Optimizing Your Online-Advertisement Asynchronously

Longbo Huang
longbohuang@tsinghua.edu.cn
IIIS, Tsinghua University

***Abstract*— We consider the problem of designing optimal online-ad investment strategies for a single advertiser, who invests at multiple sponsored search sites simultaneously, with the objective of maximizing his average revenue subject to an advertising budget constraint. A greedy online investment scheme is developed to achieve an average revenue that can be pushed to within $O(\epsilon)$ of the optimal, for any $\epsilon > 0$, with a tradeoff that the temporal budget violation is $O(1/\epsilon)$. Different from many existing algorithms, our scheme allows the advertiser to *asynchronously* update his investments on each search engine site, hence applies to systems where the timescales of action update intervals are heterogeneous for different sites. We also quantify the impact of inaccurate estimation of the system dynamics and show that the algorithm is robust against imperfect system knowledge.**

I. Introduction

Over the past years, online advertisement has grown significantly and has become one of the most important advertising approaches. According to Emarketer analysts [1], online advertising spending will reach 46.5 billion dollars in 2013 and is expected to surpass 60 billion dollars in 2016. Among the many online advertising methods, sponsored search has been a very successful mechanism and has served as a major income source for many search engines such as Google and Yahoo!.

Due to the popularity of sponsored search, designing efficient algorithms for it has received much attention, and there has been a large body of previous work in this area. These works can roughly be divided into two categories. The first category of work aims at designing optimal algorithms for maximizing the search engines' revenue. [2] proposes an online ad allocation algorithm based on online matching [3], and proves that it achieves an $1 - 1/e$ competitive ratio. [4] uses a primal-dual approach and designs a scheme that achieves the same competitive ratio. [5] looks at the problem of finding revenue-optimal rules for ranking ads. [6] formulates the problem as a stochastic optimization problem and proposes an algorithm to achieve near-optimal performance. [7] designs algorithms for associating queries with budget constrained advertisers and conducts system implementations. The second category focuses on designing optimal bidding strategies for the advertisers. [8] studies properties of greedy bidding schemes and develops a balanced bidding algorithm. [9] considers a single advertiser trying to maximize the number of clicks given a certain budget and tackles the problem with Markov decision process (MDP). [10] formulates the optimal bidding problem as an MDP with large number of bidders. [11] considers a single advertiser and proposes a greedy bidding scheme based on stochastic approximation. However, we note that the aforementioned works only consider optimizing the performance of the advertiser or the search engine in the ad auction process. Hence, they focus mainly on designing algorithms for systems with a single search engine site. In practice, however, an advertiser can typically utilize multiple sites simultaneously for advertising. In order to maximize his revenue, one must jointly optimize the investments at all sites.

In this paper, we consider a single advertiser who is trying to advertise his product by utilizing a set of sponsored search engines (called sites below), e.g., Google or Yahoo!. At every time, the advertiser decides the advertising configuration at each site, e.g., total budget and maximum pay-per-click, associating ads to different keywords, and the bidding strategies, and decides how much money to allocate to advertising at each search engine. According to the advertiser's customized setting and the sites' ad allocation and billing mechanisms, the amount of money at a search engine site will be consumed at a certain rate, and an amount of revenue will be generated to the advertiser. After the money is completely depleted at a site, the advertiser updates his investment at that site and resets his advertising configurations, based on his investment strategy and his observation of the site's performance. The objective of the advertiser is to find an investment policy to maximize his time average revenue subject to his advertising budget.

This problem captures many features of the online advertising procedure at popular sponsored search engine sites such as Google's AdWords [12] and Bing ads [13]. However, solving this problem is challenging. First of all, the advertiser has to learn over time about how to allocate his investment across different search engine sites subject to his budget constraint. With the many different features and ad mechanisms of the different sites and the many options of the advertiser, this is a nontrivial task. Second, due to the intrinsic stochastic nature of the online advertising process, the advertising performance as well as the time to re-update the advertising configurations at each site are highly dynamic and can be very different from site to site. This imposes the constraint that the investment strategy must be able to handle *asynchronous* system operations. However, most existing optimal control algorithms, e.g., [14] [15], require simultaneous update of the control actions and hence do not apply in this case. Finally, optimizing time average metrics defined over time-varying update intervals typically involves optimizing sum of ratios of functions. This is in general non-convex and problems of this kind are generally hard to solve.

In order to tackle this problem and to resolve the aforementioned difficulties, we adopt the recently developed *Lyapunov Drift-Plus-Penalty-Ratio* technique for renewal systems and asynchronous control [16] [17]. This approach has three important components: (1) a virtual deficit queue with carefully defined arrival and service rates as ratios of the corresponding system metrics, (2) a Lyapunov drift defined over variable size time intervals, and (3) a drift-plus-penalty-ratio principle in decision making. These three components enable the development of a simple yet near-optimal asynchronous control scheme called Ad-Investment (AI), which allows asynchronous updates of the investments and the advertising configurations. We show that the AI algorithm achieves an average revenue that can be pushed to within $O(\epsilon)$ of the maximum, for any $\epsilon > 0$, with a tradeoff of an $O(1/\epsilon)$ temporal budget violation. We further investigate the performance of AI with imperfect system knowledge and quantify its impact on the algorithm performance.

This paper is organized as follows: In Section II, we state our model and problem formulation. In Section III, we derive a characterization of the maximum revenue. Then, we present the dynamic Ad-Investment strategy (AI) for the advertiser in Section IV. We investigate the performance loss of AI due to imperfect knowledge of the system dynamics in Section V. Simulation results are presented in Section VI. We conclude the paper in Section VII.

II. System Model

In this section, we present our system model. We consider a single advertiser who is trying to advertise his products at N sponsored search websites (called sites below), e.g., Google or Bing, denoted by $\mathcal{S} = \{S_1, ..., S_N\}$ (see Fig. 1). We assume that the system operates in continuous time, i.e., $t \in \mathbb{R}_+$.

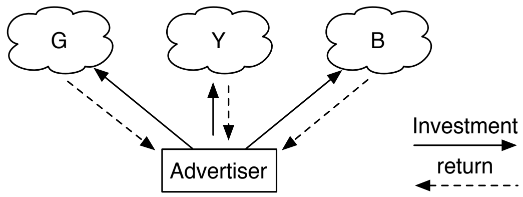

Fig. 1. A single advertiser allocates his advertising budget across multiple search engine sites (“G,” “Y” and “B” in this example). The objective of the advertiser is to achieve maximum revenue subject to his advertising budget.

A. Advertiser Actions

The advertiser first decides how much money to invest at each site and specifies the operation configuration, e.g., the maximum pay-per-click, or which ads to play for the keywords. Then, when the money deposited at a site is depleted, the advertiser updates his investment amount and the operation configuration for that search site, and the advertising process continues. [1]

Note that this operating mode is quite common in today's sponsored search engines, e.g., Google Adwords [12]. Under the advertiser's investment policy, each site goes through a sequence of asynchronous *advertising* (and *freeze*) intervals, defined as the intervals during which the advertiser has nonzero (zero) budget left at that site (Fig. 2 shows an example). In the following, we will conveniently call an advertising interval plus the subsequent freeze interval a *frame*.

