

Improvements in Practical Aspects of Optimally Scheduling Web Advertising

Atsuyoshi Nakamura
NEC Internet Systems Research Laboratories
4-1-1 Miyazaki, Miyamae-ku
Kawasaki 216-8555 JAPAN
atsu@ccm.cl.nec.co.jp

ABSTRACT

We addressed two issues concerning the practical aspects of optimally scheduling web advertising proposed by Langheinrich et al. [5], which scheduling maximizes the total number of click-throughs for all banner advertisements. One is the problem of *multi-impressions* in which two or more banner ads are impressed at the same time. The other is inventory management, which is important in order to prevent over-selling and maximize revenue. We propose efficient methods which deal with these two issues.

Categories and Subject Descriptors

C.3 [Computer Systems Organization]: Special-purpose and Application-based Systems; I.2 [Computing Methodologies]: Artificial Intelligence; I.2.8 [Artificial Intelligence]: Problem Solving, Control Methods, and Search—Scheduling

General Terms

Algorithms, Management, Design

Keywords

On-line Advertisement, World-Wide Web, Electronic Commerce, Optimization, Inventory Management

1. INTRODUCTION

The idea of optimal scheduling of web ads so as to maximize the number of click-throughs was proposed by Langheinrich et al. [5]. They reduce the problem of maximizing the number of click-throughs to a transportation problem, a kind of linear programming (LP) problem, which is known to be one that can be solved efficiently by making use of special features of the problem.

For this LP model of web advertising, some proposals to improve the method were made by Abe and Nakamura [1] and Tomlin [6]. Abe and Nakamura studied adaptive methods of estimating click-through rates, which is very important especially in the initial phase. They also studied clustering of target attribute values, which helps reduce the dimensionality. Tomlin pointed out “over-targeted” solutions of the LP model, and proposed an entropy model that incorporates a form of randomization.

Copyright is held by the author/owner(s).
WWW2002, May 7–11, 2002, Honolulu, Hawaii, USA.
ACM 1-58113-449-5/02/0005.

The effectiveness of the LP model has been demonstrated not only in large-scale simulation experiments [5, 1] but also in an experiment using a web site opened to the public [3].

In this paper, we describe our study of two practical aspects of optimally-scheduled web advertising using the LP model. One is how to deal with a *multi-impressions* in which two or more banner ads are impressed at the same time. Most of us are familiar with web pages on which multiple banner ads are displayed. What should be considered is that all the banner ads displayed on a web page at the same time must be distinct, but the same ads can be selected if we select ads randomly according to display probabilities scheduled in the LP model. In one selection of a set of ads on a web page, our solution makes use of an ad queue to which an ad is added when it is selected more than once through the display probabilities. In the selection of the next set, we first select ads from the queue and the rest through the display probabilities. Note that, if no more than three ad banners are displayed on a page, any display probability must be at most $1/3$ so as not to cause queue overflow. Thus, when we calculate display probabilities, such constraints must be taken.

The other aspect we studied is inventory management. Over-selling causes no solution for the LP problem. In order to ensure solution existence, it is important not to over-sell impressions to advertisers. What makes this difficult is the constraints that advertisers specify. For example, consider a case where one advertiser requests that his ad be displayed on a sports page only while another advertiser requests that his ad be displayed in the afternoon only. Note that here page views of a sports page in the afternoon satisfy the both constraints. In this case, to check whether 1000-impression-selling of the sports page causes over-selling or not, we must also consider impressions sold for the afternoon constraint. Of course, this can be checked by solving the LP problem and determining whether solutions exist or not. Its problem, however, is that we cannot grasp how many impressions remain for the sports-page constraint. To address this issue, we propose a method of calculating the number of remaining sellable impressions, which is formalized as another LP problem.

