\delta)(h-1)\lambda_i \mu \xi\right\} + \epsilon T + \frac{2}{1 - e^{-\epsilon^2(1-\delta)\lambda_i \mu \xi/3}} \qquad (6)
\end{aligned}$$

where the first inequality follows from Lemma 15 and the final inequality follows from the formula for the geometric series.

Thus, it suffices to show that $\sum_{h=1}^{w} \mathcal{P}\left\{Y_i(t_h - 1) \leq (1-\delta)(h-1)\lambda_i \mu \xi\right\} \leq \frac{1}{1 - e^{-\delta^2 \lambda_i \xi^2/4}}$. Recall that $B_r^{G_{t_h}}$ denote the remaining account balance in period t_h just before the arrival of the r^{th} search query (in period t_h), assuming that we bid on keywords in $G_{t_h} = \{1, \ldots, g_{t_h}\}$. And $Q_r^{t_h}$ denotes the keyword associated with the r^{th} search query in period t_h. For any $1 \leq h \leq w$, let U_h be defined by

$$U_h \equiv \sum_{r=1}^{\min\left\{S^{t_h}, \lfloor \mu \rfloor\right\}} \mathbf{1}\left(Q_r^{t_h} = i,\; B_r^{G_{t_h}} \geq c_i\right) = \sum_{r=1}^{\lfloor \mu \rfloor} \mathbf{1}\left(Q_r^{t_h} = i,\; B_r^{G_{t_h}} \geq c_i, S^{t_h} \geq r\right)$$

The random variable U_h provides a lower bound on the number of impressions that keyword i receives in period t_h. Thus, for any $1 \leq h \leq w$, $\sum_{v=1}^{h-1} U_v \leq Y_i(t_h - 1)$.

Note that, *conditioned on* $(t_1, \ell_{t_1}, F_{t_1}), \ldots, (t_w, \ell_{t_w}, F_{t_w})$, the random variables $U_1, U_2, \ldots, U_w$ are independent because S_{t_h} are i.i.d. and the distribution of U_h depends only on G_{t_h} which is fixed given our conditioning information. We can thus apply the result of Theorem 4 to the random variable $\sum_{v=1}^{h-1} U_v$. To do so, we need to determine its expectation and variance. *Conditioned on* $t_1, \ldots, t_w$, we will show that for any $1 \le h \le w$, $E\left[\sum_{v=1}^{h-1} U_v\right] \ge (h-1)\lambda_i \mu \xi$ and $\sum_{v=1}^{h-1} Var\left[U_v\right] \le (h-1)\lambda_i \mu^2$.

To prove this result, note that for any $1 \le h \le w$,

$$E\left[U_h\right] = \sum_{r=1}^{\lfloor \mu \rfloor} \lambda_i \mathcal{P}\left\{B_r^{G_{t_h}} \ge c_i, S^{t_h} \ge r\right\} \ge \lambda_i \left(1 - \frac{1}{k^{(1+2\alpha)/3}}\right) \sum_{r=1}^{\lfloor \mu \rfloor} \mathcal{P}\left\{S^{t_h} \ge r\right\} = \lambda_i \mu \xi,$$

where the inequality follows from Lemma 10 and the fact that $G_{t_h} = \{1, 2, \ldots, g_{t_h}\}$ and $g_{t_h} \le \mathcal{I}_U$ for all $1 \le h \le w$. Note that since $g_{t_h} \le \mathcal{I}_U$ for all h, it follows from the definition of $\mathcal{I}_U$ that $\mu \sum_{u=1}^{g_{t_h}} c_u \lambda_u p_u \le 1 - \frac{1}{k} - \frac{1}{k^{(1-\alpha)/3}}$, which is the required hypothesis for the application of Lemma 10. The final equality follows because $\sum_{r=1}^{\lfloor \mu \rfloor} \mathcal{P}\left\{S^{t_h} \ge r\right\} = E\left[\min\left\{S^{t_h}, \mu\right\}\right] = \rho\mu$ by definition of ρ. Thus,for any $1 \le h \le w$, $E\left[\sum_{v=1}^{h-1} U_v\right] \ge (h-1)\lambda_i \mu \xi$.