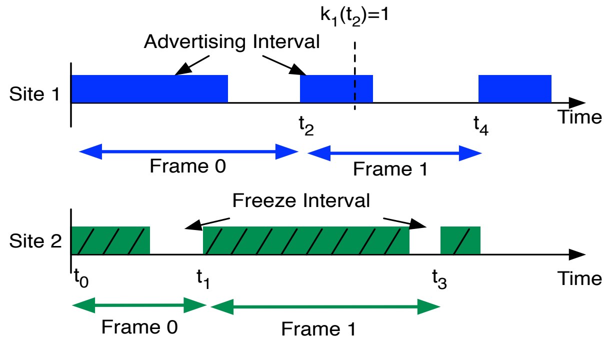

Fig. 2. An example of the advertising and freeze intervals for a 2-site system. The advertiser updates the configurations of the two sites asynchronously at times t_0, t_1, t_2, t_3 and t_4.

We use $p_n[k]$ to denote the amount of new investment the advertiser makes to search site S_n for the k^{th} advertising interval. Note that this new investment will start a new advertising interval. To allow the advertiser to temporarily suspend his investment at a site, we assume that the advertiser can always adopt a “freeze” option. Under this freeze option, the advertiser will not make further new investment into the search site for some time $T_n^{\text{fr}}[k]$ after the k^{th} advertising interval. We call this period the freeze interval. We assume that for all k, the pair $(p_n[k], T_n^{\text{fr}}[k])$ is chosen from the feasible (investment, freeze time) set $\mathcal{P}_n$ for all n. We assume that $\mathcal{P}_n$ includes the option $p_n[k] = 0$, and the constraints $0 \leq p_n[k] \leq p_n^{\max}$ and $0 \leq T_n^{\text{fr}}[k] \leq T_n^{\text{fr},\max}$. One such example can be $\mathcal{P}_n = [0, p] \times [0, T]$. We also use $m_n[k]$ to denote the operation configuration, for instance, the maximum pay-per-click together with the maximum per-day payment, or the targeted user groups, or what ads to display for the corresponding keywords, or the bidding strategy to use at that site for that frame. We assume that $m_n[k] \in \mathcal{M}_n$, where $\mathcal{M}_n$ is the set of feasible operation configurations at site S_n. We assume that both $\mathcal{P}_n$ and $\mathcal{M}_n$ are compact sets.

Given the investment amount and the operating configuration, the corresponding search sites will use their own pricing and ad allocation mechanisms to allocate advertisement opportunities to the advertisers, e.g., the generalized second price auction [18] or the VCG mechanism [19], and to generate revenue for the advertiser, e.g., by bringing visiting traffic or by directly bringing product orders to the advertiser, until the advertiser's new investment is fully depleted. Due to the dynamic nature of the advertising process, the outcome at each site will inevitably be time-varying, e.g., the volume of the visiting traffic to the search site and the set of keywords that are queried. Such a dynamic

[1] Note that our approach can also handle the more general case where the advertiser uses other criteria for deciding when to update the investment amounts and the advertising configurations.

condition affects the site's advertising performance, resulting in different amounts of profit to the advertiser and time-varying speeds in depleting the remaining advertisement deposit at the site. To capture this dynamics, we denote the duration taken to deplete the k^{th} investment at site S_n by $T_n^{\text{ad}}[k]$, and assume that $T_n^{\text{ad}}[k]$ is an independent random variable conditioning on the advertiser's investment $p_n[k]$ and the operating configuration $m_n[k]$. We further assume that there exists a function $F_n[k] \triangleq F_n(p_n[k], m_n[k])$, which maps $(p_n[k], m_n[k])$ to the expected length of $T_n^{\text{ad}}[k]$, i.e.,

$$F_n : (p_n[k], m_n[k]) \mapsto \mathbb{E}\{T_n^{\text{ad}}[k]\}. \quad (1)$$

The function $F_n(\cdot, \cdot)$ is assumed to be known to the advertiser. We assume that under any feasible actions, the frame size at each site S_n is both upper and lower bounded, i.e., [2]

$$0 < T_n^{\min} \leq T_n^{\text{ad}}[k] + T_n^{\text{fr}}[k] \leq T_n^{\max}. \quad (2)$$

We denote $R_n[k]$ the total revenue the advertiser earns during advertising interval k from site S_n. Similar to $T_n^{\text{ad}}[k]$, $R_n[k]$ is also a random variable, which is conditionally independent of others given $p_n[k]$ and $m_n[k]$. We also assume that there exists a function $G_n[k] \triangleq G_n(p_n[k], m_n[k])$ that:

$$G_n : (p_n[k], m_n[k]) \mapsto \mathbb{E}\{R_n[k]\}. \quad (3)$$

The function G_n satisfies $G_n(0, \cdot) = 0$, $G_n(\cdot, \cdot) \geq 0$, $G_n(p, m) \leq \nu p$ for some constant ν, and $G_n(p, m) < G^{\max}$ for all n and all m, p. We also assume that it is known to the advertiser.

In the following of the paper, for notational convenience, we define common upper and lower bounds by dropping the index "n," i.e., $T^{\max} = \max_n T_n^{\max}$, $T^{\min} = \min_n T_n^{\min}$, $R^{\max} = \max_n R_n^{\max}$ and $p^{\max} = \max_n p_n^{\max}$.

B. Budget Constraint and Objective

It can be seen from the above that the advertiser's time average revenue is given by: [3]

$$\text{Profit}_{\text{av}} \triangleq \sum_n \frac{\lim_{K\to\infty} \frac{1}{K} \sum_{k=0}^{K-1} R_n[k]}{\lim_{K\to\infty} \frac{1}{K} \sum_{k=0}^{K-1} (T_n^{\text{ad}}[k] + T_n^{\text{fr}}[k])}. \quad (4)$$

In practice, advertisers typically also have limited advertising budgets. We model this by requiring that the time average advertising expenditure is no more than some pre-specified budget value B_{av}, i.e.,

$$\sum_n \frac{\lim_{K\to\infty} \frac{1}{K} \sum_{k=0}^{K-1} p_n[k]}{\lim_{K\to\infty} \frac{1}{K} \sum_{k=0}^{K-1} (T_n^{\text{ad}}[k] + T_n^{\text{fr}}[k])} \leq B_{\text{av}}. \quad (5)$$

Here the term on the left-hand-side (LHS) is the time average advertising expenditure of the advertiser.

The advertiser's objective is to find an investment strategy that determines how to make the ad-investments, i.e., choosing $p_n[k]$, $T_n^{\text{fr}}[k]$ and $m_n[k]$, so as to maximize his time average revenue (4) subject to the average advertisement budget constraint (5). Below, we call a strategy that chooses $(p_n[k], T_n^{\text{fr}}[k]) \in \mathcal{P}_n$ and $m_n[k] \in \mathcal{M}_n$, and satisfies the budget constraint (5) a *feasible* strategy. Then, we use $\text{Profit}_{\text{av}}^*$ to denote the maximum time average budget-constrained profit achievable over all feasible strategies.

C. Discussion of the Model

Our model is general and does not assume specific structures of the system. The operation configuration $m_n[k]$ can be used to describe the specific bidding policy used during the advertising interval, i.e., automatic bidding or customized bidding scheme, e.g., [8]. In this case, the functions $G_n(\cdot, \cdot)$ and $F_n(\cdot, \cdot)$ represent the expected revenue and interval length under such bidding schemes. Therefore, the strategies developed in this paper can indeed be implemented with the existing bidding schemes. Under such general settings, solving this problem is challenging, since the performance metrics in (4) and (5) are equivalent to sum of ratios of functions and are generally non-convex. Moreover, different search sites may need updates asynchronously, whereas most existing optimal control algorithms [14] [15] require synchronous action updates at all components of the system.