This paper is organized as follows. In the next section, we review the LP model proposed by Langheinrich et al. Then, we explain our solutions for the multi-impression issue (Sec. 3) and inventory management (Sec. 4). Finally, we conclude with some relevant remarks.

time of day	page category	number of page views	click-through rate (%)		
			ad 1	ad 2	ad 3
afternoon	sports	10,000	2.2	1.1	1.0
afternoon	not sports	10,000	2.2	2.1	1.0
not afternoon	sports	5,000	2.2	2.1	2.0
not afternoon	not sports	5,000	2.2	2.1	2.0

Table 1: Case that the local strategy fails

2. LP MODEL

In this section, we review the LP model proposed by Langheinrich et al. [5] and explain the idea of introducing ‘degree of importance’ to settle a certain problem. In their original paper, they considered only keyword-driven and page-driven ads. For any attribute, however, we can also consider attribute-driven ads. Furthermore, we can consider attribute-set-driven ads for any attribute set. So, here, we consider a fixed set of attributes such as the set of ‘time of day’ and ‘page category’.

Before describing the formalization, we first show an example which helps to illustrate the problem and the need for an LP solution. Assume that the accurately estimated numbers of page views for combinations of attribute values (afternoon, sports), (afternoon, not sports), (not afternoon, sports) and (not afternoon, not sports) are 10,000, 10,000, 5,000 and 50,00, respectively. Also assume that there are three ads for each of which 10,000 impressions have been promised, and that the accurately estimated click-through rates of these ads for the combinations of attribute values are as shown in Table 1.

Consider the local strategy that selects an ad with the highest click-through rate among the ads whose promised number of impressions has not been achieved yet. We can easily see that this local strategy, which many people seem to think is effective, does not work in the above case as follows. Assume that page views for all combinations of attribute values occur randomly¹. The local strategy always selects ad 1 for the first 10,000 page views, ad 2 for the second 10,000 page views and ad 3 for the last 10,000 page views, because (the click-through rate of ad 1) > (the click-through rate of ad 2) > (the click-through rate of ad 3) holds for all combinations of attribute values. As a result, we find that the actual click-through rates for ad 1, ad 2 and ad 3 are 2.2%, 1.76...% and 1.33...%, and the total click-through rate for all ads is 1.76...%, which is the same rate as one obtained by random selection.

In the above case, according to the optimal display schedule in the LP model, ad 1 is always selected for (afternoon, sports), ad 2 for (afternoon, not sports) and ad 3 otherwise. The total click-through rate of this optimal schedule is 2.1% and the actual click-through rates for ad 1, ad 2 and ad 3 are 2.2%, 2.1% and 2.0%, respectively.

Now let us describe the formalization. Let $i (= 1, \dots, n)$ denote combinations of attribute values and $j (= 1, \dots, m)$ denote ads. We define k_i , h_j and $c_{i,j}$ as follows:

k_i : Estimated rate of page views for combination i of attribute values (normalized over all combinations),

h_j : Desired display rate for ad j (normalized over all ads),

$c_{i,j}$: Estimated click-through probability for ad j and combination i of attribute values.

The estimation of k_i is calculated based on the number of recent page views for combination i of attribute values. Desired display rate h_i is calculated from the contracted number of impressions by subtracting the number of actual impressions so far and considering the remaining contracted period. The estimation of $c_{i,j}$ is calculated based on the recent click-through rate for combination i of attribute values and ad j .

Let $d_{i,j}$ denote the display probability of ad j for combination i of attribute values. The problem we have to solve is the following Problem 1.

PROBLEM 1. *Maximize*

$$\sum_{i=1}^n \sum_{j=1}^m c_{i,j} k_i d_{i,j} \quad (1)$$

under the conditions

$$\sum_{i=1}^n k_i d_{i,j} = h_j \quad \text{for } j = 1, \dots, m \quad (2)$$

$$\sum_{j=1}^m d_{i,j} = 1 \quad \text{for } i = 1, \dots, n \quad (3)$$

$$d_{i,j} \geq 0 \quad \text{for } i = 1, \dots, n, j = 1, \dots, m \quad (4)$$

Problem 1 is a “transportation problem” (see e.g. [4]) and efficient algorithms (see e.g. [2]) for this problem are known. In this model, constraints specified by advertisers are easily incorporated. Assume that ad j_0 is NOT allowed to be displayed for combination i_0 of attribute values by a contract with an advertiser. Then, you incorporate this constraint into Problem 1 by considering pair $(i, j) \neq (i_0, j_0)$ only. Another approach is to set c_{i_0, j_0} to -1 , which prevents the problem from having no solution. (See [5].)

