

Budget Optimization for Online Advertising Campaigns with Carryover Effects

Nikolay Archak
New York University, Leonard
N. Stern School of Business
44 West 4th Street, 8-185
New York, NY, 10012
narchak@stern.nyu.edu

Vahab S. Mirrokni
Google Research
76 9th Ave
New York, NY 10011
mirrokni@google.com

S. Muthukrishnan
Google Research
76 9th Ave
New York, NY 10011
muthu@google.com

ABSTRACT

While it is relatively easy to start an online advertising campaign, proper allocation of the marketing budget is far from trivial. A major challenge faced by the marketers attempting to optimize their campaigns is in the sheer number of variables involved, the many individual decisions they make in fixing or changing these variables, and the nontrivial short and long-term interplay among these variables and decisions.

In this paper, we study interactions among individual advertising decisions using a Markov model of user behavior. We formulate the budget allocation task of an advertiser as a constrained optimal control problem for a Markov Decision Process (MDP). Using the theory of constrained MDPs, a simple LP algorithm yields the optimal solution. Our main result is that, under a reasonable assumption that online advertising has positive carryover effects on the propensity and the form of user interactions with the same advertiser in the future, there is a simple greedy algorithm for the budget allocation with the worst-case running time cubic in the number of model states (potential advertising keywords) and an efficient parallel implementation in a distributed computing framework like MapReduce. Using real-world anonymized datasets from sponsored search advertising campaigns of several advertisers, we evaluate performance of the proposed budget allocation algorithm, and show that the greedy algorithm performs well compared to the optimal LP solution on these datasets and that both show consistent 5-10% improvement in the expected revenue against the optimal baseline algorithm ignoring carryover effects.

Categories and Subject Descriptors

G.1.6 [Numerical Analysis]: Optimization; H.4.m [Information Systems]: Miscellaneous; J.4 [Social and Behavioral Sciences]: Economics

General Terms

Algorithms, Measurement, Economics

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

Copyright 20XX ACM X-XXXXX-XX-X/XX/XX ...\$10.00.

Keywords

ad auctions, budget optimization, online advertising, Markov Decision Processes, sponsored search

1. INTRODUCTION

The Internet has become a major advertising medium, with billions of dollars at stake [26]. It has made it relatively easy even for small advertisers to quickly set up campaigns, track expenses, monitor effectiveness of the campaigns, and tinker with campaign parameters. Nonetheless, proper allocation of the marketing budget is far from trivial. A major challenge faced by the marketers attempting to optimize their campaigns is in the sheer number of variables they can possibly change. Even within a single advertising channel such as sponsored search ads on a particular search engine, the advertiser can optimize by reallocating the budget across different keywords, choosing a particular bidding strategy to use within a single ad auction, deciding on the daily advertising budget or what demographics of users to target. Each of these tasks can be solved reasonably well when considered as a standalone optimization problem, yet one can only wonder what fraction of social surplus (and advertising revenues) is lost by ignoring sophisticated dependencies and interaction patterns between individual optimization tasks, such as long-term effects of ads interacting with other ads.

In this paper, we study interactions among individual advertising decisions using a Markov model of user behavior, and develop optimization algorithms for budget allocation in this context. In particular, we focus on a potential *positive carryover effect* that online advertising has on the propensity and the form of user interactions with an advertiser in the future, and develop improved algorithms for the problem in this setting. We clarify these ideas on a simple scenario from the sponsored search area.

EXAMPLE 1. A number of competing retailers are selling a single good with a certain brand name online. Every retailer has a choice of advertising only on the retailer specific keywords like the retailer's name or advertising on both the retailer specific keywords and the brand name of the good they sell. In this scenario, most of the users potentially interested in buying the good are initially uninformed about individual retailers' existence and therefore search directly with the brand name of the good. As the good is relatively expensive, they do not buy it from the first retailer found, instead clicking on multiple ads and comparing numerous offers. Once decided on the best offer, they often search with the retailer's name directly, proceed to the retailer's website and convert, i.e., make a purchase. Furthermore, a fraction of

the converted users may become loyal customers that in the future skip the comparison shopping phase and go to the retailer’s website directly without performing any brand related searches. Important property of this example is that analyzing profitability of retailer-specific keywords and brand-specific keywords separately improperly captures the influence of both on the retailer’s revenue. Indeed, individual analysis in our example would suggest that brand-specific keywords provide significantly worse return-on-investment (ROI) than retailer-specific keywords due to both high CPC¹ values (heavy competition with other retailers) and low conversion rates (a lot of users clicking on multiple ads before converting).² Yet it would not be wise (and many advertisers know that) for the retailer to significantly cut spending on the brand-specific keywords as it is likely to reduce inflow of users to the retailer-specific keywords as well. One can say that there is a carryover from advertising on the brand-specific keywords to the ROI of advertising on the retailer-specific keywords. □

The above was only an example scenario and we emphasize that the model of carryover that we present in this paper is not restricted to only capture interactions between brand-specific and retailer-specific keywords, nor is it restricted to the domain of the sponsored search. Motivated by Markov models of user browsing behavior [25, 6], in particular our previous study on mining advertiser-specific user behavior in sponsored search auctions [3], we model users using a Markov chain and advertising as not only affecting the current user action but also the future actions (through changing the state transition probabilities). Our contributions are as follows:

- *(Problem)* In the Markovian user model, we formulate the budget allocation task of an advertiser as a constrained optimal control problem for a Markov Decision Process (MDP).
- *(Algorithm)* Using well-developed theory of constrained MDPs [2], we show that a simple LP algorithm yields the optimal policy. As the main contribution, we show that, under a reasonable assumption on the structure of carryover effects (see Section 5), there is a faster greedy algorithm for the optimal solution of the problem with the worst-case running time cubic in the number of model states (potential advertising keywords). This greedy algorithm is inspired by the Lagrangian relaxation of the optimization problem which is solvable using a combinatorial greedy algorithm in the presence of positive carryover effects. A major advantage of this algorithm is that it can be implemented efficiently in parallel using a distributed computing framework like MapReduce.
- *(Empirical Study)* Using real-world anonymized datasets from sponsored search advertising campaigns of several advertisers, we show that our greedy algorithm performs as well as the optimal LP solution, thus justifying our carryover assumption under which we can prove the optimality of our greedy algorithm. Furthermore, our budget allocation algorithm shows 5-10% improvement in revenues

¹cost per click

²This is only a hypothetical scenario and its conclusions might not generalize to all settings. There are empirical findings that suggest that the presence of retailer-specific information in the keyword increases click-through rates, and the presence of brand-specific information in the keyword increases conversion rates [15].

against the baseline, consistent across a wide range of different settings and budget constraints.

While budget optimization problems have been studied previously in sponsored search, even in the setting of possible externalities, our paper is the first to consider the long term impacts of different ad instances on each other.

2. RELATED WORK

Advertising carryover in marketing refers to the well-known phenomenon that advertising messages affect consumers long after the initial exposure. Carryover effects have been extensively studied in marketing literature [8, 5], including online settings [28]. The exact mechanism by which carryover works is often unspecified and the effect itself is usually modeled simply by assuming that a certain fraction of the advertising effects in the current period is retained in the next period. In our paper, we model carryover at the level of individual advertising decisions within the campaign. For instance, hypothetically, the decision of JetBlue to advertise on “cheap tickets” keyword may have carryover effect on the number of users that issue search queries with the airline’s brand name in the future.

Carryover effects in our model can be thought of as a type of positive externality. There has been some work on externalities in the sponsored search literature. Ghosh and Sayedi [17] consider negative externalities that online ads impose on each other if shown together. The negative externality in their model comes not only from the fact that displaying multiple ads decreases the amount of attention a single ad gets from a user, but also from associated reduction in conversion rates: if a user notices or clicks on an ad, he may not convert on it, but instead convert on a competing advertisement (similar to the comparison shopping behavior from Example 1). They find that because the value per click of an advertiser is no longer one-dimensional and depends on what other ads are displayed on the same page, the GSP mechanism is not adequately expressive anymore. Similar models of negative externalities were considered in [21, 4, 1]. Gomes, et al. [18] use impression and click data from Microsoft Live to show that such externalities are indeed statistically and economically significant. Another potential source of negative externalities is the “broad match” functionality which allows for imprecise match between the keyword the advertiser is bidding on and the user query; because bids on multiple keywords may match the same query, the advertiser may, under some circumstances, compete with oneself [13].