III. Characterizing Optimal Revenue

In this section, we first present a theorem which characterizes the advertiser's maximum revenue. The main idea of the theorem is that the maximum budget-constrained revenue can in principle be computed by solving a complex optimization problem (typically nonconvex). The theorem will later be used for our analysis. In the theorem, the parameter $V \geq 1$ is a constant introduced for our later analysis.

Theorem 1: The maximum time average revenue $\text{Profit}_{\text{av}}^*$ satisfies $V\text{Profit}_{\text{av}}^* \leq \Phi^*$, where Φ^* is the optimal value of the following optimization problem. [4]

$$\max: \ \Phi \triangleq \sum_n \frac{V \sum_{i=0}^{\infty} \alpha_{ni} G_n(p_{ni}, m_{ni})}{\sum_{i=0}^{\infty} \alpha_{ni} F_n(p_{ni}, m_{ni}) + \sum_{i=0}^{\infty} \alpha_{ni} T_{ni}^{\text{fr}}} \quad (6)$$

$$\text{s.t.} \ \sum_n \frac{\sum_{i=0}^{\infty} \alpha_{ni} p_{ni}}{\sum_{i=0}^{\infty} \alpha_{ni} F_n(p_{ni}, m_{ni}) + \sum_{i=0}^{\infty} \alpha_{ni} T_{ni}^{\text{fr}}} \leq B_{\text{av}}$$

$$\sum_{i=0}^{\infty} \alpha_{ni} = 1, \alpha_{ni} \geq 0, \forall\, n$$

$$(p_{ni}, T_{ni}^{\text{fr}}) \in \mathcal{P}_n, m_{ni} \in \mathcal{M}_n, \ \forall\, n, i. \quad \square$$

Proof: The proof can be done by using an argument based on Caratheodory's theorem as in [14]. Omitted for brevity. ■

In Theorem 1, the values $\{\alpha_{ni}\}_{i=0}^{\infty}$ can be viewed as the probabilities of choosing the investment-freeze-configuration triple $(p_{ni}, T_{ni}^{\text{fr}}, m_{ni})$ in every frame for search site S_n. With this interpretation, we immediately have the following corollary.

Corollary 1: There exists an optimal stationary and randomized investment strategy Π_S, which works as follows: for

[2] Note that this is a reasonable assumption. If the advertiser invests nonzero money (typically must be more than a minimum), it will take some nonzero time to deplete the money. Otherwise if the advertiser suspends his investment at the site, then he will also choose a nonzero freeze time and resume later.

[3] Throughout this paper, we assume that all limits and corresponding time average values exist with probability 1. Note that this is not restrictive. In particular, the limits in (4) and (5) exist whenever the relevant quantities are bounded and the control policy results in a well-defined distribution over the actions.

[4] Note that we use the expectations for all the relevant quantities in (6). It can be shown that, if a policy results in a well-defined distribution over the actions, the average revenue and average expense can be expressed with their expected values in the form given in (6).

each search site S_n, after each frame, chooses an investment-freeze-configuration triple $\{(p_{ni}, T^{\text{fr}}_{ni}, m_{ni})\}_{i=0}^{\infty}$ with probabilities $\{\alpha_{ni}\}_{i=0}^{\infty}$, independent of other sites. In particular, Π_S achieves the followings:

$$\sum_n \frac{V\mathbb{E}\{R_n[k]\}}{\mathbb{E}\{T^{\text{ad}}_n[k] + T^{\text{fr}}_n[k]\}} = \Phi^*,$$

$$\sum_n \frac{\mathbb{E}\{p_n[k]\}}{\mathbb{E}\{T^{\text{ad}}_n[k] + T^{\text{fr}}_n[k]\}} \leq B_{\text{av}}. \quad \square$$

However, as discussed above, due to the non-convex nature of problem (6), directly solving for such a randomized policy is very difficult. In the following, we instead develop an online algorithm, which allows the advertiser to *asynchronously* update his investment profile at different search sites and achieve near-optimal profit. Our algorithm only requires *greedy* updates of each site and can be implemented efficiently in practice.

IV. Online Algorithm for Achieving Near Optimal Revenue

In this section, we develop an investment strategy to achieve a near optimal performance for the advertiser. Our algorithm is based on the renewal and asynchronous system optimization technique developed in [16] and [17].

A. The Virtual Deficit Queue

To start, we first define the notion of *decision points*. A time t is called a decision point if it marks the beginning of an advertising interval for any site. Then, we use $\{t_d\}_{d=0}^{\infty}$ to denote the set of decision points. We see then the timeline is divided into disjoint intervals by the decision points. Fig. 2 shows five such points $\{t_0, t_1, ..., t_4\}$. Since each frame is at least $T^{\min}$ long, we see that $\lim_{d\to\infty} t_d = \infty$ w.p 1.

We use $k_n(t)$ to denote the number of advertising intervals that site n has completed up to time t. Note that $k_n(t)$ also denotes the index of the advertising interval the search site is in at time t. We denote $\delta_d \triangleq t_{d+1} - t_d$ the length of the interval between decision points t_d and t_{d+1}. For our analysis, it is also useful to define the notions of *effective budget consumption rate* $a_n[t_d]$, and *effective revenue generating rate* $r_n[t_d]$ over the time interval $[t_d, t_{d+1})$ for each site S_n as follows:

$$a_n[t_d] \triangleq \frac{p_n[k_n(t_d)]}{F_n[k_n(t_d)] + T^{\text{fr}}_n[k_n(t_d)]}, \tag{7}$$

$$r_n[t_d] \triangleq \frac{G_n[k_n(t_d)]}{F_n[k_n(t_d)] + T^{\text{fr}}_n[k_n(t_d)]}. \tag{8}$$

Here $a_n[t_d]$ and $r_n[t_d]$ roughly denote the average budget consumption rate and the average revenue generation rate. Note that we use the *expected* values of the advertising interval length in both (7) and (8).

Now we define the *deficit queue process* $Q(t_d)$, to keep track of the budget deficit due to temporal violation of the average budget constraint. $Q(t_d)$ evolves according to the following dynamics:

$$Q(t_{d+1}) = \max\left[Q(t_d) - \mu[t_d], 0\right] + A[t_d], \tag{9}$$

with $Q(t_0) = 0$. Here $A[t_d]$ denotes the *aggregate* budget consumed by all the sites during $[t_d, t_{d+1})$, i.e.,

$$A[t_d] = \delta_d \sum_n a_n[t_d]. \tag{10}$$

Similarly, $\mu[t_d]$ denotes the amount of "allowed" average budget consumption, i.e.,

$$\mu[t_d] = B_{\text{av}}(t_{d+1} - t_d) = \delta_d B_{\text{av}}. \tag{11}$$

We notice that since the update of $Q(t_d)$ uses *expected* lengths of the actual advertising intervals, it is indeed an *approximation* of the true deficit. The intuition here is that $Q(t_d)$ represents the expected value of the true deficit. Thus, one can ensure the actual budget constraint (5) by stabilizing $Q(t_d)$.

For notational convenience, let us also define $c_{\max} = T^{\max}\max[\sum_n \frac{p_n^{\max}}{T_n^{\min}}, B_{\text{av}}]$. We see then $A[t_d], \mu[t_d] \leq c_{\max}$ for all t_d.

