Prompt Mechanism for Online Auctions with Multi-unit Demands

Xiangzhong Xiang

Department of Computer Science, The University of Hong Kong, Hong Kong
xzxiang@cs.hku.hk

Abstract. We study the following TV ad placement problem: m identical time-slots are on sell within a period of m days and only one time-slot is available each day. Advertisers arrive online to bid for some time-slots to publish their ads. Typically, advertiser i arrives at the a_i'th day and wishes that her ad would be published for at most s_i days. The ad cannot be published after its expiration time, the d_i'th day. If the ad is published for $x_i \leq s_i$ days, the total value of the ad for advertiser i is $x_i \cdot v_i$; otherwise, the value of the ad to be published for each day diminishes and the total value is always $s_i \cdot v_i$. Our goal is to maximize the social welfare: the sum of values of the published ads. As usual in many online mechanisms, we are aiming to optimize the competitive ratio: the worst ratio between the optimal social welfare and the social welfare achieved by our mechanism.

Our main result is a competitive online mechanism which is truthful and prompt for the TV ad placement problem. In the mechanism, each advertiser is motivated to report her private value v_i truthfully and can learn her payment at the very moment that she wins some time-slots. Before studying the general case where the maximum demands s_i's are non-uniform, we study the special case where all s_i's are uniform and prove that our mechanism achieves a non-trivial competitive ratio of 5. For the general case where the maximum demands s_i's are non-uniform, we prove that our mechanism achieves a competitive ratio of $5 \cdot \lceil s_{max}/s_{min} \rceil$, where s_{max}, s_{min} are the maximum and minimum value of s_i's. Besides, we derive a lower bound of $\min\{\frac{v_{max}+v_{min}}{2v_{min}}, \frac{s_{max}}{s_{min}}\}$ on the competitive ratio for the general case, where v_{max}, v_{min} are the maximum and minimum value of v_i's.

1 Introduction

TV advertising has long been a profitable industry and advertising revenue provides a significant portion of the funding for most privately owned television networks. Advertisers are eager to promote a wide variety of goods, services and ideas by making use of advertisements. From the viewpoint of advertising, one television station, or publisher of advertisements, owns an inventory of time-slots, which are typically between 30 seconds to 120 seconds long and available daily. For an advertiser arriving online, she wishes that her advertisement could be published in a proper time-slot and repeated for a few days before the ad

P. Widmayer, Y. Xu, and B. Zhu (Eds.): COCOA 2013, LNCS 8287, pp. 117–128, 2013.

is expired. As the value of one ad is diminishing when the ad is broadcast for too many times and the advertiser has budget constraint, the repeating times of the ad should have upper limit. In the paper, we design an auction mechanism for publishers to allocate time-slots that maximizes the social welfare while satisfying advertisers' preferences.

1.1 The Problem

We study the following *TV ad placement problem*: m identical time-slots are on sale within a period of m days and only one time-slot is available each day. Advertisers arrive online to bid for some time-slots to publish their ads. Typically, advertiser i arrives at the a_i'th day and wishes her ad could be published for at most s_i consecutive days.[1] The ad cannot be published after its expiration time, the d_i'th day. If the ad is published for $x_i \leq s_i$ days, the total value of the ad for advertiser i is $x_i \cdot v_i$; otherwise, the value of the ad to be published for each day diminishes and the total value is always $s_i \cdot v_i$.[2] The goal is to maximize the social welfare: the sum of values of the published ads.

In this paper, we focus on designing truthful mechanisms. The information of all advertisers arriving in future is unknown in the online auction. However, we assume that when one advertiser i arrives, its arrival time a_i, expiration time d_i and maximum demand s_i are public information. The only private information is the value v_i and selfish advertisers may report false values to the publisher in order to maximize their profits. A truthful mechanism would motivate selfish advertisers to reveal their true values. This goal is usually achieved by means of making the payments collected from advertisers depending on the mechanism's outcome instead of advertisers' reported values.