We will now establish an upper bound on the variance. *Conditioned on* $(t_1, \ell_{t_1}, F_{t_1}), \ldots, (t_w, \ell_{t_w}, F_{t_w})$, for any $1 \le h \le w$,

$$\begin{aligned} Var\left[U_h\right] &\le E\left[U_h^2\right] \le E\left[\left(\sum_{r=1}^{\lfloor \mu \rfloor} \mathbf{1}\left(Q_r^{t_h} = i\right)\right)^2\right] = E\left[\sum_{r=1}^{\lfloor \mu \rfloor} \mathbf{1}\left(Q_r^{t_h} = i\right) + \sum_{q \ne r} \mathbf{1}\left(Q_r^{t_h} = i, Q_q^{t_h} = i\right)\right] \\ &= \lambda_i \lfloor \mu \rfloor + \lambda_i^2 \lfloor \mu \rfloor (\lfloor \mu \rfloor - 1) \le \lambda_i \mu^2 \end{aligned}$$

where the last equality follows from the fact that, for $r \ne q$, the arrival of the r^{th} and q^{th} queries are independent of each other. The final inequality follows from our assumption that $\mu > 1$. Thus, $\sum_{v=1}^{h-1} Var\left[U_v\right] \le (h-1)\lambda_i \mu^2$.

Note that $0 \le U_h \le \mu$ for all h and i. By Theorem 4, we can conclude that for any $1 \le h \le w$,

$$\begin{aligned} \mathcal{P}\left\{Y_i\left(t_h - 1\right) \le (1-\delta)(h-1)\lambda_i \mu \xi\right\} &\le \mathcal{P}\left\{\sum_{v=1}^{h-1} U_v \le (1-\delta)(h-1)\lambda_i \mu \xi\right\} \\ &\le \exp\left\{\frac{-\frac{1}{2}\left(\delta(h-1)\lambda_i \mu \xi\right)^2}{(h-1)\lambda_i \mu^2 + \frac{1}{3}\mu\delta\left((h-1)\lambda_i \mu \xi\right)}\right\} \\ &= \exp\left\{\frac{-\frac{1}{2}\delta^2 (h-1)\lambda_i \xi^2}{1 + \frac{1}{3}\delta\xi}\right\} \le e^{-\delta^2 (h-1)\lambda_i \xi^2 / 4} \end{aligned}$$

where the last inequality follows from the fact that $1 + \frac{1}{3}\delta\xi \le 2$. Hence,

$$\sum_{h=1}^{w} \mathcal{P}\left\{Y_i\left(t_{ih} - 1\right) \le (1-\delta)(h-1)\lambda_i \mu \xi\right\} \le \sum_{h=1}^{w} e^{-\delta^2 (h-1)\lambda_i \xi^2 / 4} \le \frac{1}{1 - e^{-\delta^2 \lambda_i \xi^2 / 4}}.$$

Since $(t_1, \ell_{t_1}, F_{t_1})$, $\ldots$, $(t_w, \ell_{t_w}, F_{t_w})$ are arbitrary, the desired result follows. ■

Finally, here is the proof of Lemma 14.

Proof: Let $\xi = \rho\left(1 - \frac{1}{k^{(1+2\alpha)/3}}\right)$. It follows from Lemma 16 that for any $0 < \epsilon, \delta < 1$,

$$\begin{aligned}\mu \sum_{i=1}^{\mathcal{I}_U} c_i \lambda_i \mathcal{E}_{i,T} &\leq \epsilon \mu T \sum_{i=1}^{\mathcal{I}_U} c_i \lambda_i + \sum_{i=1}^{\mathcal{I}_U} \frac{c_i \lambda_i \mu}{1 - e^{-\delta^2 \lambda_i \xi^2/4}} + \sum_{i=1}^{\mathcal{I}_U} \frac{2 c_i \lambda_i \mu}{1 - e^{-\epsilon^2 (1-\delta) \lambda_i \mu \xi/3}} \\ &\leq \epsilon \mu T \sum_{i=1}^{\mathcal{I}_U} c_i \lambda_i + \frac{8\mu}{\delta^2 \xi^2} \sum_{i=1}^{\mathcal{I}_U} c_i + \frac{6}{\epsilon^2 (1-\delta) \xi} \sum_{i=1}^{\mathcal{I}_U} \frac{c_i \lambda_i \mu}{1 - e^{-\lambda_i \mu}},\end{aligned}$$

where the last inequality follows from the fact that for any $0 \leq x \leq 1$ and any $d \geq 0$, $1/\left(1 - e^{-x}\right) \leq 2/x$ and $1/(1 - e^{-dx}) \leq 1/\left(x(1 - e^{-d})\right)$, which implies that $1/\left(1 - e^{-\delta^2 \lambda_i \xi^2/4}\right) \leq 8/\left(\delta^2 \lambda_i \xi^2\right)$ and $1/\left(1 - e^{-\epsilon^2(1-\delta)\lambda_i \mu \xi/3}\right) \leq 3/\left(\epsilon^2 (1-\delta) \xi \left(1 - e^{-\lambda_i \mu}\right)\right)$.

