Advertising Campaigns Management: Should We Be Greedy?

Sertan Girgin, Jérémie Mary, Philippe Preux, Olivier Nicol

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Advertising Campaigns Management: Should We Be Greedy?

Sertan Girgin — Jeremie Mary — Philippe Preux — Olivier Nicol

N° 7388

Septembre 2010

Optimization, Learning and Statistical Methods

ISSN 0249-6399 ISRN INRIA/RR--7388--FR+ENG

INSTITUT NATIONAL
DE RECHERCHE
EN INFORMATIQUE
ET EN AUTOMATIQUE

INRIA

centre de recherche
LILLE - NORD EUROPE

Advertising Campaigns Management: Should We Be Greedy?

Sertan Girgin , Jeremie Mary , Philippe Preux , Olivier Nicol *

Theme : Optimization, Learning and Statistical Methods
Équipe-Projet Sequel

Rapport de recherche n° 7388 — Septembre 2010 — 24 pages

Abstract: We consider the problem of displaying commercial advertisements on web pages, in the "cost per click" model. The advertisement server has to learn the appeal of each type of visitors for the different advertisements in order to maximize the revenue. In a realistic context, the advertisements have constraints such as a certain number of clicks to draw, as well as a lifetime. This problem is thus inherently dynamic, and intimately combines combinatorial and statistical issues. To set the stage, it is also noteworthy that we deal with very rare events of interest, since the base probability of one click is in the order of 10^{-4}. Different approaches may be thought of, ranging from computationally demanding ones (use of Markov decision processes, or stochastic programming) to very fast ones. We introduce NOSEED, an adaptive policy learning algorithm based on a combination of linear programming and multi-arm bandits. We also propose a way to evaluate the extent to which we have to handle the constraints (which is directly related to the computation cost). We investigate performance of our system through simulations on a realistic model designed with an important commercial web actor.

Key-words: Advertisement selection, web sites, optimization, non-stationary setting, linear Programming, multi-arm bandit, CTR estimation, exploration-exploitation trade-off.

* name.surname@inria.fr

Centre de recherche INRIA Lille – Nord Europe
Parc Scientifique de la Haute Borne
40, avenue Halley, 59650 Villeneuve d'Ascq
Téléphone : +33 3 59 57 78 00 — Télécopie : +33 3 59 57 78 50

Gestion de campagnes publicitaires: Doit-on être gourmand?

Résumé : Nous nous intéressons au problème de la sélection de messages publicitaires sur des pages web dans le modèle de paiement au clic. Pour cela, le serveur doit apprendre l'appétance de chaque type de visiteurs pour les différentes publicités en stock afin de maximiser ses revenus. Dans un contexte réaliste, les publicités possèdent des contraintes telles qu'un nombre de clics à obtenir et une durée de vie. Ce problème est dynamique et combine intimement des aspects combinatoires et statistiques ; de plus, il est important de noter que nous considérons des événements rares, la probabilité de clic de base étant de l'ordre de 10^{-4}. Différentes approches peuvent etre envisagées, allant d'approches extrêmement gourmandes en temps de calcul (en utilisant des processus décisionnel de Markov ou une formulation de type programmation stochastique) à des approches très rapides. Nous introduisons NOSEED qui est un algorithme adaptatif d'apprentissage de politique basé sur une combinaison de programmation linéaire et de bandits multi-bras. Nous proposons également une manière d'évaluer les contraintes à satisfaire, ce qui est directement relié au coût en temps de calcul. Nous investiguons les performances de notre algorithme dans un modèle réaliste conçu avec un important acteur du web commercial.

Mots-clés : Sélection de messages publicitaires, site Web, optimisation, problème non stationnaire, programme linéaire, bandit multi-bras, estimation du CTR, compromis exploration-exploitation.

1 Introduction

The ability to efficiently select items that are likely to be clicked by a human visitor of a web site is a very important issue. Whether for the mere comfort of the user to be able to access the content he/she is looking for, or to maximize the income of the website owner, this problem is strategic. The selection is based on generic properties (date, world news events, ...), along with available personal information (ranging from mere IP related information to more dedicated information based on the login to an account). The scope of applications of this problem ranges from advertisement or news display (see for instance the Yahoo! Front Page Today Module), to web search engine result display. There are noticeable differences between these examples: in the first two cases, the set of items from which to choose is rather small, in the order of a few dozens; in the latter case, the set contains billions of items. The lifetime of items may vary considerably, from a few hours for news, to weeks for ads, to years for pages returned by search engine. Finally, the objective ranges from drawing attention and clicks on news, to providing the most useful information for search engines, to earning a maximum of money in the case of advertisement display. Hence, it seems difficult to consider all these settings at once and in this paper, we consider the problem of selecting advertisements, in order to maximize the revenue earned from clicks: we consider the "cost to click" economic model in which each single click on an advertisement brings a certain revenue. We wish to study principled approaches to solve this problem in the most realistic setting; for that purpose, we consider the problem with finite amounts of advertising campaigns, finite amounts of clicks to gather on each campaign, finite campaign lifetimes, the appearance and disappearance of campaigns along days, finite flow of visitors and page requests, ... By that, we would like to emphasize that our goal is not to optimize any asymptotic behavior and exhibit algorithms that are able to achieve optimal asymptotic behavior (but perform badly for much too long). To the opposite, we concentrate on the practical problem faced here and now by the web server owner: he/she wants to make money now, and do not really care about ultimately becoming a billionaire when the universe will have collapsed (which is likely to happen in a not so remote future with regards to asymptotic times either). In the same order of ideas, we also want to keep the solution computable in "real"-time, real meaning here within a fraction of a second, and able to support the high rate of requests observed on the web server of an important web portal. Of course, such requirements impede the quality of the solution, but these requirements are necessary from the practical point of view; furthermore, since we have to deal with a lot of uncertainty originating from various sources, the very notion of optimality is quite relative here.

In section 2, we formalize the problem we deal with; we actually define a series of problems of increasing complexity, ranging from a static setting in which all information is known, to the dynamic case where key information is missing. Assessing algorithms in the dynamic case is difficult, in particular from a methodological point of view, and spanning this range of problems let us assess our ideas in settings in which there is a computable optimal solution against which the performance of algorithms may be judged. Section 3 presents related works. Section 4 presents some experimental results in both static and dynamic settings. Finally, section 5 concludes and we briefly discuss the lines of foreseen future works.

2 Formalization of the problem

In this section, we formalize the problem under study, and introduce the vocabulary and the notation used throughout the paper. The problem we tackle is actually changing over time; for pedagogical reasons, we first introduce a static version of this problem, before moving to the general, dynamic case. We also introduce our algorithm to solve it: Near Optimal Sequential Estimation and Exploration for Decision (NOSEED).

2.1 The static version of the problem

At a given time t, there is a pool of K advertising campaigns denoted by K^t. Each advertising campaign in the pool $Ad_k \in K^t$ is characterized by a tuple $(status_k, S_k, L_k, B_k, b_k^t, r_k)$ where k is the unique identifier of the advertising campaign. $status_k$, S_k, L_k and B_k are its status, starting time, lifetime and total click budget, respectively. The advertising campaign starts at time $t = S_k$, lasts for tL_k time steps and expects to receive B_k clicks during its lifetime. The status of an advertising campaign can be either one of the following (Fig. 1):

scheduled when the campaign will begin at some time in the future (*i.e.* $t < S_k$) and accordingly, the advertisements of this campaign can not yet be displayed,

running when the campaign is active (*i.e.* $S_k \leq t < S_k + L_k$) and accordingly, the advertisements of this campaign can be displayed,

expired when the campaign has ended (*i.e.* $S_k + L_k \leq t$ or $b_k^t = 0$) and accordingly, the advertisements of this campaign can no longer be displayed.