In contrast to the prior stream of research that studied negative externalities that competing ads impose on each other, this paper instead focuses on positive externalities in which ads of an advertiser on multiple keywords reinforce each other. Certain empirical support for the presence of positive externalities in sponsored search can be found in [22]: a randomized controlled experiment, performed in cooperation between Yahoo! and a major retailer, found that the online advertising campaign had substantial positive impact not only on the users who clicked on the ads but also on those who merely viewed them. In another study, comScore [9] reported an incremental lift of 27% in the online sales after the initial exposure to an online ad, as well as lift in other important online behaviors, such as the brand site visitation and the trademark searches. Ghose and Yang [16] report positive interdependency between paid and organic search results: the presence of organic listings is associated with a higher probability of click-throughs on paid ads, and vice-versa.

Our approach is based on a Markov model of users in which

they follow a Markov random walk that can be influenced by the advertising activities. Markov models of user behavior on the Web are not novel and can be traced as far as the original PageRank paper [25]. Charikar, et al. [7] pushed this idea one step further by suggesting that online advertising can be thought of as a problem of choosing the optimal control policy for targeting heterogeneous population of users with behaviors described by Markov chains. Our model differs essentially from [7] on several fundamental assumptions. [7] assumed that properly targeted user always converts and concentrated purely on the problem of optimally targeting the heterogeneous flow of users. We instead focus on a homogeneous population but allow the advertising effect to be non-deterministic. In contrast with [7], we provide evaluation of our model on real world advertising data. Additional empirical evidence that Markov models provide useful abstraction of user behavior from the advertiser’s perspective can also be found in [3].

In the world populated by Markov users, we consider the standard *budgeted campaign optimization problem* [14, 24, 11]: find an optimal bidding policy to maximize the number of user conversions subject to the budget constraint for the expected advertising cost. Our approach allows us to apply machinery from the familiar field of constrained MDPs, in particular reduce the budget optimization problem to a regular LP (an excellent review of constrained MDPs can be found in [2]). As the main contribution of the paper, under assumption of positive carryover effects we provide a simple, fast greedy algorithm for this problem based on the ideas of Lagrangian relaxation. While Lagrangian relaxation of LP formulations has been used to design approximation algorithms for various combinatorial optimization problems like the k -median and the facility location [20, 19], to the best of our knowledge, our result does not follow from any of these and relies on the special structural properties of the budget optimization problem.

3. OUR MODEL AND PROBLEM

The notation below is chosen to be consistent with [2], except that we consider the problem of maximizing the long-term total reward (conversion probability) while [2] considered the problem of minimizing the long-term total cost.

Let X be a finite state space representing possible user *states*. In one interpretation, the state can capture the last query issued by the user. For any $x \in X$, let $A(x)$ represents the finite set of possible *actions* (advertising levels) in state x . For instance, $A(x)$ can be {advertise, do not advertise} but one can also consider more sophisticated possibilities with different levels of advertising, for instance, one can think of different slots on the search results page as possible advertising levels. Without loss of generality, we can always assume a common set of advertising levels $A(x) \equiv A$ available in all states. The user randomly “jumps” between states with transition probabilities depending on the level of the advertising the user is exposed to. Let \mathcal{P}_{xay} be the probability of moving from state x to state y if advertising level $a \in A$ is chosen. Next, let $d(x, a) \geq 0$ be the immediate monetary cost of advertising at level a in state x . This cost will relate to the budget constraint (V) for our optimization problem.

We define three special states in the system: $x_c \in X$ representing the conversion state, $x_n \in X$ representing the non-conversion state, and x_f representing the final state. The final state x_f is absorbing. All transitions from the non-conversion state x_n and the conversion state x_c lead to the final state x_f . The initial flow of users to the system is given by measure $\beta(x)$ and the advertisers’ optimization problem is to maximize the expected number of

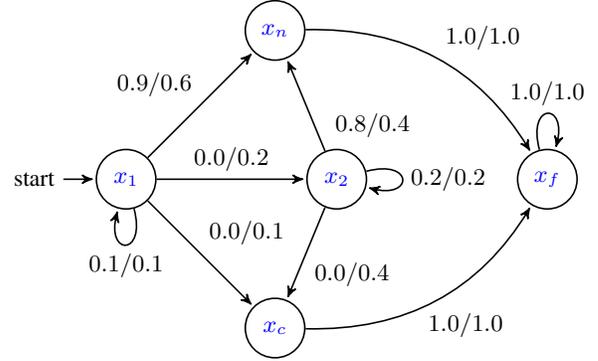


Figure 1: Sample Markov model of user behavior. The first value on the edge represents the transition probability if the “do not advertise” action is chosen; the second value on the edge represents the transition probability if the “advertise” action is chosen.

converted users subject to the budget constraint. Without loss of generality, we can assume that β is normalized to 1 and therefore represents a probability measure. In such normalization, V will represent a per-user budget constraint.

EXAMPLE 2. In a simple scenario of Example 1, we can think of users as following a Markov model with two major states: x_1 representing a search on a brand keyword and x_2 representing a search on a retailer keyword. As users are initially uninformed about retailer’s existence, they always start in state x_1 . The retailer can choose to advertise (a_1) or not to advertise (a_0) to a user based on the current user’s state. We will always assume that not advertising is costless; in the scope of this example only, the advertising cost in both states will be taken to be 1.0. A hypothetical Markov process for this scenario is shown in Figure 1. Note that, in our example, if the retailer chooses not to advertise to users searching for the brand name (state x_1), then users will have zero probability of transitioning to the state x_2 as they will never learn about retailer’s existence. Thus, although conversion rate in state x_2 is four times higher than in state x_1 (0.4 against 0.1), advertising in the second state has no value, unless one advertises in the first state as well.

We can recast the optimization problem as a particular case of constrained MDPs by defining the reward function that we are trying to maximize as $r(x, a) \equiv C$ when $x = x_c$ and $r(x, a) \equiv 0$ otherwise (i.e., we get a reward of C in the conversion state and zero everywhere else). We assume that the Markov process is absorbing, i.e. sooner or later we will end up in the final state in which we accumulate no reward and pay no cost, thus the optimization problem is well-defined. We formalize this as follows.

For $t = 1, \dots, \infty$, let H_t be the set of all possible user histories of length t . Every element $h_t \in H_t$ is a history of states and chosen actions until time t , i.e. $h_t = (x_1, a_1, x_2, a_2, \dots, x_t)$ (note that the advertising exposure at time t is not included). A general *policy* u is a collection of functions

$$u_t : H_t \rightarrow \Delta(A),$$

where $\Delta(A)$ represents the set of probability distributions over A (policies can be randomized). Note that the general policy allows for the targeted advertising, i.e., choosing the advertising level for the user based on the complete history of the prior user searches and ads the user was exposed to.

DEFINITION 1. For every distribution over initial states β and a policy u , there is a unique measure on the space of trajectories H_∞ . We can use P_β^u to denote this measure. Moreover, define

$$p_\beta^u(t; x, a) = P_\beta^u(x_t = x, a_t = a),$$

i.e., $p_\beta^u(t; x, a)$ is the probability of observing the state x and the advertising level a at the step t of the process when following the policy u . Next, define the expected total reward and the expected total cost for a policy u as

$$R(\beta, u) = \sum_{t=1}^{\infty} \sum_{X, A} p_\beta^u(t; x, a) r(x, a)$$

and

$$D(\beta, u) = \sum_{t=1}^{\infty} \sum_{X, A} p_\beta^u(t; x, a) d(x, a).$$

respectively. \square

Note that both are well-defined as we assume that the MDP is absorbing.

The budget optimization problem we face (for a single user) is simply

$$\begin{aligned} \max_{u \in U} R(\beta, u) & \quad \text{[P1]} \\ \text{s.t. } D(\beta, u) & \leq V \end{aligned}$$

where U is the set of policies of interest.