By setting $\delta = 1/2$, using the fact that $c_i \lambda_i \mu \leq 1/k^{1-\alpha}$ (Assumption 1), and observing that $\lambda_i \mu / \left(1 - e^{-\lambda_i \mu}\right) \leq \lambda^* \mu / \left(1 - e^{-\lambda^* \mu}\right)$ where $\lambda^* = \max_{1 \leq i \leq \mathcal{I}_U} \lambda_i$, we conclude that for any $0 < \epsilon < 1$,

$$\mu \sum_{i=1}^{\mathcal{I}_U} c_i \lambda_i \mathcal{E}_{i,T} \leq \frac{\epsilon T \mathcal{I}_U}{k^{1-\alpha}} + \frac{32 \mu \mathcal{I}_U}{k \xi^2} + \frac{12 \mathcal{I}_U \lambda^* \mu / \left(1 - e^{-\lambda^* \mu}\right)}{\epsilon^2 k \xi},$$

and therefore,

$$\begin{aligned}\frac{\xi}{T} \mu \sum_{i=1}^{\mathcal{I}_U} c_i \lambda_i \mathcal{E}_{i,T} &\leq \frac{\epsilon \xi \mathcal{I}_U}{k^{1-\alpha}} + \frac{32 \mu \mathcal{I}_U}{k \xi T} + \frac{12 \mathcal{I}_U \lambda^* \mu / \left(1 - e^{-\lambda^* \mu}\right)}{\epsilon^2 k T} \\ &\leq \frac{\epsilon \rho \mathcal{I}_U}{k^{1-\alpha}} + \frac{32 L}{\left(\min_{1 \leq i \leq \mathcal{I}_U} \pi_i \lambda_i p_i\right) k \xi^2 T} + \frac{12 \mathcal{I}_U \lambda^* \mu / \left(1 - e^{-\lambda^* \mu}\right)}{\epsilon^2 k T} \\ &\leq \frac{\epsilon \rho \mathcal{I}_U}{k^{1-\alpha}} + \frac{\left(128/\rho^2\right) L}{\left(\min_{1 \leq i \leq \mathcal{I}_U} \pi_i \lambda_i p_i\right) k T} + \frac{12 \mathcal{I}_U \lambda^* \mu / \left(1 - e^{-\lambda^* \mu}\right)}{\epsilon^2 k T},\end{aligned}$$

which is the desired result. Note that the second inequality follows because $\xi \leq \rho$ and because L represents the maximum profit that can be earned in any one period, and thus,

$$L \geq E\left[Z_{1,\ldots,\mathcal{I}_U}\right] \geq \rho\left(1 - \frac{1}{k^{(1+2\alpha)/3}}\right) \mu \sum_{i=1}^{\mathcal{I}_U} \pi_i \lambda_i p_i \geq \xi \left(\min_{1 \leq i \leq \mathcal{I}_U} \pi_i \lambda_i p_i\right) \mu \mathcal{I}_U,$$

where the second inequality follows from the proof of Theorem 1. Also, by the hypothesis of Theorem 2, $2/k^{1/3} \leq 2/k^{(1-\alpha)/3} \leq 1$, which implies that $k \geq 8$, and therefore, $1 - 1/k^{(1+2\alpha)/3} \geq 1/2$, which implies that $\xi^2 \geq \rho^2/4$. ∎

C Proof of Theorem 3

Restatement of Theorem 3: *Suppose that each keyword $i_1, \ldots, i_Q$ receives at least T_0 clicks during Phase 1 of the* Adaptive Bidding With Unknown Profits. *Then, under Assumption 2,*

- *With probability at least* $1 - 1/T$, *the estimated-profit-to-cost ratio* $\widehat{\Pi}_i / c_i$ *satisfies:*

$$\frac{\widehat{\Pi}_{\tau_1}}{c_{\tau_1}} \geq \frac{\widehat{\Pi}_{\tau_2}}{c_{\tau_2}} \geq \cdots \geq \frac{\widehat{\Pi}_{\tau_{\mathcal{I}^*_\tau}}}{c_{\tau_{\mathcal{I}^*_\tau}}} \geq \max\left\{ \frac{\widehat{\Pi}_{\tau_\ell}}{c_{\tau_\ell}} : \ell \geq \mathcal{I}^*_\tau + 1 \right\}$$