$b_k^t \leq B_k$ denotes the remaining budget of the campaign at time t and r_k is the revenue obtained per click on an advertisement of the campaign k. We will use $l_k^t \in [0, L_k]$ to denote the remaining lifetime of Ad_k at time t; it is defined as $l_k^t = \max(0, S_k + L_k - \max(S_k, t))$.

Now, the problem that we are interested in is as follows:

- The web server receives a continuous stream of visitors, each of which is assumed to be from one of N possible user profiles. The probability that the visitor is from a certain profile P_i is R_i with $\sum_{i=1}^{i=N} R_i = 1$, *i.e.* has a categorical distribution with probability mass function $f_{\mathbf{P}}(P_i) = R_i$.

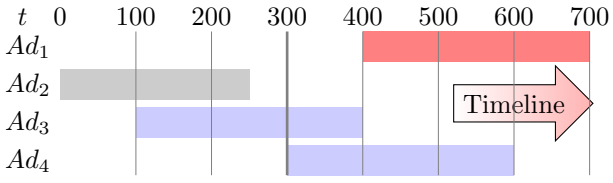

Figure 1: At time $t = 300$, Ad_1 is in scheduled state (in red), Ad_2 has expired (in gray), Ad_3 and Ad_4 are running with remaining lifetimes of 100 and 300, respectively (in blue).

- When a visitor visits the web site, a new "session" begins[1] and we observe one or several iterations of the following sequence of events:
  - the visitor requests a certain page to the web server (via its URL) at time t
  - the requested page is displayed to this visitor with an advertisement Ad_k embedded in it,
  - the visitor clicks on the advertisement with probability $p_{i,k}$ where i denotes the user profile of the visitor (*i.e.* a Bernoulli trial with success probability $p_{i,k}$); this probability is usually called the *click-through rate* (CTR),
  - if there is a click, then the revenue associated with the advertisement, that is r_k, is incurred.
- After a certain number of page requests, the visitor leaves the web site and the session terminates.

The objective is to maximize the total revenue by choosing the advertisements to be displayed "carefully". Since page requests are atomic actions, in the rest of the paper we will take a page request as the *unit of time* to simplify the discussion, *i.e.* a time step will denote a page request and vice versa. Note that in the real-world, some of the parameters mentioned above may not be known with certainty in advance. For example, we may not know the visit probabilities of the user profiles, their probability of click for each advertising campaign, the actual profiles of the visitors, or the number of requests that they will make; the number of visitors may change with time and new advertising campaigns may emerge. These and other issues that we will address throughout the paper make this problem a *non-trivial* one to solve.

We may formulate this problem as a Markov decision problem (MDP, see Bertsekas [5]). From a practical point of view, the state space would be huge, making its resolution very computationally demanding, and unable to meet our requirements in this regard. However, the fact that this problem may be formulated as an MDP provides a proof that the problem we consider has a solution.

In order to better understand the problem and derive our solution, we will first start with and investigate the simplest setting in which all the information is available, and subsequently move to the setting in which only a part of the information is available. Then, in the next section, we will further move to the dynamic setting.

2.1.1 Static setting with full information

In this setting, we assume that there is a fixed time horizon T and all parameters are known; to be more precise, (a) the pool of advertising campaigns at each time step $0 \leq t < T$ is given, (b) the visit probabilities of user profiles, R_i, and their click probabilities for each advertising campaign, $p_{i,k}$ are known, and (c) there is no uncertainty in the actual profiles of the visitors, *i.e.* we know the profile of each visitor. Note that, even if we have full information, the visitor at

[1] Returning visitors do not change the nature of the problem given that the session information persists, and for the sake of simplicity we will be focusing on this particular setting.

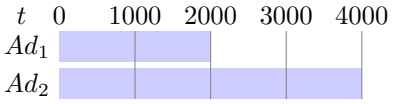

Figure 2: A toy example in which HEV and SEV policies have suboptimal performance. Ad_1 and Ad_2 have the same unit revenue per click, click probabilities of 0.005 and 0.01, and total budgets of $B_1 = 10$ and $B_2 = 20$ clicks, respectively. The expected total revenues of HEV and SEV are 20 and $23\frac{1}{3}$ compared to a maximum achievable expected total revenue of 30.

time t and whether he/she will click on the displayed advertisement or not are still unknown.

Under this setting, given a visitor from profile P_i at time t, one possible and efficient way to choose an advertising campaign to display would be to pick the running advertising campaign with the highest expected revenue per click among K^t, that is $\text{argmax}_{Ad_k \in K^t} r_k p_{i,k}$[2]; we will call this particular method the *highest expected value* (HEV) policy. Alternatively, we can employ a stochastic selection method where the selection probability of a running advertising campaign is proportional to its expected revenue per click. This variant will be called the *stochastic expected value* (SEV) policy.

As both policies exploit advertising campaigns with possibly high return and assign lower priority to those with lower return, one expects them to perform well if the lifetimes of the advertising campaigns are "long enough" to ensure their total click budgets. However, they may show inferior performances even in some trivial situations. For example, assume that there is a single user profile and two advertising campaigns, Ad_1 and Ad_2, starting at time $t = 0$ with click probabilities of 0.005 and 0.01, lifetimes of $L_1 = 2000$ and $L_2 = 4000$ time steps, and total budgets $B_1 = 10$ and $B_2 = 20$ clicks, and unit revenues per click *i.e.* $R_1 = R_2 = 1$ (Fig. 2). In this particular case, starting from $t = 0$, HEV will always choose Ad_2 until it expires (on expectation at $t = 2000$ where Ad_1 also expires) and result in an expected total revenue of 20 units; SEV will display on average twice as many advertisements from Ad_2 compared to Ad_1 during the first 2000 time steps, and perform slightly better with an expected total revenue of $23\frac{1}{3}$. However, both figures are less than the value of 25 that can be achieved by choosing one of the campaigns *randomly* with equal probability. Note that, by displaying advertisements from only Ad_1 in the first 2000 time steps until it expires and then Ad_2 thereafter, it is possible to obtain an expected total revenue of 30 that satisfies the budget demands of both advertising campaigns; the lifetime of Ad_2, which is long enough to receive a sufficient number of clicks with the associated click probability, allows this to happen. In order to derive this solution, instead of being short-sighted, it is compulsory to take into consideration the interactions between the advertising campaigns over the entire timeline and determine which advertising campaign to display accordingly, in other words *do planning*. Observing Fig. 2, it is easy to see that the interactions between the advertising campaigns materialize as *overlapping time intervals* over the timeline (see Fig. 3); in this toy example the intervals are $I_1 = [0, 2000]$ and $I_2 = [2000, 4000]$, and what we are trying to

[2]Ties are broken randomly.