B. The Dynamic Ad-Investment (AI) *Algorithm*

Now we present our investment algorithm. In the algorithm, $\mathcal{S}(t_d)$ denotes the set of search engine sites that just finish a frame and will start a new advertising interval at time t_d. The algorithm works as follows:

Ad-Investment (AI): At each time t_d, denote $k_n = k_n(t_d)$ the frame index of S_n at time t_d. For each search site $S_n \in \mathcal{S}(t_d)$, observe $Q(t_d)$ and perform the following:

1) Choose $p_n[k_n]$, $m_n[k_n]$ and $T^{\text{fr}}_n[k_n]$ that solve the following:

$$\max: \quad \Psi_n(t_d) \triangleq \frac{VG_n(p_n[k_n], m_n[k_n]) - Q(t_d)p_n[k_n]}{F_n(p_n[k_n], m_n[k_n]) + T^{\text{fr}}_n[k_n]} \tag{12}$$

$$\text{s.t.} \quad (p_n[k_n], T^{\text{fr}}_n[k_n]) \in \mathcal{P}_n,\ m_n[k_n] \in \mathcal{M}_n.$$

Then, update the investment and the advertising configuration at S_n with the chosen actions.

2) Update $Q(t_d)$ according to (9). $\diamond$

Denote $\Psi^*_n(t_d)$ the optimal value of (12). We note that since $\mathcal{P}_n$ includes the option $p_n = 0$ and $G_n(0, \cdot) = 0$, at the optimal solution we always have $\Psi^*_n(t_d) \geq 0$. We also note that the algorithm involves solving problem (12) for each search site *separately*. Hence, the advertiser can easily update his configuration at any site after the current frame completes. This is quite different from many existing algorithms that update all control actions simultaneously. Also note that (12) can often be solved efficiently in practice, especially when the actions sets $\mathcal{P}_n$ and $\mathcal{M}_n$ are finite. Finally, note that in every step, we use $F_n(p, m)$ and $G_n(p, m)$ for decision making. This requires the advertiser to have accurate prior estimation of the system dynamics. We will see in Section V that AI can also be implemented with imperfect knowledge of the system dynamics, and still achieve good performance.

C. Performance Analysis

Define the Lyapunov function $L(t_d) = \frac{1}{2}Q(t_d)^2$. Then, we define a 1-slot Lyapunov drift as follows:

$$\Delta(t_d) \triangleq L(t_{d+1}) - L(t_d). \tag{13}$$

Using the queueing dynamic equation (9), we have the following lemma.

Lemma 1: At any time t_d, we have:

$$\Delta(t_d) \leq C_0 - Q(t_d)\big(\delta_d B_{\text{av}} - A[t_d]\big), \tag{14}$$

where $C_0 \triangleq \frac{1}{2}(T^{\max})^2\big[(\sum_n \frac{p_n^{\max}}{T_n^{\min}})^2 + B_{\text{av}}^2\big]$. □

Proof: See Appendix A. ■

Adding the term $-V\delta_d \sum_n r_n[t_d]$ to both sides of (14) and rearranging terms, we obtain:

$$\begin{aligned}
&\Delta(t_d) - V\delta_d \sum_n r_n[t_d] \\
&\quad \leq C_0 - Q(t_d)\delta_d B_{\text{av}} \\
&\qquad -\delta_d \sum_n \frac{VG_n[k_n(t_d)] - Q(t_d)p_n[k_n(t_d)]}{F_n[k_n(t_d)] + T_n^{\text{fr}}[k_n(t_d)]} \\
&\quad \leq C_0 - Q(t_d)\delta_d B_{\text{av}} - \delta_d \sum_n \Psi_n(t_d). \qquad (15)
\end{aligned}$$

From (15), one sees that the AI algorithm is indeed constructed to maximize $\Psi_n(t_d)$ on the right-hand-side (RHS) of the drift inequality for every $S_n \in \mathcal{S}(t_d)$. The following lemma shows that AI also approximately maximizes $\sum_n \Psi_n(t_d)$.

***Lemma** 2:* Under the AI algorithm, at every time t_d, we have:

$$\sum_n \Psi_n^{\mathsf{AI}}(t_d) \geq \max_{\Pi}\left(\sum_n \Psi_n^{\Pi}(t_d)\right) - C_1. \qquad (16)$$

Here Π is any feasible investment policy in choosing $p_n[k_n(t_d)]$, $m_n[k_n(t_d)]$, and $T_n^{\text{fr}}[k_n(t_d)]$ at time t_d, and $C_1 = 2T^{\max}\sum_n \frac{p_n^{\max}}{T_n^{\min}} = \Theta(1)$, independent of V. □

Proof: See Appendix B. ■

Lemma 2 is important for our analysis. In practice, due to the dynamics in the advertising process, sites typically do not complete their frames at the same time. Hence, at every instance of time, only some search sites need to update their configurations. In this case, if instead a synchronous scheme is applied, some sites will have to remain idle while advertising is still going on at others. This will inevitably result in resources being wasted. Lemma 2 shows that AI roughly minimizes the RHS of the drift inequality (15) even with asynchronous updates. This guarantees the near-optimal performance of AI, as summarized in the following theorem.

***Theorem** 2:* The AI algorithm achieves the following:

a) The average revenue under AI, denoted by $\text{Profit}_{\text{av}}^{\mathsf{AI}}$, satisfies:

$$\text{Profit}_{\text{av}}^{\mathsf{AI}} \geq \text{Profit}_{\text{av}}^{*} - \frac{C_1}{V} - \frac{C_0}{VT^{\min}}. \qquad (17)$$

Here C_0 and C_1 are $\Theta(1)$ constants defined in Lemmas 1 and 2.

b) At every time t_d, we have:

$$Q(t_d) \leq V\nu + 2c_{\max}. \qquad (18)$$

Moreover, the average budget constraint (5) is satisfied with probability 1. □

Proof: See Appendix C. ■

By tuning the value of $V = 1/\epsilon$, Theorem 2 implies that AI can achieve a revenue that is within $O(\epsilon)$ of the maximum, for any $\epsilon > 0$, with a tradeoff that $Q(t_d) = O(1/\epsilon)$. Notice that (18) does not directly imply that the actual deficit is deterministically bounded. This is because the actual frame length is a random variable while $Q(t_d)$ is updated with the expected values. Despite this approximation, Part b) of Theorem 2 shows that so long as $Q(t_d)$ is stable, (5) will be satisfied with probability 1.

V. Performance under Imperfect Estimation

So far, we have assumed that the advertiser knows exactly the expected revenue that will be generated during a frame and the expected duration of the frame. In this section, we consider the more general case where the advertiser only has imperfect estimations of the functions G_n and F_n, and quantify the impact of such inaccuracy on the performance of the AI algorithm.

Specifically, we assume that now the advertiser does not have the perfect knowledge of $G_n(p,m)$ and $F_n(p,m)$. Instead, he uses his own estimated functions $\hat{G}_n(p,m)$ and $\hat{F}_n(p,m)$ for algorithm design and decision making. That is, $Q(t_d)$ and $\Psi_n(t_d)$ will be defined with $\hat{G}_n(p,m)$ and $\hat{F}_n(p,m)$, and (12) will be solved with them as well. In order to quantify the impact of the estimation errors, we first introduce the quality index of the estimations, (ρ_G, ρ_F). That is, we say that the advertiser has an estimation quality (ρ_G, ρ_F) if for all S_n,

$$\begin{aligned}
|\hat{G}_n(p,m) - G_n(p,m)| &\leq \rho_G G_n(p,m),\ \forall\, p, m, (19) \\
|\hat{F}_n(p,m) - F_n(p,m)| &\leq \rho_F F_n(p,m),\ \forall\, p, m. (20)
\end{aligned}$$

Thus, ρ_F and ρ_G capture the inaccuracy levels of the estimations.