Special Classes of Policies. There are three classes of policies of our interest:

- In *Markov policies*, u_t depends only on x_t , that is, we target users based only on their current state and the amount of time they spent in the system.
- In the special case of *stationary policies*, u_t does not depend on t , that is, we target users based on their current state only.
- Further special are *stationary deterministic policies*, for which the advertising level is chosen in each state deterministically. That is, we target users based on their current state only and all users in the same state are exposed to the same advertising level.

4. CLASSIC RESULTS FOR CONSTRAINED MARKOV DECISION PROCESSES

Below is a summary of well-known results for constrained MDPs that apply to our model. The proofs are in [2].

Fact 1. It is sufficient to restrict consideration to Markov policies only (see Theorem 2.1 of [2]) as for any general policy u , there exists some other Markov policy v such that $p_\beta^u(t; \cdot, \cdot) \equiv p_\beta^v(t; \cdot, \cdot)$.

Fact 2. Let $X' = X \setminus \{x_f\}$. An *occupation measure* is a “visit count” measure over the set of states and advertising levels ($\mu \in M(X' \times A)$) achievable by some Markov policy u :

$$\mu(x, a) = \sum_{t=1}^{\infty} p_\beta^u(t; x, a).$$

Let $L(\beta)$ be the set of all occupation measures, $L(\beta)_S$ be the set of all occupation measures achievable with a stationary policy and $L(\beta)_D$ be the set of all occupation measures achievable with a stationary deterministic policy. Theorem 3.2 from [2] gives

characterization of the set of all occupation measures. It says that $L(\beta) = L(\beta)_S = \text{co}L(\beta)_D$ (convex hull). Moreover, it is equal to $Q(\beta)$, where $Q(\beta)$ is the set of all non-negative finite measures ρ on $X' \times A$ such that

$$\forall x \in X' \sum_{y \in X'} \sum_{a \in A} \rho(y, a) (\delta_x(y) - \mathcal{P}_{yax}) = \beta(x) \quad (1)$$

Note that Equation 1 is the basic “conservation of flow” statement, thus the result can be interpreted as “any measure satisfying the set of conservation of flow constraints is achievable with some stationary policy” (the reverse is obviously true as well).

Fact 3. The previous result means that we can only look for stationary policies or, even better, we can look for the solution in form of an occupation measure. Theorem 3.5 from [2] shows that there is one to one equivalence between feasible (and optimal) solutions of **P1** and feasible (and optimal) solutions of the following linear program:

$$\begin{aligned} \max_{\rho} \sum_{x \in X'} \sum_{a \in A} r(x, a) \rho(x, a) & \quad \text{[P2]} \\ \text{s.t. } \sum_{x \in X'} \sum_{a \in A} d(x, a) \rho(x, a) & \leq V \\ \sum_{y \in X'} \sum_{a \in A} \rho(y, a) (\delta_x(y) - \mathcal{P}_{yax}) & = \beta(x) \quad \forall x \in X' \\ \rho(x, a) & \geq 0 \quad \forall x \in X', a \in A. \end{aligned}$$

In particular, if ρ^* is the optimal solution of **P2**, then the stationary policy u^* choosing the advertising level a with probability of $\frac{\rho^*(x, a)}{\sum_b \rho^*(x, b)}$ is the optimal randomized stationary policy (one can choose any advertising level if the denominator is zero).

Note that the linear program **P2** has $|X'| + 1$ constraints (the budget constraint and $|X'|$ consistency constraints) in addition to the non-negativity constraints. Thus, one can always find the optimal solution in which at most $|X'| + 1$ $\rho(y, a)$ values are positive. That implies that there is always an optimal advertising strategy with randomization in at most one state.

EXAMPLE 3. For the Markov process shown in Figure 1 (Example 2), the optimization problem **P2** can be reduced to³

$$\begin{aligned} \max_{\rho} C(\rho(x_c, a_0) + \rho(x_c, a_1)) \\ \text{s.t. } \rho(x_1, a_1) + \rho(x_2, a_1) & \leq V \\ 0.9(\rho(x_1, a_0) + \rho(x_1, a_1)) & = 1.0 \\ 0.8(\rho(x_2, a_0) + \rho(x_2, a_1)) & = 0.2\rho(x_1, a_1) \\ (\rho(x_c, a_0) + \rho(x_c, a_1)) & = 0.1\rho(x_1, a_1) + 0.4\rho(x_2, a_1) \\ \rho(x, a) & \geq 0. \end{aligned}$$

For a particular value of the budget constraint, this program can be solved to obtain the occupation measure for the optimal policy. For example, if $V = 1.0$, the optimal solution of **P2** is $\rho(x_1, a_0) = \frac{14}{45}$, $\rho(x_1, a_1) = 0.8$, $\rho(x_2, a_0) = 0$ and $\rho(x_2, a_1) = 0.2$. It follows that the optimal policy will always advertise to users in state x_2 and advertise to users in state x_1 with probability of 0.72⁴.

³Obviously, we are always indifferent whether to advertise or not to advertise once the user has already converted. Thus, sum of $\rho(x_c, a_0)$ and $\rho(x_c, a_1)$ can always be collapsed to a single variable (probability of conversion). For consistency of notation only, we keep these two variables separately.

⁴ $\frac{0.8}{0.8 + \frac{14}{45}}$

Fact 4. In the following, it will also be useful to consider the dual program of **P2**:

$$\begin{aligned} \min_{\pi, \lambda} \quad & \sum_{x \in X'} \beta(x) \pi(x) + \lambda V & \text{[P3]} \\ \text{s.t.} \quad & \lambda \geq 0 \\ & \pi(x) \geq r(x, a) - \lambda d(x, a) + \sum_{y \in X'} \mathcal{P}_{xay} \pi(y) \\ & \forall x \in X', a \in A \end{aligned}$$

Here λ is the Lagrange multiplier for the budget constraint and, for any fixed value of λ , $\pi(x)$ can be thought of as the optimal value function in the Markov model M_λ with adjusted rewards $r_\lambda(x) = r(x, a) - \lambda d(x, a)$. This intuition is captured by the following LP for a fixed λ :

$$\begin{aligned} \min_{\pi_\lambda} \quad & \sum_{x \in X'} \beta(x) \pi(x) & \text{[P3}(\lambda)] \\ \text{s.t.} \quad & \pi_\lambda(x) \geq r_\lambda(x, a) + \sum_{y \in X'} \mathcal{P}_{xay} \pi_\lambda(y) \\ & \forall x \in X', a \in A \end{aligned}$$

Because the value of λ is fixed, **[P3](\lambda)** is a classic infinite-horizon DP problem on a graph M_λ with rewards $r_\lambda(x, a)$, therefore it has a uniformly optimal stationary dual policy, which in every state x chooses the advertising level $a(x)$ deterministically and does not depend on the distribution of initial states β .

5. BUDGET OPTIMIZATION WITH POSITIVE CARRYOVER EFFECTS

The previous section shows that the budget optimization problem in Markovian world can be cast a simple linear program **P2** with $|X'| \times |A|$ variables and $|X'| + 1$ constraints. In real world online advertising settings, in particular, in sponsored search, $|X'|$ represents the number of feasible keywords to advertise on and therefore can be as large as tens of thousands for a single advertiser. Number of advertising levels can be in the order of ten (different slots) or more. Considering the fact that the constraint matrix is not sparse, the direct LP approach presents significant practical computational challenges. In this section, we identify the structure in the problem and use that to design a simpler greedy algorithm which proceeds under assumption that the advertising carryover effects are positive, which is realistic. The algorithm is guaranteed to find the optimal solution of **P2** with the worst-case running time of $|X'|^3 \times |A|^2$ under this assumption. As the experimental section shows, the suggested algorithm performs very well in the real world settings even if the underlying assumptions are violated.