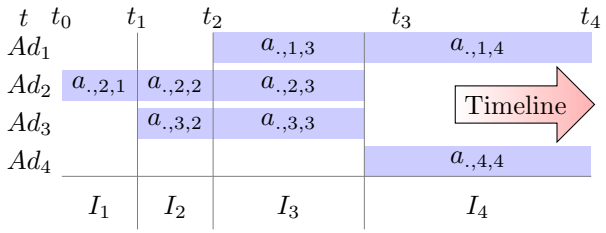

Figure 3: The timeline divided into intervals and parts. I_j denotes the j^{th} interval $[t_{j-1}, t_j]$ and $a_{k,j}$ denotes the allocation for advertising campaign Ad_k in interval I_j. The first index of a (user profile) is left unmentioned for the sake of clarity.

find is the *optimal allocation* of the number of advertising campaign displays in each interval. This can be posed as the following optimization problem where $a_{k,j}$ denotes the number of displays allocated to Ad_k in the interval I_j:

$$\begin{aligned} \text{maximize} \quad & 0.005 \times a_{1,1} + 0.01 \times (a_{2,1} + a_{2,2}) \\ \text{subject to} \quad & a_{1,1} + a_{2,1} \leq 2000, a_{2,2} \leq 2000 \\ & 0.005 \times a_{1,1} \leq 10, 0.01 \times (a_{2,1} + a_{2,2}) \leq 20 \end{aligned}$$

which has an optimal solution of $a_{1,1} = a_{2,2} = 2000$ and $a_{2,1} = 0$. One can then use this optimal allocation to calculate the display probabilities for both advertising campaigns proportional to the number of displays allocated to them in the corresponding time intervals.

Let s_k^t be the relative starting time of a *non-expired* advertising campaign Ad_k at time t defined as $s_k^t = max(0, S_k - t)$ and $e_k^t = S_k + L_k - t$ be its relative ending time. In general, given a pool of advertising campaigns $K^t = \{Ad_1, \ldots, Ad_K\}$ at time t, the time intervals during which the advertising campaigns overlap with each other can be found from the set of their relative starting and ending times. Let $[t_0, t_1, \ldots, t_M]$, $M \leq 2 \times K$, be the sorted list of elements of the set $\{x | x = s_k^t \text{ or } x = e_k, k \in K^t\}$[3]. By definition, the M intervals defined by $I_j = [t_{j-1}, t_j], 1 \leq i \leq M$ cover the entire timeline of the pool of the advertisement campaigns. Let $AI_j = \{Ad_k | s_k < t_i \leq e_k\}$ be the set of running advertising campaigns in interval I_j. Note that for some of the intervals, this set may be empty; these intervals are not of our interest as there will be no advertising campaigns to display during such intervals and without loss of generality we can ignore them. Let $\mathcal{A}^t = \{I_j | AI_j \neq \emptyset\}$ be the set of remaining intervals, $l_j = t_j - t_{j-1}$ denote the length of interval I_j, and $IA_k = \{I_j | Ad_k \in AI_j\}$ be the set of intervals that cover Ad_k. Generalizing the formulation given above, we can define the optimization problem that we want to solve as follows where $a_{i,k,j}$ denotes the number of displays allocated to Ad_k in the interval I_j for the user profile P_i:

$$\text{maximize} \quad \sum_{I_j \in \mathcal{A}^t} \sum_{Ad_k \in AI_j} r_k p_{i,k} a_{i,k,j} \tag{1}$$

$$\text{subject to} \quad \sum_{Ad_k \in AI_j} a_{i,k,j} \leq R_i l_j, \forall\ 1 \leq i \leq N, I_j \in \mathcal{A}^t \tag{2}$$

[3]With a slight abuse of notation, we will use $k \in K^t$ and $Ad_k \in K^t$ interchangeably.

$$\sum_{i=1}^{N} \sum_{I_j \in I A_k} p_{i,k} a_{i,k,j} \leq b_k^t, \forall\ Ad_k \in K^t \tag{3}$$

The objective function (Equation 1) aims to maximize the total expected revenue, the first set of constraints (Equation 2) ensures that for each interval we do not make an allocation for a particular user profile that is over the capacity of the interval (*i.e.* the portion of the interval proportional to the visit probability of the user profile), and the second set of constraints (Equation 3) ensures that we do not exceed the remaining total click budgets. This corresponds to the maximization of a *linear objective function* ($a_{i,k,j}$ being the variables), subject to *linear inequality constraints*, which is a *linear programming* problem. It can be solved efficiently using the simplex algorithm or interior-point methods and other existing large scale approaches if necessary.

The solution of the linear program at time t, *i.e.* the assignment of values to $a_{i,k,j}$, indicates the number of displays that should be allocated to each advertising campaign for each user profile and in each interval, but it does not provide a specific way to choose the advertising campaign to display to a particular visitor from user profile P_i at time t. For this, we need a method to calculate the display probability of each running advertising campaign from their corresponding allocations, *i.e.* that maps allocations to probabilities. It is easy to see that if the first interval I_0 is not of the form $[0, l_0]$ then this means that there are no running advertising campaigns to display at time t. Otherwise, let $\bar{a}_{i,j} = \sum_{Ad_k \in AI_j} a_{i,k,j}$ be the total allocation of advertising campaign displays for the user profile P_i in interval I_j and $\hat{p}_{k,0} = a_{i,k,0}/\bar{a}_{i,0}$ be the ratio of displays allocated to the advertising campaign Ad_k in the first interval (forming a categorical distribution). One can either pick the advertising campaign having the highest ratio, which we will call the *highest LP policy* (HLP), or employ a stochastic selection method similar to SEV in which the selection probability is proportional to its ratio, which will be called the *stochastic LP policy* (SLP). Note that, as we are planning for the entire timeline, the solution of the linear program at time t may not allocate any advertising campaigns to a particular user profile i, *i.e.* it may be the case that $\bar{a}_{i,0} = 0$, simply suggesting not to display any advertisement to a visitor from that user profile. In practice, when the current user is from such a user profile, choosing an advertising campaign with a low (or high) expected revenue per click instead would be a better option and likely to increase the total revenue at the end.

By defining and solving the linear program at each time step $0 \leq t < T$ for the current pool of non-expired advertising campaigns (which depends on the visitors that have visited the web site up until that time step, the advertising campaigns displayed to them and visitors' reactions to those), and employing one of the policies mentioned above, advertising campaigns can be displayed in such a way that the total expected revenue is maximized, ignoring the uncertainty in the predictions of the future events (we will subsequently discuss the issues related to uncertainty). In this case, the performance of HLP and SPL policies will be similar to each other (due to the fact that the preference will gradually shift toward advertising campaigns that have initially lower ratios as those with high ratios eventually receive more clicks reducing their remaining budgets and therefore ratios).

1: $t = 0$
2: **while** $t < T$ **do**
3: Let P_i be the user profile of the current visitor
4: **if** $\exists Ad_k \in K^t$ such that $b_k = 0$ **or** $period(t, T)$ **then**
5: Find intervals $\mathcal{A}^t$ for K^t at time t
6: Solve the optimization problem and determine the display allocation of the advertising campaigns $a_{i,k,j}$
7: **end if**
8: Let $I_j = [t_{j-1}, t_j]$ such that $t_{j-1} \leq t < t_j$ be the current interval
9: $\bar{a}_{i,j} = \sum_{Ad_k \in AI_j} a_{i,k,j}$ */* Total allocations in this interval */*
10: **if** $\bar{a}_{i,j} > 0$ **then**
11: **for all** $Ad_k \in AI_j$ **do**
12: $\hat{p}_k = a_{i,k,j}/\bar{a}_{i,j}$ */* Calculate display probabilities */*
13: **end for**
14: Choose an advertising campaign Ad_k based on $\hat{p}_k$ */* e.g. using HLP or SLP */*
15: $a_{i,k,j} = a_{i,k,j} - 1$ */* Update the allocation for Ad_k */*
16: **else**
17: Choose a running advertising campaign Ad_k if any. */* There are no advertising campaign allocations to display for this user profile */*
18: **end if**
19: **if** visitor clicks on Ad_k **then**
20: $b_k^{t+1} = b_k^t - 1$ */* Decrement the remaining budget */*
21: **end if**
22: $t = t + 1$
23: Update the statuses of the advertising campaigns
24: **end while**

Figure 4: Use of NOSEED Algorithm to choose advertising campaigns by solving the optimization problem at regular intervals and/or intermittently.