Besides truthful, we also require our mechanisms to be *prompt*, which means that any advertiser that wins some time-slots could always learn her payment immediately after winning these time-slots. Prompt mechanisms are firstly proposed in [10]. For mechanisms that are not prompt, three main disadvantages are discussed in [10]: (1) A winning advertiser does not know how much money she has spent and thus does not know how much money she has left. She cannot use her remaining money to take part in another auction. (2) A winning advertiser may pay long after she won her time-slots. If she is not honest, she can deny paying the money while her ad has already been published. (3) A winning advertiser essentially provides the publisher with a "blank check" in exchange for time-slots. It is hard for advertisers to verify the exact calculation of their payments. In prompt mechanisms, all these disadvantages are avoided, as any advertiser can learn her payment when her ad begins to be displayed.

[1] In the whole paper, we consider the scenario that each advertiser is only interested in publishing her ad on some consecutive days. Note that even if advertisers can publish ads on days which are not consecutive, our prompt mechanism still works and has the same competitive ratio; but our lower bound on the competitive ratio does not hold in such model.

[2] To utilize time-slots efficiently, any rational publisher will not allocate advertiser i more than s_i time-slots.

1.2 Our Results

Our main result is an online mechanism which is truthful and prompt for the TV ad placement problem. In the mechanism, we partition all time-slots into groups evenly and all time-slots in one group can only be allocated to one advertiser. One advertiser can win at most one group even though she may demand more. We prove that each advertiser is motivated to reveal her true value in order to win one group of time-slots. Once an advertiser wins a group of time-slots, the price she pays for each time-slot in the group can be determined to be the least value she can report to win one group.

Before studying the general case where the maximum demands s_i's are non-uniform, we study the special case where all s_i's are uniform and prove that our mechanism achieves a non-trivial competitive ratio of 5. The crucial technique we use is to construct a novel mapping from the optimal solution to the solution produced by our online mechanism. For the general case where the maximum demands s_i's are non-uniform, we prove that our mechanism achieves a competitive ratio of $5 \cdot \lceil s_{max}/s_{min} \rceil$, where s_{max}, s_{min} are the maximum and minimum value of s_i's. If s_{max} is comparable with s_{min}, our mechanism is still very competitive. Besides, we derive a lower bound of $\min\{\frac{v_{max}+v_{min}}{2v_{min}}, \frac{s_{max}}{s_{min}}\}$ on the competitive ratio for the general case, where v_{max}, v_{min} are the maximum and minimum value of v_i's.

Remarks: Besides applied in TV ad placement problem, our online auction mechanism can solve other problems. Instead of time-slots in TV stations, the ad space may be a physical newspaper sheet with ads being published on it daily or a billboard that displays a set of ads on a fixed space with changes every specific time period. Our mechanism can also be used to solve the on-demand data broadcast problem [7, 13], in which clients make requests for data and all requests have deadlines. The server broadcasts the requested data at some time.

1.3 Related Work

The advertisement placement problem has been widely studied in recent years. In [12], the auction system used by Google for allocation and pricing of TV ads is introduced. The system uses a simultaneous ascending auction to generate a schedule of ads for TV companies daily. Online keyword advertising among multiple bidders with limited budgets is studied in [1, 6, 15]. In [1], bidders are offline while the ad places arrive online and an optimal $\frac{e}{e-1}$-competitive randomized algorithm is introduced. Another important branch about advertisement auction is designing truthful mechanisms, which is studied in [2, 3].

One classical technique used in many truthful mechanisms is the VCG payment scheme where each advertiser is charged the harm she causes to other advertisers and bidding the true value is the dominant strategy [14]. However, VCG cannot be applied to online problems because it requires computing the social welfare of the optimal allocation which cannot be computed in an online fashion. Moreover, even when the optimal allocation is known, the payment of a winner cannot be determined by VCG at the time when she wins her time-slots; her payment may depend on future events.