We now have the following theorem regarding the performance of the AI algorithm in this case.

***Theorem** 3:* Suppose the advertiser has an estimation quality (ρ_G, ρ_F), where $\rho_G, \rho_F \in [0, 1)$. Then, the AI algorithm achieves the following:

a) Revenue performance:

$$\text{Profit}_{\text{av}}^{\mathsf{AI}} \geq \frac{(1-\rho_F)\text{Profit}_{\text{av}}^{*}}{(1+\rho_F)(1+\rho_G)} - \Theta(\frac{1}{V}) - \frac{N\rho_G G^{\max}}{(1+\rho_G)T^{\min}}.$$

b) Budget control: At every time t_d, we have:

$$Q(t_d) \leq V(1+\rho_G)\nu + 2\hat{c}_{\max}. \qquad (21)$$

Here $\hat{c}_{\max}$ is defined as the maximum change of $Q(t_d)$ defined with $\hat{G}_n(p,m)$ and $\hat{F}_n(p,m)$, i.e.,

$$\hat{c}_{\max} = (1+\rho_F)T^{\max}\max[\sum_n \frac{p_n^{\max}}{(1-\rho_F)T_n^{\min}}, B_{\text{av}}].$$

Moreover, the budget constraint (5) is satisfied with probability 1 if we replace B_{av} with $(1+\rho_F)B_{\text{av}}$. □

Proof: Omitted due to space limit. Please see [20]. ■

Part b) of Theorem 3 indicates that the budget constraint may be slightly violated if the estimations of G_n and F_n are inaccurate. While such violation is typical in this case, Theorem 3 shows that as long as one uses a scaled budget in the algorithm, i.e., use $B_{\text{av}}/(1+\rho_F)$, the original advertisement budget can be guaranteed. This thus provides useful guidelines for the actual algorithm implementation. Also note that Theorem 3 applies to more general cases where the system's dynamics and the advertiser's estimation quality both vary over time. In this case, if the advertiser's estimation quality stays better than a certain quality (ρ_G, ρ_F) after some finite learning time, then one can invoke Theorem 3 and conclude the performance of the algorithm.

VI. Simulation

In this section, we provide simulation results for our algorithm. We consider the case when there two sites $\{S_1, S_2\}$.

For each site S_n, we assume that $m_n = \{0.1, 0.2\}$ only denotes the maximum pay-per-click and ignore other factors.

The $G_n(p, m)$ and $F_n(p, m)$ functions are given by:

$$F_n(p, m) = \frac{\kappa_n p}{m}, \quad G_n(p, m) = \gamma_n \sqrt{\frac{p}{m}} m^q. \tag{22}$$

The function F_n is chosen to model the fact that the expected duration of the advertising interval is proportional to $\frac{p}{m}$, i.e., roughly the number of times the advertiser wins the ad opportunity. The choice of function G_n assumes that whenever an ad opportunity arises, the advertiser wins the opportunity with probability m^q, and that it roughly takes p/m ad displays to generate $\sqrt{p/m}$ revenue.

For both S_n, we assume that the feasible (investment-freeze) set is given by:

$$\mathcal{P}_n = \{(p = 0, T = 5), (p = 5, T = 0), (p = 5, T = 5),$$
$$(p = 10, T = 0), (p = 10, T = 5)\}.$$

For each action, we assume that the actual revenue is uniformly distributed in $[0.8G_n, 1.2G_n]$. Also, the advertising interval length is uniformly distributed in $[0.8F_n, 1.2F_n]$. We set $\kappa_1 = 1$ and $\kappa_2 = 2$; $\gamma_1 = 1$ and $\gamma_2 = 2$; $B_{\text{av}} = 0.2$ and we use $q = 0.2$. We simulation the algorithm for $V \in \{5, 10, 20, 50, 100, 200\}$.

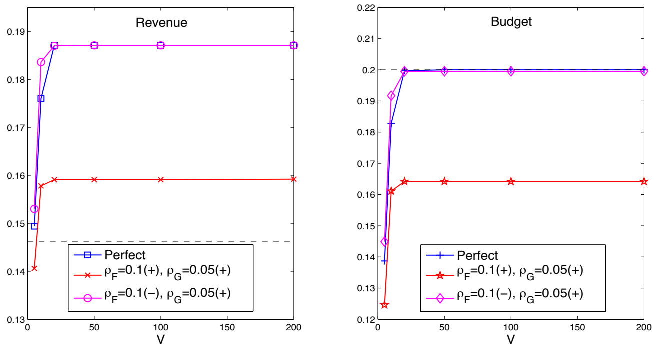

Fig. 3. Performance of the AI algorithm. The legend $\rho_F = 0.1(\pm), \rho_G = 0.05(\pm)$ means that $\hat{F_n}(p, m) = (1 \pm 0.1)F_n(p, m)$ and $\hat{G_n}(p, m) = (1 \pm 0.05)G_n(p, m)$. We see that when there is estimation error, using a scaled budget value easily guarantees the original advertising budget constraint.

First, we see from Fig. 3 that as the V value increases, the average revenue quickly converges to the optimal when there is no estimation errors. In the imperfect estimation case, we simulate AI with a scaled budget $B_{\text{av}}/(1 + \rho_F)$. In this case, we see from the right plot that the actual budget never exceeds $B_{\text{av}} = 0.2$. Also, the left plot shows that the revenue performance of AI is always above $\frac{1-\rho_F}{(1+\rho_F)(1+\rho_G)}\text{Profit}^*_{\text{av}}$ (indicated by the black dotted line), which implies that it is also always no less than $\frac{1-\rho_F}{(1+\rho_F)(1+\rho_G)}$ fraction of the optimal revenue with budget constraint $B_{\text{av}}/(1 + \rho_F)$. These results are consistent with Theorems 2 and 3.

To evaluate the effect of V on budget violation, Fig. 4 also plots the average deficit queue size. We see that it increases linearly in V. The right plot of Fig. 4 also shows the time when the two sites update their configurations, for the case when $V = 20$ and there is no estimation error. It can be seen that under AI, the site updates take place *asynchronously*.

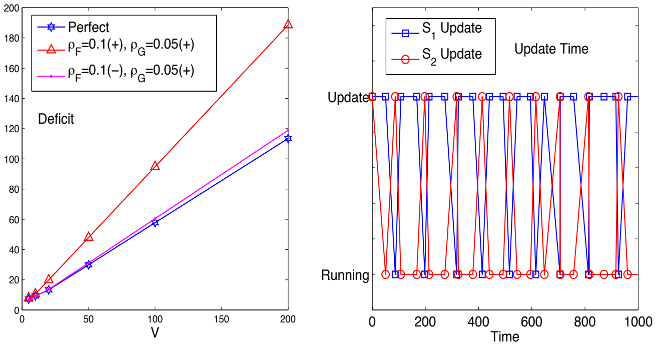

Fig. 4. The left plot shows that the average deficit under the AI algorithm increases linearly in the V parameter. The right plot shows that the advertiser updates the advertisement and configuration of the sites asynchronously.