One of the problems in the above model is that the optimal solution of Problem 1 maximizes ‘total’ click-through rate for ‘all’ ads. This means that some ads may have to put up with low click-through rate in order to obtain high total click-through rate. Consider the two ads for each of which 10,000 impressions have been promised. Assume that the accurately estimated click-through rates (%) of these ads for combination 1 and 2 of attribute values are as shown in the following table.

	ad 1	ad 2
combination 1 of attribute values	4.0	2.5
combination 2 of attribute values	2.0	1.0

¹Though this assumption is not realistic because an attribute value ‘not afternoon’ does not appear in the afternoon, it is not so unreasonable considering on the average.



Figure 1: Example of multi-impressions

Then, the optimal solution² is the one that always displays ad 1 for combination 1 and always displays ad 2 for combination 2. By this solution, ad 1 has high click-through rate 4.0% while ad 2 put up with low click-through rate 1.0. This is a problem if the advertiser of ad 2 is more important than the advertiser of ad 1. This problem can be settled to some extent by introducing ‘degree of importance’ $g_j > 0$ for each ad j . The default value of g_j is 1.0, and the greater the value is, the more important ad j is. A modification of objective function (1) incorporating degree of importance is

$$\sum_{i=1}^n \sum_{j=1}^m g_j c_{i,j} k_i d_{i,j}. \quad (5)$$

If degrees of importance for ad 1 and ad 2 are 1.0 and 2.0, respectively, the optimal solution for the above example is the one that always displays ad 2 for combination 1 and always displays ad 1 for combination 2. This decreases the click-through rate of ad 1 from 4.0 to 2.0, but increases the one of ad 2, which is supposed to be more important than the one of ad 1, from 1.0 to 2.5. In practical operation, if you find that a certain ad of an important advertiser has low click-through rate after a part of the contract period has passed, you can improve the click-through rate by increasing degree of importance for the ad.

3. MULTI-IMPRESSIONS

In this section, we explain our method of realizing multi-impressions while maintaining the scheduled display probabilities.

Figure 1 is an example of multi-impressions, where two ad banners are seen at the top of a page displayed on a web browser. One important thing to note here is that the selected two ads must be distinct.

Assume that we want to select two different ads according to the following display probabilities.

ad 1	ad 2	ad 3
0.5	0.4	0.1

²Note that the optimal solution depends on the promised number of impressions for each ads. If ad 2 has been promised to be displayed twice as many times as ad 1, then the optimal solution is the one that always displays ad 2 for combination 1 and always displays ad 1 for combination 2.

Function ADSelect

Input:

- N : requested number of ads
- i : combination of attribute values

Output:

- S : set of N ads

Functions for queue operations:

$\text{pop}(i)$: the first ad in the queue for combination i of attribute values (the first ad is removed from the queue after each call of this function)

$\text{append}(i, f)$: append ad f to the queue for combination i of attribute values

$\text{size}(i)$: the number of ads in the queue for combination i of attribute values

Function for an operation concerning scheduled display probabilities:

$\text{select}(i)$: one ad selected randomly according to the scheduled display probabilities for combination i of attribute values

```

M = size(i)
e = 0
S = ∅
do {
  if (e < M) then f = pop(i)
  else f = select(i)
  if f ∈ S then append(i, f)
  else S = S ∪ {f}
  e = e + 1
} while the number of elements in S is less than N
return S

```

Figure 2: Ad selection algorithm for multi-impressions

Consider the method of selecting ads according to the probability distribution restricted to ads that have not been selected yet. In the above example, the first one is selected from the three ads according to the probabilities above, and if the first one is ad 1, then the second one is selected from the set of ad 2 and ad 3 according to the probability 0.8 and probability 0.2, respectively, which are calculated from the above probabilities by restricting to ad 2 and ad 3 and normalizing. It can be easily verified that, when two ads are selected using this method, the display proportion of ad 1, ad 2 and ad 3 is 20:19:6, which is different from the desired proportion 5:4:1. Thus, in our solution, we do not restrict the probability distribution but append the ads which have already been selected to a queue and select from the queue later. A detailed description of our selection algorithm ADSelect is shown in Figure 2.