One special case of the TV ad placement problem is studied in [10]. They assume that advertisers arrive and depart over time. In contrast to our problem, each advertiser is interested in winning only one time-slot to publish her ad before she departs. For this special case, they show a 2-competitive prompt and truthful mechanism. The proof of truthfulness and the analysis of the competitive ratio are somewhat straightforward as any advertiser can win only one time-slot in both the optimal solution and the solution produced by their prompt mechanism. When analyzing the competitive ratio, they match at most two advertisers that win one time-slot in the optimal solution to exactly one advertiser that wins one time-slot in their prompt mechanism. In our problem, one advertiser can win more than one group of time-slots in the optimal solution. We need to design a novel matching from the optimal solution to the solution produced by our mechanism without collision. In the analysis, we map at most 5 advertisers in the optimal solution to one advertiser with higher value winning one group in our online mechanism. One difficulty in the analysis is that several advertisers may share one group of time-slots in the optimal solution. We adjust advertisers' arrival time and expiration time slightly and prove a competitive ratio of 5 successfully.

Azar *et al.* [4] studies a problem similar to that in [10]. In their auction problem, each ad can be published for at most once before the ad is expired. But each ad has an arbitrary size no greater than one and several ads can be published in one time-slot on condition that the total size does not exceed one. They design a truthful and prompt mechanism. The mechanism treats ads with size $< \frac{1}{2}$ and size $\geq \frac{1}{2}$ separately. They maintain a tentative schedule of ads for each day, and always prefers ads with higher density (i.e., the ratio of value to size). Their mechanism is proved to be 6-competitive.

In the full information setting, our TV ad placement problem is similar to the online scheduling problem with jobs arriving over time and having deadlines to be finished. Online scheduling with unit-length jobs has been studied in [5,8,9,11]. The best deterministic algorithm achieves a competitive ratio of $2\sqrt{2} - 1$ [11] and no deterministic algorithm can be better than $\frac{\sqrt{5}+1}{2}$-competitive [8]. The on-demand broadcasting problem is studied in [7,13], which can be reduced to online scheduling problem with jobs in different lengths. In [7], Chan *et al.* show an upper bound of $4\Delta + 3$ and a lower bound of $\Delta / \ln \Delta$ on the competitive ratio, where Δ is the ratio between the length of the longest and shortest jobs.

2 Preliminaries

Consider that m identical time-slots are on sale within a period of m days. Only one time-slot is available each day. There are advertisers arriving online and each advertiser has an ad to be published for a period before the ad is expired. One advertiser i can be represented by a tuple (s_i, v_i, a_i, d_i), where $s_i \in \mathbb{N}^+$ is the maximum number of consecutive time-slots the advertiser demands her ad would be published for, $v_i \in \mathbb{R}^+$ refers to the value of the ad if it is published for one day, and $a_i, d_i \in \mathbb{N}^+$ are the arrival and expiration time ($d_i - a_i + 1 \geq s_i$). The lower bound of all s_i's are s_{min} and known ahead of time. For any advertiser i, we assume that her value v_i is private while the other information is public. Advertiser i wishes that her ad is published in time window $W_i = [a_i, d_i]$. If $x \leq s_i$ consecutive time-slots in W_i are assigned to advertiser i and her payment for these time-slots is p_i, she gains a total value of $x \cdot v_i$ and a net profit of $x \cdot v_i - p_i$.

Our goal is to maximize the social welfare which is the total value of all the published ads. The auction mechanism should be:

(a) Incentive compatible: each advertiser i is incentive to reveal her true value v_i in the auction.
(b) Prompt: each advertiser can learn her total payment at the very moment that her ad begins to be published.

We say a mechanism is $c-$competitive if it can always achieve social welfare which is at least $\frac{1}{c}$ times of the optimal social welfare.