- *Let* $\xi = \rho\left(1 - 1/k^{(1+2\alpha)/3} - 3/k^{(1-\alpha)/3}\right)$. *For any* $0 < \epsilon < 1$,

$$\frac{E\left[\sum_{t=1}^T Z_{G_t}\right]}{Z^* T} \quad \geq \quad \left(1 - \frac{1}{T}\right)\left(1 - \frac{QT_0}{T}\right)\left(\xi - \frac{\sum_{t=(QT_0)+1}^T \gamma_t}{T - QT_0} - M\left(\epsilon, T - QT_0\right)\right),$$

where $M(\epsilon, T)$ *is the* same *as the one given in Theorem 2 for the* Adaptive Bidding *algorithm.*

The proof of Theorem 3 makes use of the following results.

Lemma 17 *Under the assumption of Theorem 3, the random variables* $\widehat{\Pi}_1, \ldots, \widehat{\Pi}_N$ *are jointly Gaussian with mean* $\bar{\pi}_1, \ldots, \bar{\pi}_N$, *and for any* i, $Var\left(\widehat{\Pi}_i\right) \leq \left(\Delta^2 \cdot \underline{c}^2\right) / \left(8 \ln\left(2NT\right)\right)$.

Proof: For any $1 \leq \ell \leq Q$, let N_{i_ℓ} denote the number of times that the ad associated with keyword i_ℓ has been clicked during Phase 1 of the algorithm. By Assumption 2, the random variable $\widehat{\Pi}_{i_\ell}$ is an average of N_{i_ℓ} independent Gaussian random variables, each of which has mean $\bar{\pi}_{i_\ell}$ and variance $\sigma^2_{i_\ell}$. Therefore, $\widehat{\Pi}_{i_\ell}$ is a Gaussian random variable with mean $\bar{\pi}_{i_\ell}$ and variance $\sigma^2_{i_\ell}/N_{i_\ell}$. Moreover, the random variables $\widehat{\Pi}_{i_1}, \ldots, \widehat{\Pi}_{i_Q}$ are independent and $\left(\widehat{\Pi}_1, \ldots, \widehat{\Pi}_N\right)' = UA_Q^{-1}\left(\widehat{\Pi}_{i_1}, \ldots, \widehat{\Pi}_{i_Q}\right)' \sim \mathcal{N}\left(\left(\bar{\pi}_1, \ldots, \bar{\pi}_N\right)', MDM'\right)$, where $M = UA_Q^{-1}$ and D is a diagonal matrix whose elements are $\left(\sigma^2_{i_1}/N_{i_1}\right), \ldots, \left(\sigma^2_{i_Q}/N_{i_Q}\right)$, respectively. Since $N_{i_\ell} \geq T_0$ for all ℓ, it follows that

$$Var\left(\widehat{\Pi}_i\right) \quad = \quad \left(MDM'\right)_{ii} = \sum_{\ell=1}^{Q} M^2_{i\ell} D_{\ell\ell} = \sum_{\ell=1}^{Q} M^2_{i\ell} \frac{\sigma^2_{i_\ell}}{N_{i_\ell}} \leq \frac{1}{T_0} \sum_{\ell=1}^{Q} M^2_{i\ell} \sigma^2_{i_\ell} \leq \frac{\Delta^2 \cdot \underline{c}^2}{8 \ln\left(2NT\right)},$$

where the final inequality follows from the definition T_0. ■

Here is a proof of Theorem 3.

Proof: Recall that $(\tau_1, \ldots, \tau_N)$ corresponds to the true (yet unknown) ordering of the expected-profit-to-cost. Let $\mathcal{C}$ denote the even that the estimated-profit-to-cost ratio satisfies the following relationship: $\frac{\widehat{\Pi}_{\tau_1}}{c_{\tau_1}} \geq \frac{\widehat{\Pi}_{\tau_2}}{c_{\tau_2}} \geq \cdots \geq \frac{\widehat{\Pi}_{\tau_{\mathcal{I}^*_\tau}}}{c_{\tau_{\mathcal{I}^*_\tau}}} \geq \max\left\{ \frac{\widehat{\Pi}_{\tau_\ell}}{c_{\tau_\ell}} : \ell \geq \mathcal{I}^*_\tau + 1 \right\}$. By definition, we have that