First, we impose that the set of advertising levels A is totally ordered $a_1 \preceq a_2 \preceq \dots \preceq a_k$, with interpretation that if $a_i \prec a_j$ then a_j represents a more intense level of advertising than a_i . Next, we assume that the Markov user model satisfies the following conditions which are realistic (our empirical study will not make such assumptions):

- More advertising never hurts (Postive Carryover):

$$\forall x \in X', y \in X' \setminus \{x_n\}, a \preceq b \quad \mathcal{P}_{xay} \leq \mathcal{P}_{xby}$$

- More advertising is more expensive:⁵

$$\forall x \in X', a \preceq b \quad d(x, a) \leq d(x, b)$$

⁵This assumption is not essential and can be relaxed. Indeed, if there are two advertising levels a and b such that $a \preceq b$ but

- Not advertising costs nothing: $d(x, a_1) \equiv 0$, i.e., we assume that the advertiser can always opt out of advertising in any state at no extra cost.

Now, for any fixed $\lambda \geq 0$ consider the optimization problem **P3**(λ): and its dual **P4**(λ):

$$\begin{aligned} \min_{\rho} \quad & \sum_{x \in X'} \sum_{a \in A} (r(x, a) - \lambda d(x, a)) \rho(x, a) & \text{[P4}(\lambda)] \\ \text{s.t.} \quad & \forall x \in X' \sum_{y \in X'} \sum_{a \in A} \rho(y, a) (\delta_x(y) - \mathcal{P}_{yax}) = \beta(x) \\ & \forall x \in X', a \in A \quad \rho(x, a) \geq 0. \end{aligned}$$

We emphasize that because **P3**(λ) is a classic infinite-horizon DP problem on a graph, it has a uniformly optimal stationary policy. In case of $\lambda = 0$, this policy has a particularly simple structure due to positive externalities.

LEMMA 1 (SOLUTION OF UNCONSTRAINED PROBLEM). *For $\lambda = 0$ there is a uniformly optimal policy of **P3**(λ) in which we advertise with the highest possible intensity (a_k) in every state.*

PROOF. Consider a policy u which chooses a_k in every state. Let $\rho(x, a)$ be the occupation measure induced by u and π be the value function induced by u . Obviously, ρ represents a feasible solution of the primal optimization problem. Moreover, π represents a feasible solution of the dual optimization problem, because if

$$\pi(x) = r(x, a_k) + \sum_{y \in X'} \mathcal{P}_{x a_k y} \pi(y),$$

then

$$\forall i \quad \pi(x) \geq r(x, a_i) + \sum_{y \in X'} \mathcal{P}_{x a_i y} \pi(y).$$

Finally, both solutions give the same value thus both are optimal. \square

LEMMA 2 (MONOTONICITY OF DUAL VALUE FUNCTION). *Let $0 \leq \lambda_1 < \lambda_2$, u_1, u_2 be the corresponding uniformly optimal stationary policies and π_1, π_2 be the corresponding value functions for **P3**(λ). Then, $\forall x \in X' \quad \pi_1(x) \geq \pi_2(x)$.*

PROOF. Assume not and there exists x s.t. $\pi_1(x) < \pi_2(x)$. Because u_1 and u_2 are uniformly optimal policies, they are also optimal for the initial distribution $\hat{\beta} = \delta_x$. Now, π_1 is feasible for **P3**(λ_1), therefore (straightforward to check) it is also feasible for **P3**(λ_2). Because π_2 is optimal for **P3**(λ_2), it must be that

$$\sum_{y \in X'} \hat{\beta}(y) \pi_1(y) \leq \sum_{y \in X'} \hat{\beta}(y) \pi_2(y),$$

i.e., $\pi_1(x) \leq \pi_2(x)$. This is a contradiction. \square

LEMMA 3 (CONTINUITY OF DUAL VALUE FUNCTION). *Let $f_\beta(\lambda)$ be the value of the optimization problem **P3**(λ)⁶. $f_\beta(\lambda)$ is a continuous function of λ . In particular, taking $\beta \equiv \delta_x$, we obtain that $\pi_\lambda^*(x)$ is a continuous function of λ .*

PROOF. Just look at the dual **P4**(λ). \square

⁶subscript β is used to indicate the dependence on the initial distribution β

DEFINITION 2. For any $\lambda \geq 0$ and $x \in X'$, define the set of active advertising levels $\mathcal{A}(\lambda, x)$ as

$$\left\{ a \in A \text{ s. t. } \exists \pi^* \text{ uniformly optimal for } P3(\lambda) \text{ and } \pi^*(x) = r_\lambda(x, a) + \sum_{y \in X'} \mathcal{P}_{xa_2y} \pi^*(y) \right\}$$

Note that $\mathcal{A}(\lambda, x)$ is always non-empty.

DEFINITION 3. For any $\lambda \geq 0$ and $x \in X'$, define the lowest active advertising level $a_L(\lambda, x)$ and the highest active advertising level $a_H(\lambda, x)$ as

$$a_L(\lambda, x) = \min \mathcal{A}(\lambda, x),$$

$$a_H(\lambda, x) = \max \mathcal{A}(\lambda, x).$$

LEMMA 4 (MONOTONE SELECTION).

For any $x \in X$ and $0 \leq \lambda_1 < \lambda_2$, we have $a_L(\lambda_1, x) \succeq a_L(\lambda_2, x)$ and $a_H(\lambda_1, x) \succeq a_H(\lambda_2, x)$.

PROOF. Proof for a_H (proof for a_L uses similar argument in the reverse direction). Assume otherwise, i.e., $\exists a_1 = a_H(\lambda_1, x)$, $a_2 = a_H(\lambda_2, x)$ such that $a_1 \prec a_2$. Let π_1, π_2 be the corresponding value functions.

Consider a possible value function gain in state x from choosing advertising level a_1 instead of a_2 in both cases (we consider one time deviation only, i.e., assuming that we follow the old policies afterwards). For π_1 and λ_1 the gain is

$$G_1 = \lambda_1(d(x, a_2) - d(x, a_1)) - \sum_y (\mathcal{P}_{xa_2y} - \mathcal{P}_{xa_1y}) \pi_1(y).$$

For π_2 and λ_2 the gain is

$$G_2 = \lambda_2(d(x, a_2) - d(x, a_1)) - \sum_y (\mathcal{P}_{xa_2y} - \mathcal{P}_{xa_1y}) \pi_2(y).$$

Because $a_2 \succ a_1$, we have $d(x, a_2) \geq d(x, a_1)$ and $\mathcal{P}_{xa_2y} \geq \mathcal{P}_{xa_1y}$. Moreover, $\pi_1(y) \geq \pi_2(y)$, thus the second gain G_2 is at least as large as G_1 . But G_2 is non-positive because $a_2 \in \mathcal{A}(\lambda_2, x)$. It must be that G_1 is also non-positive, i.e., $a_2 \in \mathcal{A}(\lambda_1, x)$. This is a contradiction because a_1 is the "largest" such advertising level. \square

Note. Proof of Lemma 4 looks like a standard single-crossing argument that can be used to prove monotone selection theorems for supermodular and quasisupermodular functions. While the primal optimization problem can indeed be written as a supermodular function, the Lagrangian relaxation of the dual is not supermodular, nor quasisupermodular (we omit the counterexamples due to space limitation), therefore Lemma 4 doesn't seem to be a corollary of Topkis's theorem [27] or monotone selection theorems for quasisupermodular functions [23].

LEMMA 5 (STRUCTURE OF DUAL VALUE FUNCTION).

$f_\beta(\lambda)$ is a piecewise linear continuous function. Moreover, the slope of f_β at any particular λ is equal to $-\beta^T (I - \mathcal{P}_\lambda)^{-1} d_\lambda$, where β is the column vector of $\beta(x)$, d_λ is the column vector of $d(x, a_H(\lambda, x))$ and \mathcal{P}_λ is the matrix of $\mathcal{P}_{xa_H(\lambda, x)y}$.⁷

PROOF. To show that f is piecewise linear, note that there is only finite number of possible sets $\mathbb{A}_\lambda = \{(x, a_H(\lambda, x)) | x \in X'\}$. Moreover, for any $\lambda_1 < \lambda_2$, if $\mathbb{A}_{\lambda_1} = \mathbb{A}_{\lambda_2}$, then $\mathbb{A}_\lambda \equiv \mathbb{A}_\lambda$ for any $\lambda \in [\lambda_1, \lambda_2]$ (immediately follows from Lemma 4). Thus

⁷Note that there is an equivalent representation in which $d_\lambda = d(x, a_L(\lambda, x))$ and $\mathcal{P}_\lambda = \mathcal{P}_{xa_L(\lambda, x)y}$.

the real line can be split into a finite number of intervals on which \mathbb{A}_λ does not change.