VII. Conclusion

In this paper, we design an optimal investment strategy for a single advertiser who simultaneously utilizes multiple sponsored search engine sites for advertising. An *asynchronous* investment strategy Ad-Investment (AI) is proposed. The algorithm allows the advertiser to asynchronously update his investment configuration at each site. We show that the algorithm achieves an average revenue that is within $O(\epsilon)$ of the optimal, for any $\epsilon > 0$, while guaranteeing that the temporal budget violation is $O(1/\epsilon)$. We also analyze the performance of AI under imperfect system estimation and show that it is robust against estimation errors.

VIII. Acknowledgement

This work was supported in part by the National Basic Research Program of China Grant 2011CBA00300, 2011CBA00301, the National Natural Science Foundation of China Grant 61033001, 61361136003, 61303195, the China youth 1000-talent grant.

Appendix A – Proof of Lemma 1

Here we prove Lemma 1.

Proof: Squaring both sides of (9) and using the fact that $(\max[x, 0])^2 \leq x^2$ for all $x \in \mathbb{R}$, we have:

$$Q^2(t_{d+1}) \leq Q^2(t_d) + (\mu[t_d])^2 + (A[t_d])^2$$
$$-2Q(t_d)\big(\mu[t_d] - A[t_d]\big). \tag{23}$$

Now using the facts that $a_n[t_d] \leq p_n^{\max}/T_n^{\min}$ and $\delta_d \leq T^{\max}$, we see that at any time t_d,

$$\frac{1}{2}(A[t_d])^2 + \frac{1}{2}(\mu[t_d])^2$$
$$\leq C_0 \triangleq \frac{1}{2}(T^{\max})^2\Big[(\sum_n \frac{p_n^{\max}}{T_n^{\min}})^2 + B_{\text{av}}^2\Big]. \tag{24}$$

Multiplying both sides of (23) by $\frac{1}{2}$ and using the definition of C_0, we sees that the lemma follows. ■

Appendix B – Proof of Lemma 2

Proof: To prove the lemma, we first recall that the $\Psi_n(t_d)$ function is defined as:

$$\Psi_n(t_d) \triangleq \frac{VG_n(p_n[k_n], m_n[k_n]) - Q(t_d)p_n[k_n]}{F_n(p_n[k_n], m_n[k_n]) + T_n^{\text{fr}}[k_n]} \tag{25}$$

Here $k_n = k_n[t_d]$.

If at time t_d, $S_n \in \mathcal{S}(t_d)$, then $\Psi_n(t_d)$ is maximized over all possible pure strategies, i.e., strategies that choose a particular $(p_n[k], m_n[k], T_n^{\text{fr}}[k])$ with probability 1. Now notice that for any c_1, d_1, c_2, d_2 where $d_1, d_2 \neq 0$ and $\theta \in [0, 1]$, we have:

$$\frac{\theta c_1 + (1-\theta)c_2}{\theta d_1 + (1-\theta)d_2} = \eta\frac{c_1}{d_1} + (1-\eta)\frac{c_2}{d_2}, \tag{26}$$

for some $\eta \in [0, 1]$ [21]. Therefore, if a policy Π mixes two pure strategies Π_1 and Π_2, i.e., uses two triples (p_1, m_1, T_1) and (p_2, m_2, T_2) with probabilities $a_1 > 0$ and $a_2 > 0$, where $a_1 + a_2 = 1$, then there exists $\eta_\Pi \in [0, 1]$ that:

$$\Psi_n^{\Pi}(t_d) = \eta_\Pi\Psi_n^1(t_d) + (1-\eta_\Pi)\Psi_n^2(t_d) \leq \Psi_n^{\mathsf{AI}}(t_d). \tag{27}$$

By induction, we see that $\Psi_n(t_d)$ is maximized under AI over all strategies for all $S_n \in \mathcal{S}(t_d)$.

It remains to show that for any other $S_n \notin \mathcal{S}(t_d)$, the function $\Psi_n(t_d)$ is also *approximately* maximized. Consider an $S_n \notin \mathcal{S}(t_d)$. Let $\tilde{t}_n$ be the last time the advertiser updates his configuration at S_n before t_d, i.e., $S_n \in \mathcal{S}(\tilde{t}_n)$ but $S_n \notin \mathcal{S}(t'_d)$ for all $\tilde{t}_n < t'_d \leq t_d$. Note that $t_d - \tilde{t}_n \leq T^{\max}$. This is because the maximum frame length of any site S_n is bounded by $T^{\max}$. Note that at time t_d, the site S_n is still operating under the actions chosen at time $\tilde{t}_n$, which is based on the queue size $Q(\tilde{t}_n)$. Denote the actions chosen then by $\tilde{p}_n$, $\tilde{m}_n$ and $\tilde{T}_n^{\text{fr}}$.

Denote p_n^*, m_n^* and $T_n^{\text{fr}*}$ the actions chosen at time t_d by solving (12) with the backlog $Q(t_d)$ for this $S_n \notin \mathcal{S}(t'_d)$. Note that these actions are not actually implemented because the advertiser will not update the configuration of S_n at time t_d. Then, we have:

$$\begin{aligned}
&\frac{VG_n(\tilde{p}_n, \tilde{m}_n) - Q(t_d)\tilde{p}_n}{F_n(\tilde{p}_n, \tilde{m}_n) + \tilde{T}_n^{\text{fr}}} \\
&= \frac{VG_n(\tilde{p}_n, \tilde{m}_n) - Q(\tilde{t}_n)\tilde{p}_n}{F_n(\tilde{p}_n, \tilde{m}_n) + \tilde{T}_n^{\text{fr}}} - \frac{(Q(t_d) - Q(\tilde{t}_n))\tilde{p}_n}{F_n(\tilde{p}_n, \tilde{m}_n) + \tilde{T}_n^{\text{fr}}} \\
&\overset{(a)}{\geq} \frac{VG_n(p_n^*, m_n^*) - Q(\tilde{t}_n)p_n^*}{F_n(p_n^*, m_n^*) + T_n^{\text{fr}*}} + \frac{(Q(t_d) - Q(\tilde{t}_n))\tilde{p}_n}{F_n(\tilde{p}_n, \tilde{m}_n) + \tilde{T}_n^{\text{fr}}} \\
&= \frac{VG_n(p_n^*, m_n^*) - Q(t_d)p_n^*}{F_n(p_n^*, m_n^*) + T_n^{\text{fr}*}} - \frac{(Q(t_d) - Q(\tilde{t}_n))\tilde{p}_n}{F_n(\tilde{p}_n, \tilde{m}_n) + \tilde{T}_n^{\text{fr}}} \\
&\quad + \frac{(Q(t_d) - Q(\tilde{t}_n))p_n^*}{F_n(p_n^*, m_n^*) + T_n^{\text{fr}*}}.
\end{aligned} \tag{28}$$

Here the inequality (a) is because $\tilde{p}_n$, $\tilde{m}_n$ and $\tilde{T}_n^{\text{fr}}$ maximize (12) given $Q(\tilde{t}_n)$.