When the number of advertising campaigns, and consequently the number of variables and constraints, is high, or there is a need for fast response time, solving the optimization problem at each time step may not be feasible. An alternative approach would be to solve it with regular periods and/or intermittently (such as, when the budget of an advertising campaign is met and hence it becomes expired), and use the resulting allocation to determine the advertising campaigns to be displayed until the next problem instance is solved, *i.e.* iterations of planning followed by multiple steps of execution. This can be accomplished by updating the allocated number of advertisement campaign displays as we move along the timeline and reducing the allocation of the chosen advertising campaigns in the corresponding intervals. The complete algorithm is presented in Fig. 4. Note that, in practice the planning and execution steps can be asynchronous as long as the events that have occurred from the time that planning has started until its end are reflected properly to the resulting allocation.

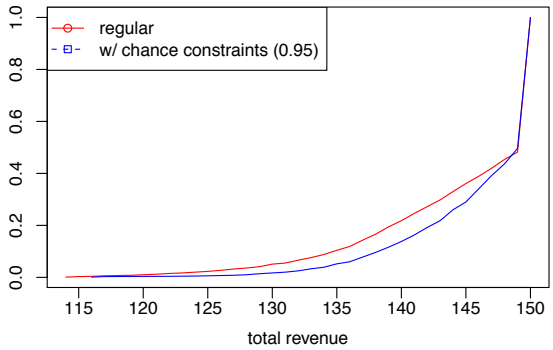

Figure 5: Empirical cumulative distribution of the total revenue over 1000 independent runs on the toy example with two advertisement campaigns. In reality, the realization will be only one of the runs and therefore more concentration near the maximum value is better (see text for more explanation).

2.1.2 Uncertainty in the static setting with full information

The static setting with full information has two sources of uncertainty: (a) the user profiles of visitors are drawn from a categorical distribution, and (b) each advertising campaign display is a Bernoulli trial with a certain probability, which is known, and the result is either a success (*i.e.* click) or a failure (*i.e.* no click). The aforementioned linear programming solution of the optimization problem focuses on what happens in the expectation. Following the resulting policy in different instances of the same problem[4] may lead to different realizations of the total revenue that vary from its expected value (due to the fact that the number of visitors from each user profile and the number of clicks on the displayed advertising campaigns will not exactly match their expected values). As a simple example, consider the case in which there is a single user profile and two advertising campaigns Ad_1 and Ad_2 both having the same unit revenue per click and a lifetime of 10^5 time steps, click probabilities of 0.001 and 0.002, and total budgets of 50 and 100, respectively. The solution of the linear program would allocate 50000 displays to each advertising campaign with an expected total revenue of 150, thus satisfying the budget demands. Fig. 5 shows the cumulative distribution of the total revenue over 1000 independent runs for this problem using the stochastic LP policy and solving the optimization problem once at the beginning. Although values that are equal to or near the expected total revenue are attained in more than half of the runs, one can observe a substantial amount of variability. In reality, reducing this variability may also be important and could be considered as a secondary objective to obtaining a high total revenue. For the given example, slightly increasing the display probability of Ad_2 and decreasing that of Ad_1 would enable the accomplishment of this objective by protecting against the risk of receiving fewer clicks than expected for Ad_2 without considerably compromising the outcome as the same risk also exists for Ad_1. This leads to the question of how to incorporate risk-awareness to our formulation of the optimization problem.

[4] An "instance" refers here to a certain realization of the random problem.

When we look closely at the objective function and the constraints of the linear program (Equations 1-3), we can identify two sets of expressions of the form $R_i l_j$ and $p_{i,k} a_{i,k,j}$; the first one denotes the expected number of visitors from user profile P_i during the timespan of interval I_j, and the second one denotes the expected number of clicks that would be received if the advertising campaign Ad_k is displayed $a_{i,k,j}$ times to the visitors from user profile P_i. Note that visits from a particular user profile occur with a known average rate R_i, and each visit occurs independently of the time since the previous visit. Therefore, the number of such visits in a fixed period of time t can be considered a random variable having a Poisson distribution with parameter $\lambda = R_i t$ which is equal to the expected number of visits that occur during that time period. Similarly, the number of clicks that would be received in a fixed period of time if advertising campaign Ad_k is displayed to the visitors from user profile P_i can also be considered a random variable having a Poisson distribution with parameter $\lambda = p_{i,k} t$. Let $\mathbf{Po}(\lambda)$ denote a Poisson-distributed random variable with parameter λ. Replacing $R_i l_j$ and $p_{i,k} a_{i,k,j}$ with corresponding random variables, we can convert the linear program into the following stochastic optimization problem:

$$\max \quad \sum_{I_j \in \mathcal{A}^t} \sum_{Ad_k \in AI_j} r_k \mathbb{E}[\mathbf{Po}(p_{i,k} a_{i,k,j})] \tag{4}$$

$$\text{s.t.} \quad \sum_{Ad_k \in AI_j} a_{i,k,j} \leq \mathbf{Po}(R_i l_j), \forall\ 1 \leq i \leq N, I_j \in \mathcal{A}^t \tag{5}$$

$$\sum_{i=1}^{N} \sum_{I_j \in IA_k} \mathbf{Po}(p_{i,k} a_{i,k,j}) \leq b_k^t, \forall\ Ad_k \in K^t \tag{6}$$

The summation of independent Poisson-distributed random variables also follows a Poisson distribution whose parameter is the sum of the parameters of the random variables. Assuming that $\mathbf{Po}(p_{i,k} a_{i,k,j})$ are independent, the budget constraints in equation (6) can be written as

$$\mathbf{Po}(\sum_{i=1}^{N} \sum_{I_j \in IA_k} p_{i,k} a_{i,k,j}) \leq b_k^t, \forall\ Ad_k \in K^t \tag{7}$$

which is equivalent to its linear program counterpart in expectation. The rationale behind this set of constraints is to bound the total expected number of clicks for each advertising campaign (while at the same time trying to stay as close as possible to the bounds due to maximization in the objective function). Without loss of generality, assume that in the optimal allocation the budget constraint of advertising campaign Ad_k is met. This means that the expected total number of clicks for Ad_k will be a Poisson-distributed random variable with parameter b_k^t and in any particular instance of the problem the probability of realizing this expectation (our target) would be 0.5. In order to increase the likelihood of reaching the target expected total number of clicks, a possible option would be to use a higher budget limit in the constraint. Let α_k be our risk factor[5] and $\mathbf{Po}(\lambda_k)$ be the Poisson-distributed random variable having the smallest parameter λ_k such that $Pr(\mathbf{Po}(\lambda_k) > b_k^t - 1) \geq \alpha_k$ which is equivalent to

$$1 - \alpha_k \geq \mathbf{F}_{\mathbf{Po}(\lambda_k)}(b_k^t - 1)$$

where $F_{\mathbf{Po}(\lambda_k)}$ is the cumulative distribution function of $\mathbf{Po}(\lambda_k)$. Note that b_k^t and α_k are known, and λ_k can be found using numerical methods. If we

[5]Typical values include 0.90, 0.95, and 0.99.

replace b_k^t with λ_k in the budget constraint and solve the linear optimization problem again, the expected total number of clicks for Ad_k based on the new allocation would be greater than or equal to b_k^t and will have an upper bound of λ_k. Following the same strategy, one can derive new bounds for the user profile constraints and replace $R_i l_j$ terms in equation (5) with the smallest value of $\lambda_{i,j}$ such that the Poisson-distributed random variable $\mathbf{Po}(\lambda_{i,j})$ satisfies $1 - \alpha_{i,j} \geq \mathbf{F}_{\mathbf{Po}(\lambda_{i,j})}(R_i l_j)$ and $\alpha_{i,j}$ is the risk factor. In this case, an additional set of constraints defined below is necessary to ensure that for each interval the sum of advertising campaign allocations for all user profiles do not exceed the length of the interval:

$$\sum_{i=1}^{N} \sum_{Ad_k \in AI_j} a_{i,k,j} \leq l_j, \forall I_j \in \mathcal{A}^t \tag{8}$$

As presented in Fig. 5, in our simple example using a common risk factor of 0.95 results in a cumulative distribution of total revenue which is more concentrated toward the optimal value compared to the regular linear programming approach.