The further structure of the paper is as follows: in section 3 we show a lower bound on the competitive ratio of the problem. In section 4, we introduce a mechanism which is truthful and prompt. In section 5, we show the mechanism achieves a competitive ratio of 5 when all advertisers' demands are uniform and a competitive ratio of $5 \cdot \lceil s_{max}/s_{min} \rceil$ when their demands are non-uniform.

3 A Lower Bound on the Competitive Ratio

Before showing our main result, a prompt mechanism for the TV ad placement problem, we derive a lower bound on the competitive ratio for this problem. Assume that for any advertiser i, $s_{min} \leq s_i \leq s_{max}$ and $v_{min} \leq v_i \leq v_{max}$. For the case where all s_i's are one, a lower bound of 2 is shown in [10]. For the general case where s_i's are not uniform, we show that the lower bound of the competitive ratio is $\min\{\frac{v_{max}+v_{min}}{2v_{min}}, \frac{s_{max}}{s_{min}}\}$.

Theorem 1. *The competitive ratio of the TV ad placement problem is at least* $\min\{\frac{v_{max}+v_{min}}{2v_{min}}, \frac{s_{max}}{s_{min}}\}$.

Proof. To prove the lower bound of the competitive ratio, we measure the performance of any prompt mechanism against an adversary that knows all information and adjusts the input sequence according to the decisions made by the prompt mechanism. On the 1'st day, advertiser $u_1 : (s_{max}, v_{min}, 1, s_{max})$ arrives. Wlog, all the x time-slots in time window $[x_0, x_0 + x - 1]$ are allocated to u_1 at the x_0'th day in the prompt mechanism. These x time-slots will not be available to any advertiser arriving after the x_0'th day.

– If $x \leq s_{min}$, the adversary stops the input sequence. The social welfare of the prompt mechanism is: $ALG = x \cdot v_{min} \leq s_{min} \cdot v_{min}$. In the optimal solution, all time-slots in window $[1, s_{max}]$ are allocated to u_1 and the optimal social welfare is: $OPT = s_{max} \cdot v_{min}$. So: $OPT/ALG \geq s_{max}/s_{min}$.

– Else, $x > s_{min}$, the adversary sends another advertiser $u_2 : (x-1, v_{max}, x_0+1, x_0+x-1)$ and then stops the input sequence. No time-slot in u_2's time window $W_2 = [x_0+1, x_0+x-1]$ is available for u_2 and $ALG = x \cdot v_{min}$. One feasible solution is to allocate u_2 all the $x-1$ time-slots in window W_2 and allocate u_1 all the x_0 time-slots in windows $[1, x_0]$. Then we get: $OPT \geq (x-1) \cdot v_{max} + x_0 \cdot v_{min} \geq (x-1) \cdot v_{max} + v_{min}$. So

$$\frac{OPT}{ALG} \geq \frac{(x-1) \cdot v_{max} + v_{min}}{x \cdot v_{min}} = \frac{v_{max} - \frac{v_{max}-v_{min}}{x}}{v_{min}} \geq \frac{v_{max} + v_{min}}{2 \cdot v_{min}},$$

the last inequality is true as $x \geq 2$.

No matter what the value of x is, we get that $OPT/ALG \geq \min\{\frac{v_{max}+v_{min}}{2v_{min}}, \frac{s_{max}}{s_{min}}\}$.

4 A Prompt and Truthful Mechanism

In this section, we introduce a prompt and truthful mechanism for the TV ad placement problem. In section 5, we will continue to analyze the competitive ratio of the mechanism.