Next, we can show left-continuity of \mathbb{A}_λ . Take an **increasing** sequence $\{\lambda_n\}$ converging to a certain value λ . By observation 6.3, it must be that $\mathbb{A}_{\lambda_n} \succeq \mathbb{A}_\lambda$. Moreover, sequence \mathbb{A}_{λ_n} is nonincreasing (w.r.t. ordering \succeq) thus it must converge to some $\mathbb{A} \succeq \mathbb{A}_\lambda$. To show the actual equality, assume otherwise, i.e, there is some $x \in X'$ such that $a_H(\lambda, x) = a_1$, however $a_H(\lambda_n, x) \equiv a_2 \succ a_1$ for any sufficiently large n . It must be that

$$\pi_{\lambda_n}^*(x) = r_{\lambda_n}(x, a) + \sum_{y \in X'} \mathcal{P}_{xa_2y} \pi_{\lambda_n}^*(y).$$

By going to the limit

$$\pi_\lambda^*(x) = r_\lambda(x, a) + \sum_{y \in X'} \mathcal{P}_{xa_2y} \pi_\lambda^*(y).$$

That means that $a_2 \in \mathcal{A}(\lambda, x)$ and thus we have a contradiction. Now, we know that \mathbb{A}_λ is continuous on intervals of the form $(\lambda_0 = 0, \lambda_1], (\lambda_1, \lambda_2], (\lambda_2, \lambda_3] \dots$

Finally, take any such interval $(\lambda_i, \lambda_{i+1}]$. Note that \mathcal{P}_λ and d_λ are constant within this interval. Moreover, by definition of \mathbb{A}_λ we have that

$$\pi_\lambda^* = \text{Const} - \lambda d_\lambda + \mathcal{P}_\lambda \pi_\lambda^*$$

or

$$\pi_\lambda^* = (I - \mathcal{P}_\lambda)^{-1} (\text{Const} - \lambda d_\lambda),$$

i.e. π_λ^* is linear on $(\lambda_i, \lambda_{i+1}]$ and

$$\frac{d}{d\lambda} \pi_\lambda^* = -(I - \mathcal{P}_\lambda)^{-1} d_\lambda. \quad \square$$

5.1 Greedy BO Algorithm

Lemma 5 suggests a simple greedy algorithm for determining all breakpoint values λ_i for the dual value function $f_\beta(\lambda)$. Algorithm 2 keeps the the current set of highest active advertising levels \mathbb{A}_{λ_i} as a part of its state. \mathbb{A}_{λ_i} is stored as a simple set of numbers $m(x)$ for every state x , representing that $\mathbb{A}_{\lambda_i} = \{(x, a_{m(x)}) | x \in X'\}$. At every step of the algorithm we choose one candidate node x^* . One way to define this node is to imagine that we freeze the current set of active advertising levels \mathbb{A}_{λ_i} and start gradually increasing the value of λ . The first node, for which it will be locally optimal to decrease the advertising level $m(x)$, is the node x^* , the new value λ_{i+1} at which that happens is given by $\lambda_{i+1} = \lambda_i + \delta^*$ and the new advertising level at x^* will be m^* .

THEOREM 1 (GREEDY BO ALGORITHM). *Algorithm 2 correctly constructs the dual value function $f_\beta(\lambda)$.*

PROOF. Obviously, Algorithm 2 will finish (the vector m always decreases by at least 1 in one of the components). The correctness of the algorithm can be proved by induction. At $\lambda_0 = 0$, the result is correct by Lemma 1. Assume now that the claim holds up to and including λ_i . From Lemma 5, we know that $f_\beta(\lambda)$ is a piecewise linear continuous function. Let $\hat{\lambda}_{i+1}$ be its next breakpoint.

Observation 1: $\hat{\lambda}_{i+1}$ is at least as large as λ_{i+1} . Assume not and $\hat{\lambda}_{i+1} < \lambda_{i+1}$. The proof of Lemma 5 shows that there must be some x and advertising level b such that $b = a_H(\lambda, x)$ for $\lambda \in (\lambda_i, \hat{\lambda}_{i+1}]$. If $a_{m(x)} \succ b$, then, by continuity and monotonicity it must be possible to decrease $a_{m(x)}$ to b without changing the value function π . If that is the case, then the **Candidate**

Selection() method should return $\delta^* = 0$ at the step i and therefore $\lambda_{i+1} = \lambda_i$. That is a contradiction with assumption that $\hat{\lambda}_{i+1} < \lambda_{i+1}$, so we can assume that $a_{m(x)} = b$. If so, then

$$\pi_{\bar{\lambda}}^*(x) = r_{\lambda}(x, a_{m(x)}) + \sum_{y \in X'} \mathcal{P}_{x a_{m(x)} y} \pi_{\bar{\lambda}}^*(y)$$

on $[\lambda_i, \hat{\lambda}_{i+1}]$. Consider $\bar{\lambda} = \hat{\lambda}_{i+1} + \varepsilon$. By construction $a_{m(x)} \neq a_H(\bar{\lambda}, x)$ and therefore $a_{m(x)} \succ a_H(\bar{\lambda}, x) = a_t$. By continuity,

$$r_{\hat{\lambda}_{i+1}}(x, a_{m(x)}) + \sum_{y \in X'} \mathcal{P}_{x a_{m(x)} y} \pi_{\hat{\lambda}_{i+1}}^*(y)$$

must be equal to

$$r_{\hat{\lambda}_{i+1}}(x, a_t) + \sum_{y \in X'} \mathcal{P}_{x a_t y} \pi_{\hat{\lambda}_{i+1}}^*(y).$$

Easy to check that if this is the case then the triple $(x, \hat{\lambda}_{i+1} - \lambda_i, t)$ should have been returned by the **CandidateSelection()** method. That didn't happen and so we have a contradiction and Observation 1 is proved.

Observation 2: On interval $[\lambda_i, \lambda_{i+1}]$, the value function π grows linearly as $\pi_i + (\lambda - \lambda_i) d\pi_i$.

This observation follows immediately from Lemma 5 and the fact that $\hat{\lambda}_{i+1} \geq \lambda_{i+1}$. Thus, on interval $[\lambda_i, \lambda_{i+1}]$, Algorithm 2 properly reconstructs the function $f_{\beta}(\lambda)$. The proof is complete.

We note that there is an alternative version of Algorithm 2, in which the optimization starts $\lambda_0 = +\infty$ and active advertising level equal to a_0 in every node and we reconstruct the value function $f_{\beta}(\lambda)$ by gradually decreasing λ . \square

Algorithm 1 Candidate node selection for BO algorithm.

Candidate Selection():

```

 $\delta^* \leftarrow +\infty$ 
 $m^* \leftarrow +\infty$ 
for every  $x \in X'$  such that  $m(x) > 1$  do
  for every  $m \in [1, m(x) - 1]$  do
     $\Delta \leftarrow d(x, a_{m(x)}) - d(x, a_m)$ 
     $\delta \leftarrow -\frac{\sum_{y \in X'} (\mathcal{P}_{x a_{m(x)} y} - \mathcal{P}_{x a_m y}) \pi_i(y) - \lambda_i \Delta}{\sum_{y \in X'} (\mathcal{P}_{x a_{m(x)} y} - \mathcal{P}_{x a_m y}) d\pi_i(y) - \Delta}$ 
    if  $\delta < \delta^*$  then
       $\delta^* \leftarrow \delta$ 
       $m^* \leftarrow m$ 
       $x^* \leftarrow x$ 
    end if
  end for
end for
Returns  $(x^*, \delta^*, m^*)$ 

```

5.2 Improved Greedy BO Algorithm

Number of iterations of Algorithm 2 is bounded by $|X| \times |A|$. The most expensive operation inside a single iteration is solving a linear system with $|X|$ unknowns and $|X|$ variables. This can be done in $O(|X|^3)$ operations in practice or in $O(|X|^{2.376})$ asymptotically [10]. Fortunately, we can significantly improve the performance of the algorithm by noting that it proceeds one variable at a time, always adjusting advertising level in a single state only. Thus, we do not really need to solve the system $d\pi_{i+1} \leftarrow -(I - P_{i+1})^{-1} d_{i+1}$ from the scratch each time. Instead, the algorithm can keep an LU decomposition of the matrix $I - P_i$, updating it in every step. Because only a single row is

Algorithm 2 Greedy algorithm for BO with positive carryover effects.