2.1.3 Static setting with partial information

In the settings discussed so far, we have assumed that two important sets of parameters, the visit probabilities of user profiles $\{R_i\}$ and their click probabilities for each advertisement campaign $\{p_{i,k}\}$ are known. However, this is a rather strong assumption and in reality these probabilities are hardly known in advance; instead, they have to be estimated based on observations, such as the profiles of the existing visitors, the advertisement campaigns that have been displayed to them and their responsive actions (*i.e.* whether they have clicked on a displayed advertisement or not). An accurate prediction of these probabilities results in the display of more attractive advertisements to the web site visitors.

The simplest way to estimate unknown probabilities would be to use maximum likelihood estimation. In our problem, the profile of a visitor can be considered a categorical random variable $\mathbf{R}$ with profile P_i having an estimated probability of $\hat{R}_i$, and the click of a visitor from user profile P_i on an advertisement from advertising campaign Ad_k can be considered a Bernoulli random variable $\mathbf{p}_{i,k}$ with success probability $\hat{p}_{i,k}$. Let $visit_i^t$ denote the total number of visitors from user profile P_i that have visited the web site at time $0 \leq t$, then the maximum likelihood estimate of $\hat{R}_i$ will be $visit_i^t/(t+1)$, and similarly the maximum likelihood estimate of $\hat{p}_{i,k}$ at time t will be $click_{i,k}^t/display_{i,k}^t$ where $click_{i,k}^t$ is the number of times that visitors from user profile P_i clicked on advertisement Ad_k and $display_{i,k}^t$ is the number of times Ad_k had been displayed to them.

Alternatively, we can employ Bayesian maximum a posteriori estimates using the conjugate priors. The conjugate priors of the categorical and Bernoulli distributions are Beta and Dirichlet distributions, respectively. If $\mathbf{Beta}(\alpha_{i,k}, \beta_{i,k})$ is the Beta prior with hyper-parameters $\alpha_{i,k}$ and $\beta_{i,k}$ for $\mathbf{p}_{i,k}$, then the posterior at time t is the Beta distribution $\mathbf{Beta}(\alpha_{i,j} + click_{i,k}^t, \beta_{i,j} + display_{i,k}^t)$. $\mathbf{Beta}(1,1)$ corresponds to having a uniform prior. At time t, the posterior of the prior Dirichlet distribution with hyper-parameters v_i for $\mathbf{R}$ will have hyper-parameters $v_i + visit_i^t$. The initial hyper-parameters can be guessed or

determined empirically based on historical data. As we will see later in the experiment section, choosing good priors may have a significant effect on the outcome.

By estimating probabilities at each time step (or periodically) and replacing the actual values with the corresponding estimates, we can use the previous algorithm presented for the full information setting (Fig. 4) to determine allocations (optimal up to the accuracy of the estimations) and choose advertising campaigns to display. For maximum a posteriori estimates, the mode of the posterior distribution can be used as a point estimate and a single instance of the problem can be solved, or several instances of the problem can be generated by sampling probabilities from the posterior distributions, solved separately and then the resulting allocations can be merged (for example taking their mean; note that, in this case the final allocations will likely be not bound to the initial constraints).

As in many online learning problems, one important issue that arises in this approach is the need for balancing the exploitation of the current estimates and exploration, i.e. estimation of the unknown or less-known (*e.g.*, with higher variance) parameters. Using the solution of the optimization problem without introducing any additional exploration may introduce substantial bias to the results. This exploration/exploitation trade-off problem can be formulated as a multi-arm bandit problem (with the advertising campaigns in the role of arms). Based on the multi-arm bandit framework, exploration can be introduced to the allocation policy in various ways, among which we mention the following two:

Policy-Modification The existing non-exploratory policies can be augmented with an additional mechanism in order to have exploration. Such modifications includes ε-greedy in which the underlying policy is followed with a high probability $1 - \varepsilon$ and a running advertising campaign is chosen at random with probability ε. One can derive other possible solutions from the bandit literature such as the UCB rule.

Estimation-Modication In this approach, the probability of click estimates are systematically modified (before solving the optimization problem) in order to favor the advertising campaign and user profile couples according to the uncertainty on their estimation based on the following principle: *the more uncertain the estimate, the more exploration may be rewarding.* By giving them artificially a higher probability of click tends to favor their use, and consequently the exploration. For this purpose, Abe and Nakamura [1] use Gittins indices. Similarly one can also use UCB indices associated with the estimates, or with a value sampled from the posterior Beta distribution over the expected reward (see Granmo [6]). Empirically, this second way of increasing exploration does not seem to work as well as the first one (for example, ε-greedy with fine-tuned ε) especially if we do not replan at each time step. We believe that the reason for this situation is that such methods lead to solutions that only explore the most uncertain areas of the search space.

2.2 Dynamic Setting

In this more general and realistic setting, the time horizon is no longer fixed, *i.e.* does not have a limited length T but instead it is assumed that T is infinite,

and furthermore new advertisement campaigns may appear with time. We will consider two main cases in which either we have a *generative model* or not; given a set of parameters and the current state, a generative model can (stochastically) generate a continuous stream of advertisement campaigns (together with all related-information, such as click probabilities of user profiles for each generated advertisement campaign) during a specified time period.

When a generative model is not available, what we have is an incomplete and uncertain image of the timeline; we know only about advertising campaigns that have been revealed, and new advertisement campaigns may appear periodically or randomly according to a model which is unknown. In this setting, at any time step t the known pool of advertising campaigns (running or scheduled) imposes a maximum time horizon H_{max}. Although, it is possible to apply the aforementioned methods and calculate the allocations for the known advertising campaigns, doing so would ignore the possibility of the arrival of new advertising campaigns that may overlap and intervene with the existing ones; the resulting *long-term* policies may perform well if the degree of dynamism in the environment is not high. On the contrary, one can focus only on short or medium-term conditions omitting the known advertising campaigns that start after a not-too-distant time H in the future, *i.e.* do planning for the advertising campaigns within the chosen planning horizon. The resulting policies will be greedier as H is smaller and disregard the long-time interactions between the existing advertisement campaigns; however, they will also be less likely to be affected by the arrival of new campaigns. An example that demonstrates the effect of the planning horizon on the resulting policies is presented in Fig. 6. For such policies, choosing the optimal value of the planning horizon is not trivial due to the fact that it strongly depends on the unknown underlying model. One possible way to overcome this problem would be to solve for a set of different planning horizons $H_1, \ldots, H_u = H_{max}$ (as the planning horizons are different, the structure of the optimization problems, *i.e.* variables, objective function, constraints *etc.* would also be different from each others) and then combine the resulting probability distributions of advertising campaign displays (such as by majority voting).