In the auction, advertisers arrive online and it is known that s_{min} is the lower bound of all s_i's. When advertiser i arrives, we are not clear about the advertisers arriving later. As shown in the analysis of the section 3, we cannot allocate i either too many or too few time-slots to achieve a low competitive ratio. In our prompt mechanism, no matter what s_i is, we allocate each advertiser 0 or $s = \lceil s_{min}/2 \rceil$ time-slots. We partition all the m time-slots into $M = \lceil m/s \rceil$ groups and call all the s time-slots in time window $[(j-1) \cdot s + 1, j \cdot s]$ as *group* G_j $(1 \leq j \leq M)$[3]. In our mechanism, each group can be allocated to only one advertiser and each advertiser can win only one group which is totally included in her time window[4] .

Our mechanism is implemented by the HALF-algorithm, as shown in Algorithm 1. In the HALF-algorithm, we maintain one candidate advertiser c_j for each group G_j. Whenever a new advertiser i arrives, look at the candidates for groups totally included in W_i and let c_j be the candidate with the lowest value (we say i *competes* on group G_j). If $v_{c_j} < v_i$, i will replace c_j as the candidate of G_j; otherwise, i is rejected irrevocably. On day $(k-1) \cdot s + 1$, the group G_k is allocated to its current candidate c_k and the payment for the group is calculated. The price that any winner pays for each time-slot in her winning group equals her critical value: the minimum value she can declare and still win one group.

In the HALF-algorithm, any advertiser i can only be allocated s time-slots or 0 slots, although she bids for as many as s_i time-slots. Before proving the truthfulness and promptness of HALF-algorithm, we will prove an important property shown in Lemma 2.

[3] When M is not a multiple of s, introduce some dummy slots which will not be used by any advertiser.

[4] Group G_j is totally included in time window $W_i = [a_i, d_i]$ if and only if $a_i \leq (j-1) \cdot s + 1$ and $d_i \geq j \cdot s$.

```
Data: Advertisers arriving online
Result: Allocation of time-slots
Set s := ⌈s_min/2⌉ and t := 1 ;          /* t means it is the t'th day now */
Initialize all candidates for groups as dummy advertisers with value of 0;
while t ≤ m do
    while there is a new advertiser u_i : (s_i, v_i, a_i, d_i) arriving on day t do
        Let S be the set of candidate advertisers for groups which are totally
        included in windows W_i;
        Let c_j be the candidate with the lowest value in S (if there are more
        than one such candidate, choose one arbitrary) ; /* We say i competes
        on group G_j. */
        if v_{c_j} < v_i then
            Make i be the new candidate for group G_j.
        end
    end
    if t == (k − 1) · s + 1 then
        Allocate group G_k to its current candidate advertiser c_k;
        Let p be the minimum value that advertiser c_k can declare and still win
        one group;
        The payment of advertiser c_k is s · p;
    end
    t := t + 1;
end
```

Algorithm 1. HALF-algorithm

Lemma 2. *Assume that one advertiser wins a group in the HALF-algorithm. If she has reported a higher bid and others' bids are unchanged, she can still win one group.*

Proof. Suppose that advertiser $i : (s_i, v_i, a_i, d_i)$ wins one group G_j in the HALF-algorithm. We will prove that if she reports $v'_i > v_i$ and others' bids are unchanged, she can still win one group to publish her advertisement. First, note that advertiser i will compete on the same group G_j, no matter what value she reports. Second, compare two runs of the HALF-algorithm in two cases: i reports v_i in case 1 and v'_i in case 2, and we can show that at any time the candidate for any group is the same in these two cases (this implies that i can win the same group in both cases). Before i arrives, these two cases are identical. Look at the next advertiser r arriving after i. For a contradiction, assume that the candidate for some group changes after r arrives in case 2. This can only happen when r competes on group G_j in case 1 and competes on another group G_h in case 2. Assume c_h is the candidate of G_h before r arrives. In case 1, both i and r compete on G_j and i wins. Thus, $v_i \geq v_r$. r competes on G_j instead of G_h so $v_{c_h} \geq v_i$. It follows that $v_{c_h} \geq v_i \geq v_r$. In case 2, r competes on G_h instead of G_i. But as $v_r \leq v_{c_h}$, r cannot become the candidate of G_h. The candidates of G_j and G_h are unchanged and so do the candidates of all the other groups. A contradiction occurs. Thus, the candidates of all groups are unchanged after advertiser r arrives. To finish the proof of monotonicity, we observe all the advertisers arriving after i one by one and use the same analysis.