Using the queueing dynamic equation (9), we see that: $Q(t_d) - (t_d - \tilde{t}_n)c_{\max} \leq Q(\tilde{t}_n) \leq Q(t_d) + (t_d - \tilde{t}_n)c_{\max}$. Together with the fact that $t_d - \tilde{t}_n \leq T^{\max}$, we have:

$$|Q(t_d) - Q(\tilde{t}_n)| \leq T^{\max}c_{\max}.$$

Using this in (28), we get:

$$\frac{VG_n(\tilde{p}_n, \tilde{m}_n) - Q(t_d)\tilde{p}_n}{F_n(\tilde{p}_n, \tilde{m}_n) + \tilde{T}_n^{\text{fr}}} \geq \Psi_n^*(t_d) - C_{1n}, \tag{29}$$

where $\Psi_n^*(t_d)$ is the value of $\Psi_n(t_d)$ with the actions p_n^*, m_n^*, $T_n^{\text{fr}*}$, and $Q(t_k)$, and $C_{1n} \triangleq \frac{2T^{\max}c_{\max}p_n^{\max}}{T_n^{\min}}$. This shows that even for $S_n \notin \mathcal{S}(t_d)$, $\Psi_n(t_d)$ is approximately maximized. Now by defining $C_1 = \sum_n C_{1n}$ and using a similar argument as in (27), we see that the lemma follows. ■

Appendix C – Proof of Theorem 2

Here we prove Theorem 2.

Proof: (Theorem 2) (Revenue Performance) We first prove the revenue performance. To do so, we recall the drift inequality (15) as follows:

$$\begin{aligned}
&\Delta(t_d) - V\delta_d\sum_n r_n[t_d] \\
&\qquad \leq C_0 - Q(t_d)\delta_d B_{\text{av}} - \delta_d\sum_n \Psi_n(t_d).
\end{aligned} \tag{30}$$

By Lemma 2, we note that AI maximizes the term $\sum_n \Psi_n(t_d)$ over all possible investment policies to within a finite constant C_1. Thus, we can plug into the RHS of (30) any control algorithm and the following holds:

$$\begin{aligned}
&\Delta(t_d) - V\delta_d\sum_n r_n[t_d] \\
&\qquad \leq C_0 + C_1\delta_d - Q(t_d)\delta_d B_{\text{av}} - \delta_d\sum_n \Psi_n^{\text{ALT}}(t_d).
\end{aligned} \tag{31}$$

Therefore, we plug the randomized and stationary policy Π_S in Corollary 1 into (30) and obtain:

$$\Delta(t_d) - V\delta_d\sum_n r_n[t_d] \leq C_0 + \delta_d C_1 - \delta_d\Phi^*. \tag{32}$$

Taking expectations on both sides, carrying out a telescoping sum from $d = 0$ to $D-1$, and rearranging terms, we obtain:

$$\begin{aligned}
&V\sum_{d=0}^{D-1}\mathbb{E}\big\{\delta_d\sum_n r_n[t_d]\big\} \\
&\qquad \geq \sum_{d=0}^{D-1}\mathbb{E}\big\{\delta_d\big\}(\Phi^* - C_1) - DC_0 - \mathbb{E}\big\{L(t_0)\big\}.
\end{aligned}$$

Dividing both sides by $V\sum_d \mathbb{E}\big\{\delta_d\big\}$ and letting $D \to \infty$,

$$\begin{aligned}
&\lim_{D\to\infty}\frac{\sum_{d=0}^{D-1}\mathbb{E}\big\{\delta_d\sum_n r_n[t_d]\big\}}{\sum_{d=0}^{D-1}\mathbb{E}\big\{\delta_d\big\}} \\
&\qquad \geq \Phi^*/V - \frac{C_1}{V} - \lim_{D\to\infty}\frac{C_0 D}{V\sum_{d=0}^{D-1}\mathbb{E}\big\{\delta_d\big\}} \\
&\qquad \geq \text{Profit}_{\text{av}}^* - \frac{C_1}{V} - \frac{C_0}{VT^{\min}}.
\end{aligned} \tag{33}$$

It remains to show that the actual average revenue also satisfies (33). Denote $z_n^{p,m,T}$ the long term fraction of update instances that (p, m, T) is used at S_n under AI. Because for any (p, m), $\mathbb{E}\big\{R_n[k]\big\}$ and $\mathbb{E}\big\{T_n^{\text{ad}}[k]\big\}$ are bounded, we use the strong law of large numbers (SLLN) [22] to conclude:

$$\begin{aligned}
&\sum_n \frac{\lim_{K\to\infty}\frac{1}{K}\sum_{k=0}^{K-1} R_n^{\mathsf{AI}}[k]}{\lim_{K\to\infty}\frac{1}{K}\sum_{k=0}^{K-1}(T_n^{\text{ad},\mathsf{AI}}[k] + T_n^{\text{fr},\mathsf{AI}}[k])} \\
&\qquad = \sum_n \frac{\sum_{p,m,T} z_n^{p,m,T}G_n(p, m)}{\sum_{p,m,T} z_n^{p,m,T}[F_n(p, m) + T]}.
\end{aligned} \tag{34}$$

Fix a search site S_n. We see that for the t_d values that are in the middle of a frame of S_n, $r_n[t_d]$ remains unchanged. Let $\tilde{t}_n$ be the last time S_n starts a new frame before t_D, and let $\mathcal{J}_n^D(p, m, T)$ be the set of frames during which the triple (p, m, T) is adopted at site S_n up to t_D. We have:

$$\begin{aligned}
&\sum_{d=0}^{D-1}\mathbb{E}\big\{\delta_d r_n[t_d]\big\} \\
&= \sum_{p,m,T}\frac{G_n(p, m)}{F_n(p, m) + T}\mathbb{E}\big\{\sum_{j\in\mathcal{J}_n^D(p,m,T)}(T_n^{\text{ad}}[j] + T_n^{\text{fr}}[j])\big\} \\
&\quad + \mathbb{E}\big\{r_n[t_D](t_D - \tilde{t}_n)\big\}
\end{aligned}$$

$$
\begin{aligned}
&= \sum_{p,m,T} G_n(p,m)\mathbb{E}\big\{|\mathcal{J}_n^D(p,m,T)|\big\} \\
&\quad + \mathbb{E}\big\{r_n[t_D](t_D - \tilde{t}_n)\big\}.
\end{aligned}
$$

In this case, we see that:

$$
\begin{aligned}
&\lim_{D\to\infty} \frac{\sum_n \sum_{d=0}^{D-1} \mathbb{E}\big\{\delta_d r_n[t_d]\big\}}{\sum_{d=0}^{D-1} \mathbb{E}\big\{\delta_d\big\}} \\
&= \lim_{D\to\infty} \sum_n \sum_{p,m,T} G_n(p,m) \frac{\mathbb{E}\big\{|\mathcal{J}_n^D(p,m,T)|\big\}}{\sum_{d=0}^{D-1} \mathbb{E}\big\{\delta_d\big\}} \qquad (35) \\
&\quad + \lim_{D\to\infty} \sum_n \frac{\mathbb{E}\big\{r_n[t_D](t_D - \tilde{t}_n)\big\}}{\sum_{d=0}^{D-1} \mathbb{E}\big\{\delta_d\big\}}
\end{aligned}
$$

Since $(t_D - \tilde{t}_n) \leq T_n^{\max}$, $r_n[t_D] \leq \frac{G^{\max}}{T_n^{\min}}$, and $\lim_{D\to\infty} \sum_{d=0}^{D-1} \mathbb{E}\big\{\delta_d\big\} = \infty$, we see that the last term vanishes. Moreover, it can be shown that for all S_n:

$$
\begin{aligned}
&\lim_{D\to\infty} \frac{\mathbb{E}\big\{|\mathcal{J}_n^D(p,m,T)|\big\}}{\sum_{d=0}^{D-1} \mathbb{E}\big\{\delta_d\big\}} \\
&= \lim_{D\to\infty} \frac{\mathbb{E}\big\{|\mathcal{J}_n^D(p,m,T)|\big\}}{\sum_{p,m,T} \mathbb{E}\big\{|\mathcal{J}_n^D(p,m,T)|\big\}[F_n(p,m)+T]} \\
&= \frac{z_n^{p,m,T}}{\sum_{p,m,T} z^{p,m,T}[F_n(p,m)+T]}.
\end{aligned}
$$