When a generative model of advertising campaigns is available, it can be utilized to compensate for the uncertainty in future events. In this case, in addition to the known pool of advertising campaigns, the model allows us to generate a set of *hypothetical* advertising campaigns (for example, up to H_{max}), simulating what may happen in future, and include them in the planning phase. By omitting allocations made for these hypothetical advertisement campaigns from the (optimal) allocation scheme found by solving the optimization problem, display probabilities that inherently take into consideration the effects of future events can be calculated. Note that, this would introduce bias to the resulting policies which can be reduced by running multiple simulations and combining their results as discussed before.

3 Related work

We review the existing work on the problem of advertisement selection for display on web pages, and related problems. We also discuss our own work in respect to these works.

H	R_i	Budget	100	100
			Ad_1	Ad_2
20	$1/2 \rightarrow$	P_1	**10**	0
	$1/2 \rightarrow$	P_2	**10**	0

Planning with $H = 20 \rightarrow$ **Greedy**
P_1: 100% on Ad_1
P_2: 100% on Ad_1

H	R_i	Budget	100	100
			Ad_1	Ad_2
300	$1/2 \rightarrow$	P_1	**125**	**25**
	$1/2 \rightarrow$	P_2	0	**150**

Planning with $H = 300 \rightarrow$ **Far-sighted**
P_1: 73% on Ad_1, 17% on Ad_2
P_2: 100% on Ad_2

Figure 6: The effect of the planning horizon H. Ad_1 and Ad_2 start at time 0 and have the same unit revenue per click. The click probabilities are $p_{1,1}) = 0.8, p_{1,2} = 0.1$ for the user profile P_1 and $p_{2,1} = 0.8, p_{2,2} = 0.5$ for the user profile P_2. Both profiles have the same visit probability.

The oldest reference we were able to spot is Langheinrich et al. [10] who mixed a linear program with a simple estimation of CTR to select advertisements to display. In this work, no attention is paid to the exploration/exploitation trade-off and more generally, the problem of the estimation of the CTR is very crudely addressed. Then, Abe and Nakamura [1] introduce a multi-arm bandit approach to balance exploration with exploitation. Their work is based on display proportions, that is unlimited resources; they also deal with a static set of advertisements. This was later improved by Nakamura and Abe [15] who deal with the important problem of multi-impression of ads on a single page; they also deal with the exploration/exploitation trade-off by way of Gittins indices. Ideas drawn from their work on multi-impression may be introduced in ours to deal with that issue.

Aiming at directly optimizing the advertisement selection, side information (information about the type of advertisement, page, date of the request, ...) is used to improve the accuracy of prediction in several recent papers Kakade et al. [7], Langford and Zhang [9], Li et al. [12], Pandey et al. [17], Wang et al. [19]. Interestingly, Pandey and Olston [16] also deals with the multi-impression problem. However, all these works do not consider finite budget constraints, and finite lifetime constraints, as well as the continuous creation of new advertising campaigns; they also do not consider the CTR estimation problem. Very recently, Li et al. [12] focuses on the exploration/exploitation trade-off and proposes interesting ideas that may be combined to ours (varying ε in the ε-greedy strategy, and taking into account the history of the displays of an advertisement). Though not dealing with advertisement selection but news selection, which implies that there is no revenue maximization, and no click budget con-

straint, but merely maximization of the amount click, Agarwal et al. [4], Li et al. [11] investigate a multi-arm bandit approach.

Some works have specifically dealt with the accurate prediction of the CTRs, either in a static setting (Richardson et al. [18]), or dealing with a dynamic setting, and non stationary CTRs (Agarwal et al. [3]). Agarwal et al. [2], Wang et al. [20] also use a hierarchically organized side information on advertisements and pages.

A rather different approach is that of Mehta et al. [14] who treated this problem as an on-line bipartite matching problem with daily budget constraints. However, it assumed that we have no knowledge of the sequence of appearance of the profile, whereas in practice we often have a good estimate of it. Mahdian and Nazerzadeh [13] tried then to take advantage of such estimates while still maintaining a reasonable competitive ratio, in case of inaccurate estimates. Extensions to click budget were discussed in the case of extra estimates about the click probabilities. Nevertheless, the daily maximization of the income is not equivalent to a global maximization.

4 Experiments

4.1 The Model

Our approach was tested on a toy-model designed with experts from Orange Labs, the research division of an important commercial web actor with tens of millions of page views per day over multiple web sites, to fit the real-world problem. We took care that each advertisement campaign has its own characteristics that more or less appeal to the different visits. The model assumes that each advertising campaign A_k has a *base click probability* p_k that is sampled from a known distribution (*e.g.* uniform in an interval, or normally distributed with a certain mean and variance). As clicking on an advertisement is in general a rare event, the base click probabilities are typically low (around 10^{-4}). The click probability of a visitor from a particular user profile is then set to $p_{i,k} = p_k \gamma^{\mathbf{d}-1}$ where $\gamma > 1$ is a predefined multiplicative coefficient and the random variable $\mathbf{d}$ is sampled from the discrete probability distribution with parameter n that has the following probability mass function $Pr[\mathbf{d} = x] = 2^{n-x}/(2^n - 1), 1 \leq x \leq n$. When n is small, all advertising campaigns will have similar click probabilities that are close to the base click probability; as n increases, some advertising campaigns will have significantly higher click probabilities for some but not all of the user profiles. Note that, the number of such assignments will be exponentially low; if γ is taken as fixed then there will be twice as many advertising campaigns with click probability p compared to those with click probability γp. This allows us to effectively model situations in which a small number of advertising campaigns end up being popular in certain user profiles. In the experiments we used two values for the *gamma* parameter, 2 and 4; experts recommended use of the latter value, but as we will see shortly having a higher γ value may be advantageous for the greedy policy. The value of n is varied between 2 and 6.

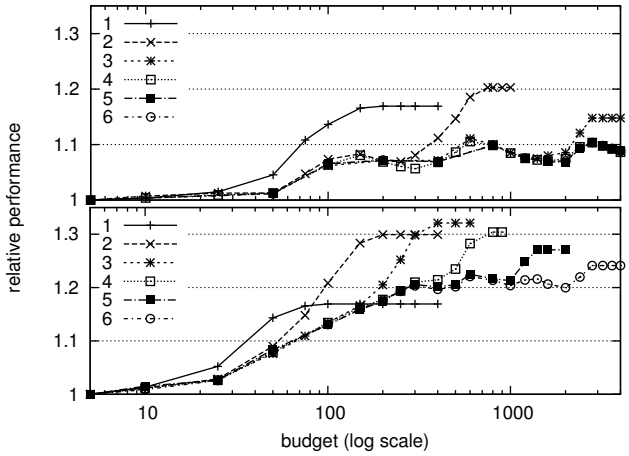

Figure 7: The relative performance of the HLP policy with respect to the HEV policy for different values of the click probability generation parameter n under the static setting with one user profile and 40 advertising campaigns. The multiplicative coefficient γ is 2 (bottom) and 4 (top).