Theorem 3. *The HALF-algorithm is truthful and prompt.*

Proof. We prove the truthfulness first. The true value of advertiser i is v_i. Let u_i, u'_i be the net profits that advertiser i gains when bidding v_i, v'_i respectively. We argue that $u_i \geq u'_i$ in each of the following cases, as a result bidding truthfully is a dominant strategy.

1. i wins one group when bidding either v_i or v'_i. In these two cases, i competes on one identical group G_j and then wins that group. The price p that i pays for each time-slot in G_j equals her critical value. So p is independent of i's bidding values and her total payment is $p_i = s \cdot p$. Thus, $u_i = s \cdot v_i - s \cdot p = u'_i$.
2. i wins one group when bidding v_i and no group when bidding v'_i. From lemma 2, we know that $v_i \geq v'_i$. When bidding v_i, i wins group G_j and the price paid for each time-slot is p. As p is the minimum value that i can bid to win group G_j, we get $p \leq v_i$. Hence, $u_i = s \cdot v_i - s \cdot p \geq 0 = u'_i$.
3. i wins one group when bidding v'_i and no group when bidding v_i. When bidding v'_i, i wins group G_j and the price paid for each time-slot is p. As p is the minimum value that i can bid to win group G_j and i wins no group when bidding v_i, we have $p \geq v_i$. Thus, $u'_i = s \cdot v_i - s \cdot p \leq 0 = u_i$.
4. i wins no group when bidding either v_i or v'_i. Thus, $u_i = u'_i = 0$.

Now we prove the promptness. Recall that regardless what value advertiser i reports, she will compete on one identical group G_j. Moreover, the winner of group G_j cannot be advertisers arriving after time-slot $(j-1) \cdot s + 1$. As the algorithm is monotone, the payment of i for group G_j is well defined and can be calculated at the very moment when G_j is allocated to i, which is time-slot $(j-1) \cdot s + 1$. Thus the algorithm is prompt.

5 Competitive Ratios

We have shown a lower bound of $\min\{\frac{v_{max}+v_{min}}{2v_{min}}, \frac{s_{max}}{s_{min}}\}$ on the competitive ratio for the ad placement problem in section 3. We analyze the competitive ratio of the HALF-algorithm in this section. For the case where all demands s_i's are uniform, the HALF-algorithm is proved to be 5-competitive. For the general case where s_i's are non-uniform, the algorithm is proved to be $5 \cdot \lceil s_{max}/s_{min} \rceil$-competitive. Note that when s_{max} is comparable with s_{min}, the algorithm is still very competitive.

5.1 Competitive Ratio When Demands Are Uniform

Assume that all demands s_i's have the same value of s_{min}. Then in HALF-algorithm, $s = \lceil s_{min}/2 \rceil$ and each advertiser i can only win one group (s time-slots) which is totally included in her time window W_i. However, in the optimal solution with the optimal social welfare OPT, i can win any time-slot in W_i. To compare the social welfare of HALF-algorithm, ALG, with OPT, we need to define an intermediate variable:

Definition 4. *OPT' is the optimal social welfare when the maximum number of time-slots that any advertiser i demands is $2s$ and her time window is $W_i' = [a_i', d_i'] = [\lfloor \frac{a_i-1}{s} \rfloor \cdot s + 1, \lceil \frac{d_i}{s} \rceil \cdot s]$.*

We call W_i' as i's *extended time window.* As in the optimal solution, any advertiser i can win at most $s_{min} \leq 2s$ time-slots in windows $W_i \subseteq W_i'$, we can get that $OPT \leq OPT'$. We will compare ALG with OPT'. Consider one solution which achieves the social welfare of OPT' now. In the solution, each advertiser bids for at most $2s$ time-slots in time window W'. Note that there are s time-slots in one group and any a_i' is the beginning of one group while any d_i' the end of one group. Without loss of generality, we can find one solution O' in which the social welfare is OPT', each group is allocated to one advertisers and each advertiser i wins 0, 1 or 2 groups in her extended window W_i'. In the following theorem, we will study this solution O' in detail.