$$\begin{aligned}
\mathcal{P}\left\{\mathcal{C}\right\} \quad \geq \quad & 1 - \mathcal{P}\left\{ \frac{\widehat{\Pi}_{\tau_i}}{c_{\tau_i}} < \frac{\widehat{\Pi}_{\tau_{i+1}}}{c_{\tau_{i+1}}} \text{ for some } 1 \leq i < \mathcal{I}^*_\tau \text{ such that } \frac{\bar{\pi}_{\tau_i}}{c_{\tau_i}} > \frac{\bar{\pi}_{\tau_{i+1}}}{c_{\tau_{i+1}}} \right\} \\
& - \mathcal{P}\left\{ \frac{\widehat{\Pi}_{\tau_{\mathcal{I}^*_\tau}}}{c_{\tau_{\mathcal{I}^*_\tau}}} < \frac{\widehat{\Pi}_{\tau_\ell}}{c_{\tau_\ell}} \text{ for some } \ell \geq \mathcal{I}^*_\tau + 1 \text{ such that } \frac{\Pi_{\tau_{\mathcal{I}^*_\tau}}}{c_{\tau_{\mathcal{I}^*_\tau}}} > \frac{\widehat{\Pi}_{\tau_\ell}}{c_{\tau_\ell}} \right\} \\
\geq \quad & 1 - \sum_{i < \mathcal{I}^*_\tau} \mathbf{1}\left[\frac{\bar{\pi}_{\tau_i}}{c_{\tau_i}} > \frac{\bar{\pi}_{\tau_{i+1}}}{c_{\tau_{i+1}}} \right] \mathcal{P}\left\{ \frac{\widehat{\Pi}_{\tau_i}}{c_{\tau_i}} < \frac{\widehat{\Pi}_{\tau_{i+1}}}{c_{\tau_{i+1}}} \right\} - \sum_{\ell > \mathcal{I}^*_\tau} \mathbf{1}\left[\frac{\Pi_{\tau_{\mathcal{I}^*_\tau}}}{c_{\tau_{\mathcal{I}^*_\tau}}} > \frac{\widehat{\Pi}_{\tau_\ell}}{c_{\tau_\ell}} \right] \mathcal{P}\left\{ \frac{\widehat{\Pi}_{\tau_{\mathcal{I}^*_\tau}}}{c_{\tau_{\mathcal{I}^*_\tau}}} < \frac{\widehat{\Pi}_{\tau_\ell}}{c_{\tau_\ell}} \right\} \quad (7)
\end{aligned}$$

For any $1 \leq \ell \leq N$, let $W_\ell = \left(\widehat{\Pi}_\ell - \bar{\pi}_\ell\right)/c_\ell$. For any $1 \leq i, j \leq N$ such that $\bar{\pi}_i/c_i > \bar{\pi}_j/c_j$, the event that $\widehat{\Pi}_i/c_i < \widehat{\Pi}_j/c_j$ is equivalent to the event that $W_i - W_j < -\left(\frac{\bar{\pi}_i}{c_i} - \frac{\bar{\pi}_j}{c_j}\right)$. By Lemma 17, the random variables $\widehat{\Pi}_i$ and $\widehat{\Pi}_j$ are jointly Gaussian with means $\bar{\pi}_i$ and $\bar{\pi}_j$, respectively. Therefore, the random variable $W_i - W_j$ is Gaussian with mean zero, and by Cauchy-Schwartz's Inequality, we have

$$Var\left(W_i - W_j\right) \;\leq\; \left(\sqrt{Var\left(W_i\right)} + \sqrt{Var\left(W_j\right)}\right)^2 \leq \left(\frac{2}{\underline{c}} \cdot \sqrt{\frac{\Delta^2 \cdot \underline{c}^2}{8\ln\left(2NT\right)}}\right)^2 = \frac{\Delta^2}{2\ln\left(2NT\right)},$$

where the last inequality follows from Lemma 17. It is a well known result that if Z is a standard normal random variable with mean zero and variance one, then for any $x > 0$, $\mathcal{P}\left\{Z > x\right\} \leq 2e^{-x^2/2}$, which implies that for $i < \mathcal{I}_\tau^*$ such that $\frac{\bar{\pi}_{\tau_i}}{c_{\tau_i}} > \frac{\bar{\pi}_{\tau_{i+1}}}{c_{\tau_{i+1}}}$,