Function ADSelect uses one queue for each combination of attribute values. Given a requested number N of ads and the combination i of attribute values, ADSelect returns set S of N different ads according to a scheduled probability distribution for combination i . ADSelect first selects as many different ads from the queue as possible ($\text{pop}(i)$), and

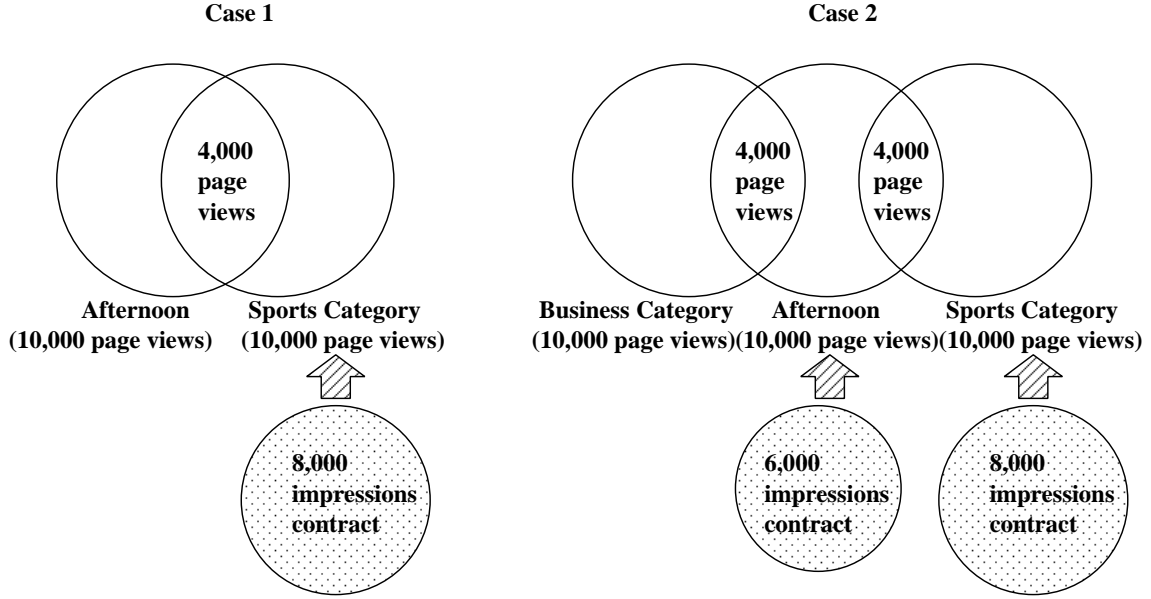


Figure 3: Two example cases of inventory status

then selects according to the scheduled probability distribution ($\text{select}(i)$). If selected ad f has already been selected ($f \in S$), ADSelect appends it to the queue ($\text{append}(i, f)$). The algorithm repeats the selection until selected set S of ads contains the requested number of ads.

There is one thing that must be borne in mind, however. We have to determine the scheduled display probabilities so as not to exceed $1/N$, where N is the number of ads displayed on the page at the same time. If this rule is broken, the possibility of queue overflow becomes very high. The rule is formulated as follows using the notation in the previous section:

$$d_{i,j} \leq 1/N \text{ for } i = 1, \dots, n \text{ and } j = 1, \dots, m. \quad (6)$$

Note that this constraint moderates the ‘over-targeted’ feature of the LP solution which is pointed out by Tomlin [6] when $N \geq 2$. Since Problem 1 is still a transportation problem even if these constraints are incorporated, modifying Problem 1 with these constraints enables it to be efficiently solved.

4. INVENTORY MANAGEMENT

Web ad agencies and companies that host web sites want to contract for as many ad impressions as possible to earn as much ad revenue as possible. However, over-selling results in additional cost and reduced revenue. The important thing for preventing over-selling and maximizing revenue is to grasp how many sellable impressions remain. Assuming that the estimated number of page views is accurate³, this is easy if no constraints are specified by advertisers. However, this becomes non-trivial when such constraints are taken into account.

³Accurately estimating the number of page views itself is an important issue, but this is not the focus of our paper.