Theorem 5. *The HALF-algorithm is 5-competitive when maximum demands are uniform.*

Proof. Let $A = (p_1, \ldots, p_M)$ be the solution of HALF-algorithm where advertiser p_j wins group G_j. Let $O' = (o_1, \ldots, o_M)$ be the solution which achieves the social welfare of OPT'. In O', advertiser o_j wins group G_j and some advertisers may appear twice in O' (e.g. $o_j = o_{j+1}$). We will match each o_j in O' to exactly one advertiser ℓ in A where $v_{o_j} \leq v_\ell$. Each advertiser in A is associated with at most 5 members of O'. In this way, $OPT \leq OPT' \leq 5 \cdot ALG$ and the competitive ratio of 5 is proved.

The matching is constructed as follows. Let $(o_{j_1}, \ldots, o_{j_{k_j}})$ be the members of O' that compete on time-slot j in HALF-algorithm (ordered by their arrival time). Note that o_{j_r} wins group G_{j_r} in O' and G_{j_r} should be in o_{j_r}'s extended time window $W'_{o_{j_r}}$. The number of groups in the extended time window $W'_{o_{j_r}}$ may be one or two more than that in the original time window $W_{o_{j_r}}$. Before showing the matching, we define function P mapping o_{j_r} to one member in A which wins one group in $W_{o_{j_r}}$:

1. $P(o_{j_r}) = p_{j_r}$ if group G_{j_r} is totally included in $W_{o_{j_r}}$;
2. $P(o_{j_r}) = p_{j_r+1}$ if group G_{j_r} is not totally included in $W_{o_{j_r}}$ and is the first group in $W'_{o_{j_r}}$;
3. $P(o_{j_r}) = p_{j_r-1}$ if group G_{j_r} is not totally included in $W_{o_{j_r}}$ and is the last group in $W'_{o_{j_r}}$;

In case (2) group G_{j_r+1} should be totally included in $W_{o_{j_r}}$ and in case (3) group G_{j_r-1} should also be totally included in $W_{o_{j_r}}$ as $s = \lceil s_{min}/2 \rceil$ and $d_{o_{j_r}} - a_{o_{j_r}} + 1 \geq s_{min}$. Now we show the rules of matching:

1. If $o_{j_{k_j-1}} = o_{j_{k_j}}$, match both $o_{j_{k_j-1}}$ and $o_{j_{k_j}}$ to p_j (denoted by $o_{j_{k_j-1}}, o_{j_{k_j}} \to p_j$); otherwise, $o_{j_{k_j}} \to p_j$.
2. If $r < k_j$ and $o_{j_r} \neq o_{j_{r+1}}$ and $o_{j_r} \neq o_{j_{r-1}}$, then $o_{j_r} \to P(o_{j_{r+1}})$.
3. If $r < k_j - 1$ and $o_{j_r} = o_{j_{r+1}}$, then:
   (a) If $o_{j_{r+2}} = o_{j_{r+3}}$, then $o_{j_r} \to P(o_{j_{r+2}})$ and $o_{j_{r+1}} \to P(o_{j_{r+3}})$;

(b) Else if there exists t s.t. $t > r$ and $o_{j_t} = o_{j_{t+1}}$, then choose the minimum t, $o_{j_r} \rightarrow P(o_{j_{r+2}})$ and $o_{j_{r+1}} \rightarrow P(o_{j_{t+1}})$;