Initialization:

```

 $i \leftarrow 0$  (state number)
 $\lambda_0 \leftarrow 0$  (initial breakpoint)
 $\forall x \in X'$   $m(x) \leftarrow k$  (set current advertising level to the highest possible)
 $P_0 \leftarrow \mathcal{P}_0$  (set transition probabilities according to the current advertising levels)
 $d_0 \leftarrow d(\cdot, a(k))$  (set advertising costs according to the current advertising levels)
 $\pi_0 \leftarrow (I - P_0)^{-1} r$  (find the optimal value function for the initial advertising levels; this is equivalent to solving a linear system of equations)
 $d\pi_0 \leftarrow \frac{d}{d\lambda} \pi_0^*(y) = -(I - P_0)^{-1} d_0$  (find the derivative of the optimal value function; this is equivalent to solving a linear system of equations)

```

Main Loop():

```

while  $\exists x \in X'$   $m(x) > 1$  do
   $(x^*, \delta^*, m^*) \leftarrow$  Candidate Selection()
   $m(x^*) \leftarrow m^*$ 
   $P_{i+1} \leftarrow P_i$  with row for  $x^*$  replaced by  $\mathcal{P}_{x^* a_{m^*} y}$ 
   $d_{i+1} \leftarrow d_i$  with value for  $x^*$  replaced by  $d(x^*, a_{m^*})$ 
   $\pi_{i+1} \leftarrow \pi_i + \delta^* d\pi_i$ 
   $d\pi_{i+1} \leftarrow -(I - P_{i+1})^{-1} d_{i+1}$ 
   $\lambda_{i+1} \leftarrow \lambda_i + \delta^*$ 
   $i \leftarrow i + 1$ 
end while

```

replaced in the matrix, updating the LU decomposition can be trivially done in a quadratic time by solving a system of equations with a triangular matrix. That results in the $O(|X|^2 \times |A|)$ worst-case performance of the inner loop and $O(|X|^3 \times |A|^2)$ worst-case performance of the whole algorithm assuming a sequential processing model. The improved version of the algorithm is given by Algorithm 3.

5.3 Parallel Implementation of Greedy BO Algorithm

The most interesting property of Algorithm 3 is that it supports an efficient parallel implementation using a distributed programming framework like MapReduce [12]. This might be an important advantage for solving large-scale advertising campaigns with several thousands of keywords. This is in contrast to the original LP program **P2**, which is not a packing-covering linear program and, therefore, we are not aware of any distributed or parallel algorithm to solve it. Below, we give a brief description of the idea behind this parallel implementation.

The **Candidate Selection()** function can be parallelized to run on $|X|$ machines simply by distributing every iteration of the outer loop (**for** every $x \in X'$) to a separate machine and aggregating the results afterwards. Similarly, solution of a system of linear inequalities with a triangular matrix can be done in $|X|$ time on $|X|$ machines. Thus, we state that in a parallel processing framework with $|X|$ machines, Algorithm 3 worst-case performance is $O(|X|^2 \times |A|^2)$ plus the time needed to perform LU decomposition of the matrix $I - P_0$ in the initialization step. Details of the implementation are beyond the scope of this paper.

Algorithm 3 Improved Greedy algorithm for BO with positive carryover effects.

Initialization:

$i \leftarrow 0$
 $\lambda_0 \leftarrow 0$
 $\forall x \in X' \ m(x) \leftarrow k$
 $P_0 \leftarrow \mathcal{P}_0$
 $d_0 \leftarrow d(\cdot, a(k))$
 $(L_0, U) \leftarrow \text{LU decomposition of } (I - P_0)$
 $\pi_i \leftarrow U^{-1}L_0^{-1}r$
 $d\pi_i \leftarrow \frac{d}{dx}\pi_i^*(y) = -U^{-1}L_0^{-1}d_i$

Main Loop:

while $\exists x \in X' \ m(x) > 1$ **do**
 $(x^*, \delta^*, m^*) \leftarrow \text{Candidate Selection}()$
 $m(x^*) \leftarrow m^*$
 $P_{i+1} \leftarrow P_i$ with row for x^* replaced by $\mathcal{P}_{x^* a_{m^*} y}$
 $L_{i+1} \leftarrow L_i$ with row for x^* replaced by $U^{-1}z^i$ where z is the row for x^* from $I - P_{i+1}$
 $d_{i+1} \leftarrow d_i$ with value for x^* replaced by $d(x^*, a_{m^*})$
 $\pi_{i+1} \leftarrow \pi_i + \delta^* d\pi_i$
 $d\pi_{i+1} \leftarrow -U^{-1}L_{i+1}^{-1}d_{i+1}$
 $\lambda_{i+1} \leftarrow \lambda_i + \delta^*$
 $i \leftarrow i + 1$
end while

6. EVALUATION

We performed evaluation of our budget optimization algorithms on nine real world datasets containing data from nine different sponsored search campaigns. All datasets were advertiser-specific and included only user activities (such as ad clicks) related to a single search campaign of a single advertiser. The dataset was collected at user level and contained information on a random sample of users who converted with the advertiser within a period of two weeks in December 2009. For every anonymous user, the dataset recorded ad clicks of this user before the conversion. Every ad click event had associated timestamp and the keyword the user query was matched with. In this data set, we do not observe the events in which the users did not click on the ad. Moreover, since we focus on advertiser-specific information, user searches for which the advertiser’s ad was not shown, such as searches with irrelevant search queries, keywords on which the advertiser bid too low, or keywords for which the advertiser was excluded from the auction due to a daily budget constraint, are *not* included in our dataset. While such extra data might be available in some form to the search engine, due to several privacy and competition issues, it would not be reported to the advertisers, therefore we intentionally focus on the restricted advertiser-specific data described above.

In addition to the above datasets, we compiled cost information for all keywords from our sample. To simplify the experiments, we used average CPC (cost per click) values, computed as the average cost of the clicks that the advertiser got for a particular keyword in a similar time period of two weeks. Summary statistics for the average CPC per keyword and the number of keywords per campaign are given in Table 1.

To represent user behavior by a Markov chain, we follow the approach of [3]. [3] suggests that from advertisers’ perspective, user behavior can be reasonably approximated by a first order Markov chain. In such Markov Chain, state represents the last observed event for the user (for example, user searching for “Prada shirt”) and transition probabilities between states are directly es-

timated from the data. Following [3], we model user state by the keyword that the last user search was matched with. In contrast with [3], we only include clicked ads as model states because pure impression information was missing from our sample.

Next, we add four special states: the begin state (x_b) representing a new user entering the system, the conversion state representing the user conversion event (x_c), the non-conversion state representing the user leaving the system without converting (x_n)⁸ and the final state (x_f). The final state is absorbing and, by construction, conversion and non-conversion states always lead to the final state. The begin state has no incoming edges.

Due to the nature of our data, we only consider two possible advertising levels for every keyword, “advertise” and “do not advertise”, and restrict consideration to the top 250 keywords in each campaign⁹. “Do not advertise” decisions cost nothing and “advertise” decisions cost the average CPC of the corresponding keyword. Consistent with our theoretical model, transition probabilities between states depend on whether the user was exposed to the advertisement or not. The challenge here is that we do not have any observations on the behavior of users not exposed to the advertisement. We suggest a simple workaround for this problem: assume that if the time gap between two consecutive user states (consecutive searches) is large enough (at least one day), the transition between states was not due to influence of the online ad and therefore would have happened even if the ad was not shown to the user. We acknowledge that this is a strong assumption and that transition probability estimates constructed in such way might be biased, however, as our goal is only to evaluate performance of the budget optimization algorithm across multiple campaigns and wide range of parameters, such bias can be tolerated. The summary of graph construction steps is given by Algorithm 4. The algorithm has two configuration parameters that can be tuned. The first parameter α represents the probability that a user can leave the system at any moment of time. We follow a conservative approach and assume that this probability is unaffected by whether the user was exposed to the advertisement in the last step or not. The second parameter C represents the advertiser’s value for a single converted user. As both parameters were unknown in our dataset, we have validated the model across a wide range of them. In the paper, we present results assuming $\alpha = 0.5$ and $C = \$5$.