Consider Case 1 in Figure 3, where the estimated numbers of page views for afternoon and sports-category constraints are 10,000 each and 4,000 of them satisfy both constraints. Assume that there exists a contract which requests 8,000 impressions on sports-category pages only. In this case, 8,000 sellable impressions remain for afternoon constraint when no other contracts exist. In this manner, calculation of the remaining sellable impressions for a certain constraint t needs to take contracts for other constraints which overlap constraint t into consideration. Should only overlapping constraints be considered? Consider Case 2 in Figure 3, where the estimated number of page views is 10,000 for each of three constraints and the afternoon constraint overlaps the sports-category and business-category constraints while sports-category and business-category constraints do not overlap. The estimated numbers of page views for the two overlaps are 4,000 each. Assume that there exist two contracts, one requesting 8,000 impressions on sports-category pages only and the other requesting 6,000 impressions in the afternoon only. In this case, 8,000 sellable impressions for business-category constraint remain when no other contracts exist. Thus, the sports-category-constraint contract affects the number of remaining sellable impressions for the business-category constraint, which does not overlap the sports-category constraint.

We formalize this problem as follows. Assume that we want to know the number of remaining impressions for constraint t . Let $i (= 1, \dots, n)$ denote other constraints and $j (= 1, \dots, m)$ denote attribute subspaces divided by all constraints. Define N_i and P_j as follows:

- N_i : Number of impressions contracted for constraint i ,
- P_j : Estimated number of page views for attribute subspace j .

Note that N_i is the total sum of the impressions for all contracts which specify constraint i . In order to equalize

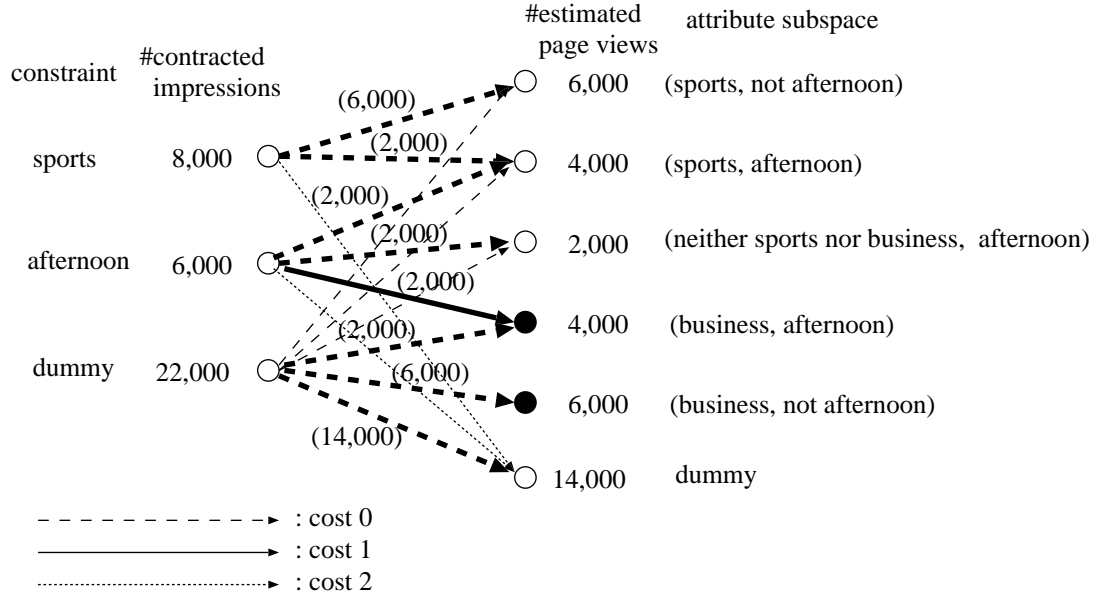


Figure 4: Solution of case 2 using our formalization

the sum of N_i and the sum of P_j , we introduce dummy constraint $n+1$ and dummy attribute subspace $m+1$ and let N_{n+1} be $\sum_{j=1}^m P_j$ and P_{m+1} be $\sum_{i=1}^n N_i$. We assume that constraint $n+1$ contains every attribute subspace j and attribute subspace $m+1$ is contained in every constraint i . We define S and Q as follows:

$$S^{\text{def}} = \{(i, j) : \text{constraint } i \text{ contains attribute subspace } j\} \quad (7)$$

$$Q^{\text{def}} = \{j : \text{constraint } t \text{ contains attribute subspace } j\} \quad (8)$$

Let $x_{i,j}$ be the number of scheduled impressions for $(i, j) \in S$. Then, the problem is described as follows.