$$\begin{aligned}\mathcal{P}\left\{\frac{\widehat{\Pi}_{\tau_i}}{c_{\tau_i}} < \frac{\widehat{\Pi}_{\tau_{i+1}}}{c_{\tau_{i+1}}}\right\} \;&=\; \mathcal{P}\left\{W_{\tau_i} - W_{\tau_{i+1}} < -\left(\bar{\pi}_{\tau_i}/c_{\tau_i} - \bar{\pi}_{\tau_{i+1}}/c_{\tau_{i+1}}\right)\right\} \leq 2\exp\left\{-\frac{\left(\bar{\pi}_{\tau_i}/c_{\tau_i} - \bar{\pi}_{\tau_{i+1}}/c_{\tau_{i+1}}\right)^2}{2 \cdot Var\left(W_{\tau_i} - W_{\tau_{i+1}}\right)}\right\} \\ &\leq\; 2\exp\left\{-\frac{\left(\bar{\pi}_{\tau_i}/c_{\tau_i} - \bar{\pi}_{\tau_{i+1}}/c_{\tau_{i+1}}\right)^2 \cdot 2\ln(2NT)}{2\Delta^2}\right\} \leq 2\exp\left\{-\ln(2NT)\right\} = \frac{1}{NT},\end{aligned}$$

where the second inequality follows from the upper bound on the variance and the final inequality follows from the definition of Δ, which implies that $\Delta^2 \leq \left(\frac{\bar{\pi}_{\tau_i}}{c_{\tau_i}} - \frac{\bar{\pi}_{\tau_{i+1}}}{c_{\tau_{i+1}}}\right)^2$. A similar bound can be established for $\mathbf{1}\left[\Pi_{\tau_{\mathcal{I}_\tau^*}}/c_{\tau_{\mathcal{I}_\tau^*}} > \widehat{\Pi}_{\tau_\ell}/c_{\tau_\ell}\right]\mathcal{P}\left\{\widehat{\Pi}_{\tau_{\mathcal{I}_\tau^*}}/c_{\tau_{\mathcal{I}_\tau^*}} < \widehat{\Pi}_{\tau_\ell}/c_{\tau_\ell}\right\}$. Therefore, it follows from Equation (7) that $\mathcal{P}\left\{\mathcal{C}\right\} \geq 1 - \sum_{i=1}^N 1/(NT) = 1 - 1/T$, which establishes the first part of Theorem 3.

We will now prove the second part of Theorem 3. From the definition,

$$E\left[\sum_{t=1}^T Z_{G_t}\right] \;\geq\; E\left[\sum_{t=(QT_0)+1}^T Z_{G_t}\right] \geq \mathcal{P}\left\{\mathcal{C}\right\} E\left[\sum_{t=(QT_0)+1}^T Z_{G_t} \;\Big|\; \mathcal{C}\right] \geq \left(1 - \frac{1}{T}\right) E\left[\sum_{t=(QT_0)+1}^T Z_{G_t} \;\Big|\; \mathcal{C}\right],$$

where the last inequality follows from the first part of Theorem 3. Under the event $\mathcal{C}$, the ordering based on the estimated-profit-to-cost ratio $\widehat{\Pi}_\ell/c_\ell$ satisfies Assumption 1(b), then it then follows from the performance guarantee of the Adaptive Bidding algorithm (Theorem 2) that for any $0 < \epsilon < 1$,

$$E\left[\sum_{t=(QT_0)+1}^T Z_{G_t} \;\Big|\; \mathcal{C}\right] \geq Z^*\left(T - QT_0\right)\left(\xi - \frac{\sum_{t=(QT_0)+1}^T \gamma_t}{T - QT_0} - M\left(\epsilon, T - QT_0\right)\right),$$

where $\xi = \rho\left(1 - \frac{1}{k^{(1+2\alpha)/3}} - \frac{3}{k^{(1-\alpha)/3}}\right)$. Therefore,

$$\frac{E\left[\sum_{t=1}^T Z_{G_t}\right]}{Z^* T} \;\geq\; \left(1 - \frac{1}{T}\right)\left(1 - \frac{QT_0}{T}\right)\left(\xi - \frac{\sum_{t=(QT_0)+1}^T \gamma_t}{T - QT_0} - M\left(\epsilon, T - QT_0\right)\right),$$

which is the desired result. ■