In the following, we compare performance of three budget optimization algorithms. The baseline algorithm is a simple greedy solution of the fractional knapsack, in which the advertiser sorts all keywords by the immediate ROI value

$$\frac{\mathcal{P}_{x a_1 y} - \mathcal{P}_{x a_0 y}}{\text{CPC}}$$

(ignoring the potential carryover effects to other keywords) and picks the keywords to advertise on in sequence starting from the keyword with the highest ROI. The process stops once we reach the expected allowed budget of the advertising campaign. As the expected campaign budget depends on the assumed model of user behavior, we still have to assume Markovian world when estimating the expected budget in the baseline algorithm. To

⁸We never really know whether the user has dropped out or he is going to come back later and convert. As only small number of users converts, it is always reasonable to assume that the user, who hasn’t converted so far, is not going to.

⁹The main reasons for limiting the number of keywords to only 250 are slow performance of the LP algorithm with large number of variables (the greedy BO algorithm works fine) and presence of significant noise in transition probability estimates for infrequently used keywords.

	Min	Max	Median	Mean
CPC	1.43¢	\$1.34	5.49¢	25.46¢
Keywords	285	998	933	782.33

Table 1: Summary Statistics for CPC and number of keywords per campaign

reconcile this fact with the assumption that the advertiser is optimizing myopically, we assume that, in the baseline algorithm, we advertise to every user only the first time the user enters into the system. We compare the performance of the baseline algorithm with performance of the two alternative budget optimization algorithms:

- the direct approach which is based on solving the linear program **P2** and therefore is guaranteed to construct the optimal solution,
- the greedy budget optimization technique of Algorithm 3.¹⁰

We perform comparison across a range of possible budget values starting from zero budget (in which case the only feasible solution is not to advertise) up to the value V_{max} which represents the budget value for which the budget constraint does not bind the optimal solution anymore. Results of the three algorithms on all nine advertising campaigns are shown in Figure 2. As can be seen from the plot, there was no significant difference in performance of the LP algorithm and the greedy BO algorithm, confirming the positive carryover assumptions and the overall validity of our approach. Both algorithms consistently performed better than the baseline (the fractional knapsack) algorithm. If we measure the algorithm performance by AUC (area under the curve) in Figure 2, then the median gain in AUC was 5.79% and the mean gain in AUC was 9.14%. The largest observed difference in AUC was a gain of 27.14% and the smallest one was a gain of 1.77%. Furthermore, the difference in performance was particularly significant for medium values of the budget constraint, that are neither too small nor too large.

7. CONCLUSIONS

The Internet has become a major advertising medium. While it is relatively easy to start an online advertising campaign, proper allocation of the marketing budget is far from trivial. A major challenge faced by the marketers attempting to optimize their campaigns is in the sheer number of variables they can possibly change and nontrivial interactions between them. In this paper, we consider the important interaction effect between individual advertising decisions: a potential carryover effect that online advertising has on the propensity and the form of user interactions with an advertiser in the future. We adopt the Markov model of user browsing behavior and formulate the budget allocation task of an advertiser as a constrained optimal control problem for a Markov Decision Process (MDP). Using well-developed theory of constrained MDPs, we show that a simple LP algorithm yields the optimal policy. Furthermore, we show that, under reasonable assumptions on the structure of carryover effects, there is a simple greedy algorithm for the optimal solution of the problem that is faster and has an efficient implementation in a parallel processing framework. Using real-world anonymized datasets

¹⁰In fact, we use the alternative version of Algorithm 3, in which we start from $\lambda = +\infty$ and reconstruct $f_\beta(\lambda)$ by gradually decreasing λ .

Algorithm 4 Graph Construction Algorithm.

Configurable Parameters:

- $\alpha \leftarrow 0.5$ (“death” probability)
- $C \leftarrow \$5$ (value per conversion)

Define Actions:

- $\{a_0, a_1\} \leftarrow \{\text{do not advertise, advertise}\}$

Define States:

- $X \leftarrow \text{set of keywords}$
- $X \leftarrow X \cup \{x_b, x_c, x_n, x_f\}$

Define Costs:

- $\forall x \in X' d(x, a_0) \leftarrow 0$
- $\forall x \in X' \setminus \{x_b, x_c, x_n\} d(x, a_1) \leftarrow \text{CPC}(\text{keyword}(x))$
- $\forall x \in \{x_b, x_c, x_n\} d(x, a_1) \leftarrow 0$

Define Rewards:

- $\forall a \in A, x \in X' \setminus \{x_c\} r(x, a) \leftarrow 0$
- $\forall a \in A r(x_c, a) \leftarrow C$

Define Transitions:

- $\forall a \in A, x \in \{x_c, x_n, x_f\} \mathcal{P}_{xaxf} = 1$
- $\forall a \in A, x \in \{x_c, x_n, x_f\}, y \neq x_f \mathcal{P}_{xay} = 0$
- $\forall a \in A, y \in X' \setminus \{x_c, x_n, x_b\} \mathcal{P}_{xby} = \text{fraction of times the first user search was matched with keyword}(y)$
- $\forall a \in A, y \in \{x_c, x_n, x_b, x_f\}, \mathcal{P}_{xby} = 0$
- $\forall x, y \in X' \setminus \{x_c, x_n, x_b\} \mathcal{P}_{xay} = (1 - \alpha)$ times the fraction of times the user search was matched with keyword(y) assuming the user’s previous search was matched with keyword(x) and **the time gap between events was more than one day**
- $\forall x, y \in X' \setminus \{x_c, x_n, x_b\} \mathcal{P}_{xay} = (1 - \alpha)$ times the fraction of times the user search was matched with keyword(y) assuming the user’s previous search was matched with keyword(x) and **the time gap between events was less than one day**
- $\forall x \in X' \setminus \{x_c, x_n, x_b\} \mathcal{P}_{xa_0x_c} = (1 - \alpha)$ times the fraction of times the user converted assuming the user’s previous search was matched with keyword(x) and **the time gap between events was more than one day**
- $\forall x \in X' \setminus \{x_c, x_n, x_b\} \mathcal{P}_{xa_1x_c} = (1 - \alpha)$ times the fraction of times the user converted assuming the user’s previous search was matched with keyword(x) and **the time gap between events was less than one day**
- $\forall x \in X' \setminus \{x_c, x_n, x_b\} \mathcal{P}_{xa_0x_n} = \alpha$
- $\forall x \in X' \setminus \{x_c, x_n, x_b\} \mathcal{P}_{xa_1x_n} = \alpha$
- $\forall a \in A, x \in X' \setminus \{x_c, x_n\} \mathcal{P}_{xaxf} = 0$

Define Inflow of Users:

- $\beta(x_b) \leftarrow 1.0$
 - $\forall x \neq x_b \beta(x) \leftarrow 0.0$
-

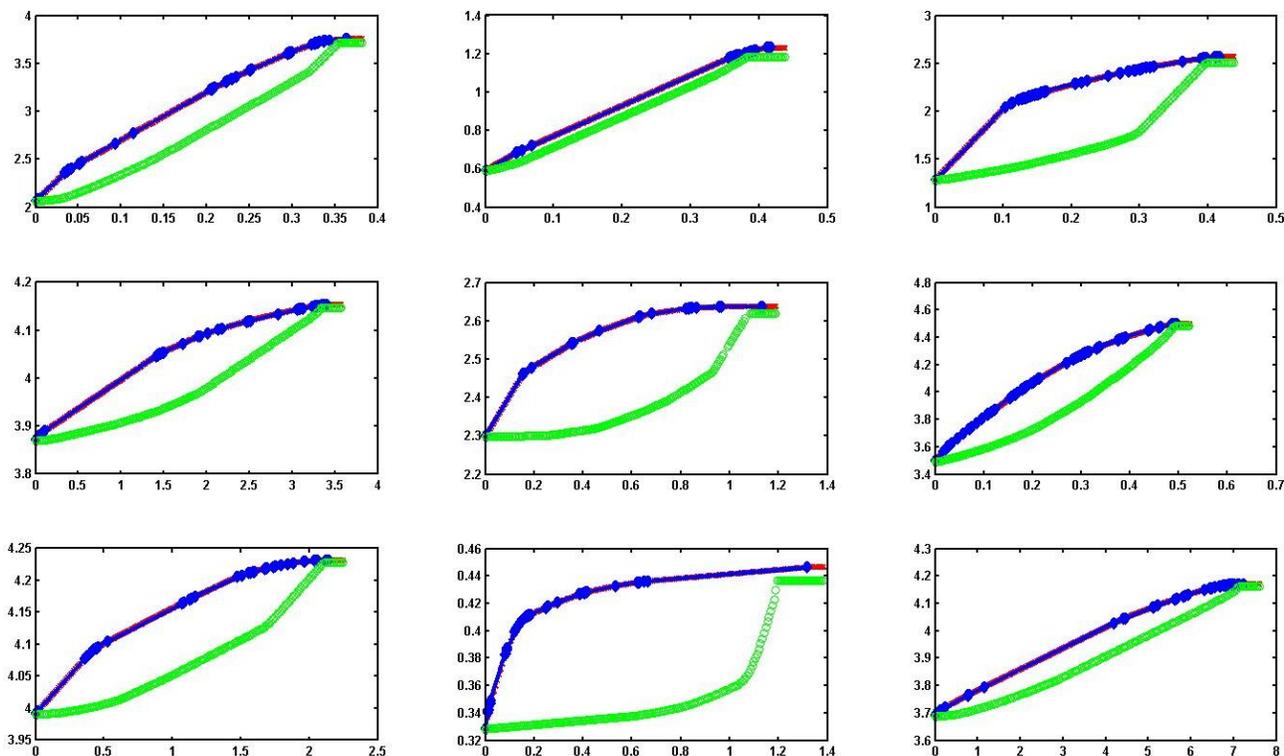


Figure 2: Performance of budget optimization techniques: LP-based (red), greedy BO (blue) and baseline assuming no carry-over (green). Horizontal axis: budget per user in cents. Vertical axis: expected value per user in cents.

from sponsored search advertising campaigns of some large advertisers, we evaluate applicability of our model and performance of the proposed budget allocation algorithm. Our budget allocation algorithm shows 5-10% improvement in revenues against the optimal baseline algorithm ignoring carryover effects, consistent across a wide range of different settings and budget constraints.

References

- [1] G. Aggarwal, J. Feldman, S. Muthukrishnan, and M. Pál. Sponsored search auctions with markovian users. In *WINE '08: Proceedings of the 4th International Workshop on Internet and Network Economics*, pages 621–628, 2008.
- [2] E. Altman. Constrained Markov Decision Processes. Technical Report RR-2574, INRIA, 05 1995.
- [3] N. Archak, V. Mirrokni, and M. Muthukrishnan. Mining advertiser-specific user behavior using adfactors. In *Proceedings of the Nineteenth International World Wide Web Conference (WWW 2010)*, 2010. Forthcoming.
- [4] S. Athey and G. Ellison. Position auctions with consumer search. Working Paper, Available at SSRN: <http://ssrn.com/abstract=1454986>, 2008.
- [5] R. C. Blattberg and A. P. Jeuland. A micromodeling approach to investigate the advertising-sales relationship. *Management Science*, 27(9):988–1005, Sept. 1981.
- [6] M. Charikar, R. Kumar, P. Raghavan, S. Rajagopalan, and A. Tomkins. On targeting markov segments. In *STOC '99: Proceedings of the thirty-first annual ACM symposium on Theory of computing*, pages 99–108, 1999.
- [7] M. Charikar, R. Kumar, P. Raghavan, S. Rajagopalan, and A. Tomkins. On targeting markov segments. In *STOC '99: Proceedings of the thirty-first annual ACM symposium on Theory of computing*, pages 99–108, 1999.
- [8] D. G. Clarke. Econometric measurement of the duration of advertising effect on sales. *Journal of Marketing Research*, 13(4):345–357, Nov. 1976.
- [9] comScore. Whither the click? comscore brand matrix norms prove ‘view-thru’ value of on-line advertising. Available at <http://www.comscore.com/press/release.asp?press=2587>, 2008.
- [10] D. Coppersmith and S. Winograd. Matrix multiplication via arithmetic progressions. In *STOC '87: Proceedings of the nineteenth annual ACM symposium on Theory of computing*, pages 1–6, 1987.
- [11] E. E. Dar, Y. Mansour, V. Mirrokni, S. Muthukrishnan, and U. Nadav. Budget optimization for broad-match ad auctions. In *WWW, World Wide Web Conference*, 2009.
- [12] J. Dean and S. Ghemawat. Mapreduce: simplified data processing on large clusters. *Communications of ACM*, 51(1):107–113, 2008.
- [13] J. Feldman, S. Muthukrishnan, M. Pal, and C. Stein. Budget optimization in search-based advertising auctions. In *EC '07: Proceedings of the 8th ACM conference on Electronic commerce*, pages 40–49, 2007.
- [14] J. Feldman, S. Muthukrishnan, M. Pal, and C. Stein. Budget optimization in search-based advertising auctions. In *EC*

'07: *Proceedings of the 8th ACM conference on Electronic commerce*, pages 40–49, 2007.

- [15] A. Ghose and S. Yang. An empirical analysis of sponsored search performance in search engine advertising. In *WSDM '08: Proceedings of the international conference on Web search and web data mining*, pages 241–250, 2008.
- [16] A. Ghose and S. Yang. Analyzing the Relationship between Organic and Sponsored Search Advertising: Positive, Negative or Zero Interdependence? *Marketing Science*, 2009. Forthcoming.
- [17] A. Ghosh and A. Sayedi. Expressive auctions for externalities in online advertising. In *Proceedings of the Nineteenth International World Wide Web Conference (WWW 2010)*, 2010. Forthcoming.
- [18] R. Gomes, N. Immorlica, and E. Markakis. Externalities in keyword auctions: An empirical and theoretical assessment. In *WINE '09: Proceedings of the 5th International Workshop on Internet and Network Economics*, pages 172–183, 2009.
- [19] S. Guha, K. Munagala, and S. Sarkar. Information acquisition and exploitation in multichannel wireless networks. *CoRR*, abs/0804.1724, 2008.
- [20] K. Jain and V. V. Vazirani. Approximation algorithms for metric facility location and -median problems using the primal-dual schema and lagrangian relaxation. *J. ACM*, 48(2):274–296, 2001.
- [21] D. Kempe and M. Mahdian. A cascade model for externalities in sponsored search. In *WINE '08: Proceedings of the 4th International Workshop on Internet and Network Economics*, pages 585–596, 2008.
- [22] R. Lewis and D. Reiley. Retail advertising works!: Measuring the effects of advertising on sales via a controlled experiment on yahoo! Working paper, Yahoo! Research, 2009.
- [23] P. Milgrom and C. Shannon. Monotone comparative statics. *Econometrica*, 62(1):157–80, January 1994.
- [24] S. Muthukrishnan, M. Pal, and Z. Svitkina. Stochastic models for budget optimization in search-based advertising. In *Lecture Notes in Computer Science, Internet and Network Economics*, pages 131–142, 2007.
- [25] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical Report. Stanford InfoLab., 1999.
- [26] PricewaterhouseCoopers and the Interactive Advertising Bureau. IAB Internet advertising revenue report. Available at <http://www.docstoc.com/docs/5134258/IAB-2008-Report>, 2009.
- [27] D. M. Topkis. Minimizing a Submodular Function on a Lattice. *Operations Research*, 26(2):305–321, 1978.
- [28] J. Wu, J. Cook, Victor J., and E. C. Strong. A Two-Stage Model of the Promotional Performance of Pure Online Firms. *Information Systems Research*, 16(4):334–351, 2005.