PROBLEM 2. *Minimize*

$$\sum_{j \in Q} \sum_{i: (i,j) \in S, i \neq n+1} x_{i,j} + \sum_{i: (i,m+1) \in S, i \neq n+1} 2x_{i,m+1} \quad (9)$$

under the conditions

$$\sum_{j: (i,j) \in S} x_{i,j} = N_i \text{ for } i = 1, \dots, n+1 \quad (10)$$

$$\sum_{i: (i,j) \in S} x_{i,j} = P_j \text{ for } j = 1, \dots, m+1 \quad (11)$$

$$x_{i,j} \geq 0 \text{ for } (i, j) \in S \quad (12)$$

Objective function (9) is seen as cost function

$$\sum_{(i,j) \in S} C_{i,j} x_{i,j},$$

where $C_{i,j}$ is a unit cost for (i, j) that is 2 for $i = 1, \dots, n$ and $j = m+1$, 1 for $i \neq n+1$ and $j \in Q$ and 0 otherwise. The first term of function (9) is the cost of the impressions for the contracts for other constraints which are scheduled for page views available for constraint t . Since the purpose

is to know the maximum estimated number of page views available for constraint t , we want this cost to be as low as possible. The second term of function (9) is overflow cost that charges for the impressions scheduled for the page views of the dummy attribute subspace. If there exists $i \neq n+1$ such that $x_{i,m+1} > 0$, this means over-selling for other constraints but does not mean over-selling for constraint t .

Let the solution of Problem 2 be $x_{i,j} = a_{i,j}$. Then, the number of remaining sellable impressions for constraint t is

$$\sum_{j \in Q} a_{n+1,j} - N_t, \quad (13)$$

where N_t is the number of impressions contracted for constraint t .

Since Problem 2 also falls into the “transportation problem” category, it can be solved efficiently.

Figure 4 shows a solution of case 2 using our formalization. Left nodes represent constraints i and right nodes represent attribute subspaces j . An arrow from a left node to a right node is drawn for each $(i, j) \in S$. An arrow with a coarse broken line, a continuous line and a fine broken line means that the unit cost is 0, 1 and 2, respectively. Black nodes means that the corresponding attribute subspaces belong to Q . A solution for the problem is represented by an assignment of the number of scheduled impressions to each arrow. An arrow with a bold line is the one to which non-zero number of scheduled impressions is assigned in the optimal solution (the assigned number is parenthesized in the figure). For the arrows from the dummy-constraint node to the black nodes, the assigned numbers of scheduled impressions are 2,000 and 6,000. Therefore, the number of remaining sellable impressions for business constraint is 8,000.

5. CONCLUDING REMARKS

We have provided solutions for two practical issues of optimally scheduling web advertising. Our solutions are not only

efficient but also easy to implement, and an actual system incorporating these solutions has already been developed.

There is one important issue relevant to inventory management we did not address in this paper – inventory estimation. Our solution for inventory management is based on the assumption that the estimated number of page views is accurate to some extent. In order to accurately estimate the amount of inventory, periodicity and adaptivity should be taken into consideration, while the most important thing is to find what information it depends on. In our current system, we adopt a simple method based on statistics but we hope and plan to develop a sophisticated method in the future.

6. REFERENCES

- [1] N. Abe and A. Nakamura. “learning to optimally schedule internet banner advertisements”. In *Proc. of the 16th International Conference on Machine Learning*, pages 12–22, 1999.
- [2] G. Bradley, G. Brown, and G. Graves. “design and implementation of a large scale primal transshipment algorithm”. *Management Science*, 24:1–34, 1977.
- [3] D. Chickering and D. Heckerman. “targeted advertising with inventory management”. In *Proc. of ACM Special Interest Group on E-Commerce (EC00)*, pages 145–149, 2000.
- [4] G. Dantzig. *“Linear Programming and Extensions”*. Princeton University Press, 1963.
- [5] M. Langheinrich, A. Nakamura, T. K. N. Abe, and Y. Koseki. “unintrusive customization techniques for web advertising”. *Computer Networks*, 31:1259–1272, 1999.
- [6] J. Tomlin. “an entropy approach to unintrusive targeted advertising on the web”. In *Proc. of the 9th International World Wide Web Conference*, 2000.