4.2 The Experiments

Similar to the way that we introduce the proposed method in Section 2, in the experiments we will also proceed from simpler settings to more complex ones. Due to the space limitations, we opted to focus on core measures and therefore omit some of the extensions that have been discussed in the text. We begin with the static setting with full information. In this setting, we consider a fixed time horizon of one day which is assumed to be equivalent to 4 million page visits. The distribution of user profiles is uniform and the budget and lifetime of advertisement campaigns are also sampled uniformly from fixed intervals. In order to determine the starting times of advertising campaigns, we partitioned the time horizon into M equally spaced intervals (in our case 80) and set the starting time of each advertisement to the starting time of an interval chosen randomly such that the ending times do not exceed the fixed time horizon. The base click probability is set to 0.0001. We solved the optimization problem every 10000 steps.

Fig. 7 shows the relative performance of HLP policy with respect to the HEV policy for different values of the click probability generation parameter n and budget for the case in which there is a single user profile, and 40 advertising campaigns with an average lifetime of $1/10^{th}$ of the time horizon; all advertisement campaigns have the same budget. We can make two observations, all other parameters being fixed HLP is more effective with increasing budgets, and the performance gain depends largely on the value of γ. For $\gamma = 4$, which is considered to be a realistic value by experts, and reasonable budgets the greedy policy would perform well. A similar situation also arises when the number of advertisement campaigns is low, whereas increasing the number of user profiles favors planning (Fig. 8).

Next, we tried longer static settings of over one week period with and without full information in which the advertising campaign lifetimes and their budget

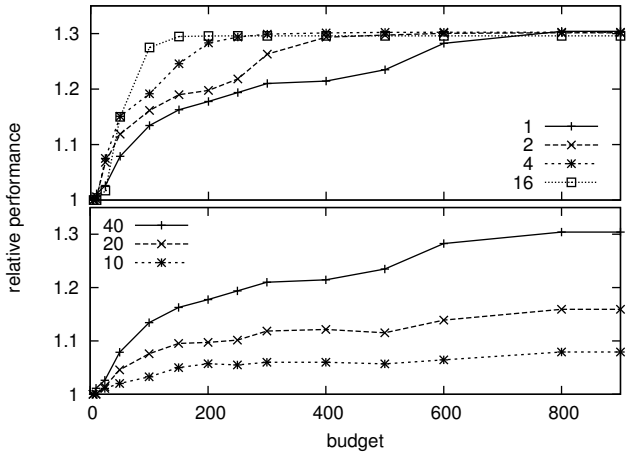

Figure 8: The effect of the number of user profiles (top) and the number of advertising campaigns (bottom) when other parameters are kept constant and n and γ are set to 2 and 4, respectively.

are more realistic (2-5 days, 500-4000 clicks). The campaigns are generated on a daily basis at the beginning of a run, *i.e.* a set of 7-9 new advertisement arrives at every 4 million steps. We tested different values for the click probability generation parameters. There were 8 user profiles with equal visit probabilities. As presented in Fig. 9 (a), in this setting although HLP policy performs better than the greedy policy, the performance gain stays limited. While the greedy policy quickly exploits and consumes new advertisements as they arrive, HLP tends to keep a consistent and uniform click rate at the beginning and progressively becomes more greedy towards the end of the period (Fig. 10). Fig. 9 (b) shows the effect of the planning horizon, *i.e.* when we focus on near future and ignore or do not have information about distant events; note that, this prominently depends on the degree of interaction between the advertising campaigns and in this and other experiments we observed that being very far-sighted may not be necessary.

Finally, we conducted experiments in the dynamic setting with partial information where the probabilities are not known in advance but estimated online. We employed ε-greedy exploration mechanism with different values of ε and maximum a posteriori estimation with Beta priors. The results show that HLP can perform better than HEV, however for both policies the chosen set of parameters influences the outcome (Fig. 11).

5 Conclusion and future work

In this paper, we considered the advertisement selection problem on web pages. Aiming at considering the problem in the most realistic setting, and providing effective and efficient algorithms to perform this selection on a production system, we have formalized the problem by providing a series of increasing complexity settings. This let us discuss various algorithmic approaches, and clearly identify the issues. While defining this set of problems, we provided a way to

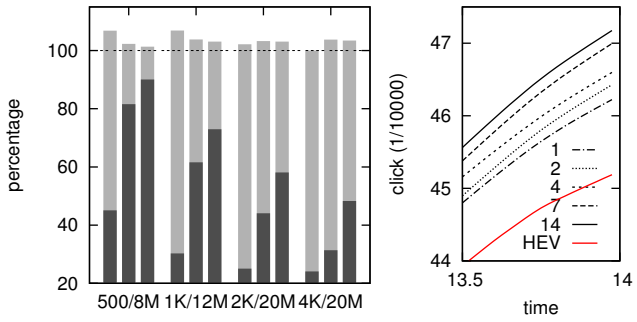

Figure 9: (a) The performance of the random (dark gray and lowest) and the HLP (light gray and highest) policies with respect to the HES policy under the 7 days static setting for different budget (500 to 4000), lifetime (2-5 days) and generation parameter n values. The three sets of bars in each group corresponds to the case where n is taken as to 2, 4, and 6 in that order. (b) The effect of horizon (1, 2, 4, 7 and 14 days) in the 14 days static setting with full information. Since we are not in the dynamic setting, using less information than available hinders the performance.

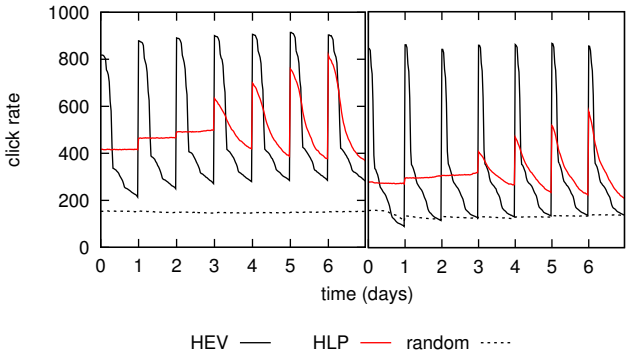

Figure 10: The moving average of click rate for different policies under the 7 day static setting; the lifetime of advertising campaigns is 5 days (20 million time steps) and their budgets are either 4000 (left) or 2000 (right).

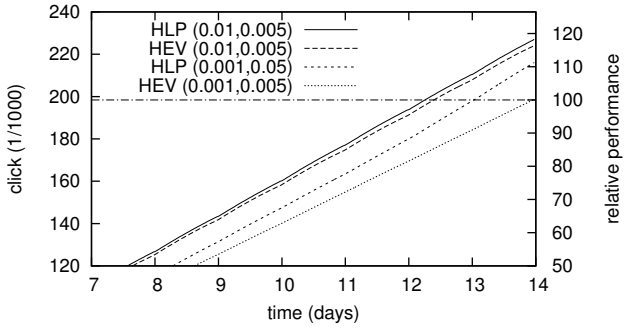

Figure 11: The performance of HEV and HLP algorithms in the dynamic setting with partial information using ε-greedy exploration. The numbers in paranthesis denote the values of the parameter of the Beta prior and ε.

effectively tackle this problem, and provided an experimental study of some of their key features. The experimental study is based on a realistic model, carefully designed with a major commercial Internet portal.

We have shown that optimizing ad display handling finite budgets and finite lifetimes, in a dynamic and non stationary setting, is feasible within realistic computational time constraints. We have also given some insights in what can be gained by handling this constraint, depending on the properties of the advertisements to display. We have also exhibited that lifetime of the advertisements impact the overall performance, and so should be taken into account into the pricing policy. Moreover our work may be seen as a part of a decision aid tool. For instance, it can help to price the advertisements in the case in which a fraction of the advertising campaigns are in the "cost per display" model, while the rest is in the cost per click model. This is rather easy because the LP solution provides an estimation of the revenue for each visitor profile.