Using this in (35), we have:

$$
\begin{aligned}
&\lim_{D\to\infty} \frac{\sum_n \sum_{d=0}^{D-1} \mathbb{E}\big\{\delta_d r_n[t_d]\big\}}{\sum_{d=0}^{D-1} \mathbb{E}\big\{\delta_d\big\}} \\
&\qquad = \sum_n \frac{\sum_{p,m,T} z_n^{p,m,T} G_n(p,m)}{\sum_{p,m,T} z_n^{p,m,T}[F_n(p,m)+T]}. \qquad (36)
\end{aligned}
$$

This and (34) show that the average revenue satisfies:

$$
\text{Profit}_{\text{av}}^{\mathsf{AI}} \geq \text{Profit}_{\text{av}}^{*} - \frac{C_1}{V} - \frac{C_0}{VT^{\min}}. \qquad (37)
$$

(Budget) We now prove that the average budget constraint (5) is ensured under the AI algorithm.

We first prove (18). To see that (18) holds, note that $c_{\max}$ is the maximum change of $Q(t_d)$ during any interval $[t_d, t_{d+1})$. Thus, if $Q(t_d) < V\nu$ at time k, then we must have $Q(t_{d+1}) \leq V\nu + c_{\max}$. On the other hand, suppose now $Q(t_d) > V\nu$. Without loss of generality, assume that $Q(t_d)$ is the first time $Q(t_d) > V\nu$ occurs. One sees that $Q(t_d) \leq V\nu + c_{\max}$. Also, we see that from now on until $Q(t_d)$ drops below $V\nu$ again, the advertiser will choose $p_n = 0$ for any S_n that requires update during this interval. This is because once $Q(t_d) > V\nu$, choosing any $p_n > 0$ will result in $\Psi_n(t_d) < 0$. This implies that $a_n[t_d] = 0$ for all $S_n \in \mathcal{S}(t_d)$ during this time. Hence, the only possible increment of $Q(t_d)$ will be due to the sites that are in the middle of their frames. $Q(t_d)$ will also start to decrease once all the active sites complete their current frames. This increasing interval will last for at most $T^{\max}$ long. Also, for each active site, $a_n[t_d] \leq \frac{p_n^{\max}}{T_n^{\min}}$. Thus, the maximum possible increment after $Q(t_d) \geq V\nu$ will be $c_{\max}$. Combining this fact with $Q(t_d) \leq V\nu + c_{\max}$, we conclude that $Q(t_d) \leq V\nu + 2c_{\max}$ for all t_d.

This shows that the deficit queue with $A[t_d]$ and $\mu[t_d]$ being the arrival and service rates is bounded. Using a similar argument as in the revenue part, one can show that:

$$
\sum_n \frac{\sum_{p,m,T} z_n^{p,m,T} p}{\sum_{p,m,T} z_n^{p,m,T}[F_n(p,m)+T]} \leq B_{\text{av}}.
$$

Using SLLN again, one sees that the LHS equals the average expenditure with probability 1. Hence, (5) is satisfied. ■

References

[1] Emarketer.com. Us online advertising spending to surpass print in 2012. *http://www.emarketer.com/Article/US-Online-Advertising-Spending-Surpass-Print-2012/1008783, accessed on July 24 2013.*

[2] A. Mehta, A. Saberi, U. Vazirani, and V. Vazirani. Adwords and generalized online matching. *Journal of the ACM, vol. 54, no. 5, p. 22*, 2007.

[3] R. Karp, U. Vazirani, and V. Vazirani. An optimal algorithm for online bipartite matching. *Proceedings of the 22nd Annual ACM Symposium on Theory of Computing, pp. 352358*, 1990.

[4] N. Buchbinder, K. Jain, and J. S. Naor. Online primaldual algorithms for maximizing ad-auctions revenue. *Proceedings of the 15th Annual European Symposium, pp. 253264*, 2007.

[5] S. Lahaie and D. Pennock. Revenue analysis of a family of ranking rules for keyword auctions. *Proceedings of the 8th ACM conference on Electronic commerce*, 2007.

[6] B. Tan and R. Srikant. Online advertisement, optimization and stochastic networks. *50th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC) Orlando, FL, USA*, December 2011.

[7] C. Karande, A. Mehta, and R. Sikant. Optimizing budget constrained spend in search advertising. *Proceedings of the sixth ACM international conference on Web search and data mining*, 2013.

[8] M. Cary, A. Das, B. Edelman, K. Heimerl I. Giotis, A. Karlin, C. Mathieu, and M. Schwarz. Greedy bidding strategies for keyword auctions. *Proceedings of the 8th ACM conference on Electronic commerce, pages 262271*, 2007.

[9] K. Amin, M. Kearns, P. Key, and A. Schwaighofer. Budget optimization for sponsored search: Censored learning in mdps. *Proceedings of the Twenty-Eighth Conference: Uncertainty in Artificial Intelligence*, 2012.

[10] R. Gummadi, P. Key, and A. Proutiere. Repeated auctions under budget constraints: Optimal bidding strategies and equilibria. *SSRN working draft*, 2012.

[11] C. Jiang, C. Beck, and R. Srikant. Bidding with limited statistical knowledge in online auctions. *W-PIN+NetEcon: The joint Workshop on Pricing and Incentives in Networks and Systems*, June 2013.

[12] Google adwords. *http://www.google.com/ads/.*

[13] Bing ads. *https://secure.bingads.microsoft.com/.*

[14] L. Georgiadis, M. J. Neely, and L. Tassiulas. *Resource Allocation and Cross-Layer Control in Wireless Networks.* Foundations and Trends in Networking Vol. 1, no. 1, pp. 1-144, 2006.

[15] L. Huang, S. Moeller, M. J. Neely, and B. Krishnamachari. LIFO-backpressure achieves near optimal utility-delay tradeoff. *IEEE/ACM Transactions on Networking, Volume 21, Issue 3, Pages 831-844*, June 2013.

[16] M. J. Neely. *Stochastic Network Optimization with Application to Communication and Queueing Systems.* Morgan & Claypool, 2010.

[17] M. J. Neely. Asynchronous scheduling for energy optimality in systems with multiple servers. *Proceedings of 46th Annual Conference on Information Sciences and Systems (CISS) (Invited)*, March 2012.

[18] B. Edelman, M. Ostrovsky, and M. Schwarz. Internet advertising and the generalized second-price auction: Selling billions of dollars worth of keywords. *American Economic Review, v. 97(1)*, March 2007.

[19] N. Nisan, T. Roughgarden, E. Tardos, and V. Vazirani. *Algorithmic Game Theory.* 2007.

[20] L. Huang. Optimizing your online-advertisement asynchronously. *arXiv Technical Report*, 2014.

[21] S. Boyd and L. Vandenberghe. *Convex Optimization.* Cambridge University Press, 2004.

[22] R. Durrett. *Probability: Theory and Examples.* Duxbury Press; 3 edition, 2004.