To address the question of the title *Should we be Greedy?*, our work shows that it depends on the parameters of the advertisements we have to use. Fig. 7 illustrates how these parameters interact. To summarize, we may say that if there are few overlapping advertisements, or many advertisements with long lifetimes and good click rates, then we should be greedy. Between these two extreme solutions, one should consider the constraints associated to each advertisement campaign.

This work calls for many further developments. A possibility is to solve the problem from the perspective of the advertiser, *i.e.* help the advertiser to set the value of a click, and adjust it optimally with respect to his/her expected number of visitors. It would be equivalent to a local sensitivity analysis of the LP problem. A more difficult issue is that of handling multiple advertisement displays on the same page. It may be possible to handle them by estimating the correlation between the advertisements, and trying to update multiple click probabilities at the same time. Some very recent developments in the bandit setting (Koolen et al. [8]) are interesting in this regard.

We are also willing to draw some theoretical results on how far from the optimal strategy we are. Dealing with finite resources, under finite time constraints, in a dynamic setting makes that kind of study very difficult. An other

work originates from the analysis of some real web server logs. We have already been very slightly using such source of information, but much more has to be done.

Finally, we think we should go towards learning on-line the profiles of the visitors depending on their click behavior instead of having pre-existing ones.

Acknowledgments

This research was supported, and partially funded by Orange Labs, under externalized research contract number CRE number 46 146 063 - 8360, and by Ministry of Higher Education and Research, Nord-Pas de Calais Regional Council and FEDER through the "Contrat de Projets Etat Region (CPER) 2007-2013", and the contract "Vendeur Virtuel Ubiquitaire" of the "Pôle de compétitivité Industries du Commerce".

The model used in this paper, as well as its parameters have been designed with, and validated by, Orange Labs, to keep up with the essential characteristics of the real problem.

The realization of simulations were carried out using the Grid'5000 experimental testbed, an initiative from the French Ministry of Research, INRIA, CNRS and RENATER and other contributing partners.

References

[1] N. Abe and A. Nakamura, "Learning to Optimally Schedule Internet Banner Advertisements," in *Proc. of the 16th International Conference on Machine Learning*, 1999, pp. 12–21.

[2] D. Agarwal, A. Broder, D. Chakrabarti, D. Diklic, V. Josifovski, and M. Sayyadian, "Estimating rates of rare events at multiple resolutions," in *Proc. of the 13th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining*, 2007.

[3] D. Agarwal, B.-C. Chen, and P. Elango, "Spatio-temporal models for estimating click-through rate," in *Proc of the 18th Int'l World Wide Web Conference (WWW)*, Apr. 2009, pp. 21–30.

[4] D. Agarwal, B. Chen, and P. Elango, "Explore/exploit schemes for web content optimization," in *Proc. of the 2009 IEEE Int'l Conf. on Data Mining (ICDM)*, 2010, pp. 661–670.

[5] D. P. Bertsekas, *Dynamic Programming and Optimal Control, Vol. II.* Athena Scientific, 2007.

[6] O.-C. Granmo, "A Bayesian Learning Automaton for Solving Two-Armed Bernoulli Bandit Problems," in *Proc. of the 7th International Conference on Machine Learning and Applications (ICML-A).* IEEE Computer Society, 2008, pp. 23–30.

[7] S. M. Kakade, S. Shalev-Shwartz, and A. Tewari, "Efficient Bandit Algorithms for Online Multiclass Prediction," in *Proc. of the 25th International Conference on Machine Learning (ICML).* New York, NY, USA: ACM, 2008, pp. 440–447.

[8] W. M. Koolen, M. K. Warmuth, and J. Kivinen, "Hedging structured concepts," in *Proceeding of the 23rd Annual Conference on Learning Theory (COLT)*, 2010, pp. 93–105.

[9] J. Langford and T. Zhang, "The Epoch-Greedy Algorithm for Multi-armed Bandits with Side Information," in *20th Advances in Neural Information Processing Systems (NIPS)*, J. Platt, D. Koller, Y. Singer, and S. Roweis, Eds. MIT Press, 2008, pp. 817–824.

[10] M. Langheinrich, A. Nakamura, N. Abe, T. Kamba, and Y. Koseki, "Unintrusive customization techniques for web advertising," *Computer Networks*, vol. 31, Jan. 1999.

[11] L. Li, W. Chu, J. Langford, and R. Schapire, "A contextual-bandit approach to personalized article recommendation," in *Proc. of the 19th Int'l World Wide Web Conference (WWW)*, apr 2010.

[12] W. Li, X. Wang, R. Zhang, Y. Cui, J. Mao, and R. Jin, "Exploitation and exploration in a performance based contextual advertising system," in *Proc. of the 16th ACM SIGKDD Int'l Conf on Knowledge Discovery and Data Mining*, 2010.

[13] M. Mahdian and H. Nazerzadeh, "Allocating Online Advertisement Space with Unreliable Estimates," in *ACM Conference on Electronic Commerce*, 2007, pp. 288–294.

[14] A. Mehta, A. Saberi, U. Vazirani, and V. Vazirani, "Adwords and Generalized On-line Matching," in *Proc. of the 46^{th} Annual IEEE Symposium on Foundations of Computer Science (FOCS)*. IEEE Computer Society, 2005, pp. 264–273.

[15] A. Nakamura and N. Abe, "Improvements to the linear programming based scheduling of web advertisements," *Electronic Commerce Research*, vol. 5, no. 1, pp. 75–98, Jan. 2005.

[16] S. Pandey and C. Olston, "Handling Advertisements of Unknown Quality in Search Advertising." in *18^{th} Advances in Neural Information Processing Systems (NIPS)*, B. Schölkopf, J. Platt, and T. Hoffman, Eds. MIT Press, 2006, pp. 1065–1072.

[17] S. Pandey, D. Agarwal, D. Chakrabarti, and V. Josifovski, "Bandits for Taxonomies: A Model-based A pproach," in *Proc. of the 7^{th} SIAM International Conference on Data Mining*, 2007.

[18] M. Richardson, E. Dominowska, and R. Ragno, "Predicting clicks: Estimating the click-through rate for new ads," in *Proc. of the 16^{th} Int'l World Wide Web Conference (WWW)*, 2007.

[19] C.-C. Wang, S. Kulkarni, and H. Poor, "Bandit Problems With Side Observations," *IEEE Transactions on Automatic Control*, vol. 50, no. 3, pp. 338–355, 2005.

[20] X. Wang, W. Li, Y. Cui, B. Zhang, and J. Mao, "Click-through rate estimation for rare events in online advertising," in *Online Multimedia Advertising: Techniques and Technologies*. IGI Global, 2010.

Contents

Centre de recherche INRIA Lille – Nord Europe
Parc Scientifique de la Haute Borne - 40, avenue Halley - 59650 Villeneuve d'Ascq (France)

Centre de recherche INRIA Bordeaux – Sud Ouest : Domaine Universitaire - 351, cours de la Libération - 33405 Talence Cedex
Centre de recherche INRIA Grenoble – Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier
Centre de recherche INRIA Nancy – Grand Est : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex
Centre de recherche INRIA Paris – Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex
Centre de recherche INRIA Rennes – Bretagne Atlantique : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex
Centre de recherche INRIA Saclay – Île-de-France : Parc Orsay Université - ZAC des Vignes : 4, rue Jacques Monod - 91893 Orsay Cedex
Centre de recherche INRIA Sophia Antipolis – Méditerranée : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399