(c) Else, $o_{j_r} \rightarrow P(o_{j_{r+2}})$ and $o_{j_{r+1}} \rightarrow p_j$;

Rule (1) is used to deal with the last advertiser $o_{j_{k_j}}$ competing on time-slot j. Rule (2) is for advertisers appearing once in O' and rule (3) is for those appearing twice in O'. As one advertiser can appear at most twice in O', our matching covers all cases and each element in O' can be matched to exactly one advertiser in A. Another important fact we will use later is that there are no two elements, $o_{j'_1}$, $o_{j'_2}$ s.t. $o_{j'_1} \rightarrow P(o_{j'_3})$ and $o_{j'_2} \rightarrow P(o_{j'_3})$ in our matching.

Firstly, we prove that any advertiser p_j is associated with at most 5 elements in O'. There are two possible cases: (**a**) $o_{j_{k_j-1}} = o_{j_{k_j}}$, which implies that rule (3c) is not applicable. Rule (1) matches two elements, $o_{j_{k_j}-1}$ and $o_{j_{k_j}}$, to p_j. p_j can also appear in matching like $o_{j'_r} \rightarrow P(o_{j'_{r'}})$, where $P(o_{j'_{r'}}) = p_j$. Note that function P can map at most three elements in O' to p_j (they are o_{j-1}, o_j, o_{j+1}) , and it does not happen that there are two elements, $o_{j'_1}$, $o_{j'_2}$ s.t. $o_{j'_1} \rightarrow P(o_{j'_3})$ and $o_{j'_2} \rightarrow P(o_{j'_3})$. So rule (2) and (3) can match at most three elements in O' to p_j. (**b**) $o_{j_{k_j-1}} \neq o_{j_{k_j}}$. Rule (1) and (3c) matches two elements, $o_{j_{k_j}}$ and $o_{j_{r+1}}$, to p_j ($o_{j_{r+1}}$ may not exist). Similar to the former case, p_j can also appear at most three times in matching like $o_{j'_r} \rightarrow P(o_{j'_{r'}})$, where $P(o_{j'_{r'}}) = p_j$.

It remains to be proved that any element in O' is always matched to an advertiser with higher or equal value. In rule (1), since both $o_{j_{k_j}}$ and p_j compete on G_j and p_j wins, $v_{o_{j_{k_j}}} \leq v_{p_j}$. In rule (2), when $o_{j_{r+1}}$ arrives, she competes on G_j rather than the group advertiser $P(o_{j_{r+1}})$ wins. At this moment, o_{j_r} has already arrived, thus the current candidate h for the group advertiser $P(o_{j_{r+1}})$ wins has value at least $v_{o_{j_r}}$, i.e., $v_{o_{j_r}} \leq v_h$. As $v_{P(o_{j_{r+1}})}$ should be no less than v_h, $v_{o_{j_r}} \leq v_h \leq v_{P(o_{j_{r+1}})}$. In rule (3a), when $o_{j_{r+2}}$ arrives, she competes on G_j rather than the group advertiser $P(o_{j_{r+2}})$ or $P(o_{j_{r+3}})$ wins. Similarly, $v_{o_{j_r}} \leq v_{P(o_{j_{r+2}})}$ and $v_{o_{j_{r+1}}} \leq v_{P(o_{j_{r+3}})}$. In rule (3b) and (3c), we can get similar results.

5.2 Competitive Ratio When Demands Are Non-uniform

Now we consider the general case where s_{max} is not necessarily equal to s_{min}. In this case, let $ALG2$ be the social welfare achieved by the HALF-algorithm. Let $O2$ be the optimal solution and $OPT2$ be the optimal social welfare. In $O2$, advertisers may win more than s_{min} time-slots. We will use $O2$ to construct one new solution $O2'$ in which any advertiser wins no more than s_{min} time-slots. The social welfare of $O2'$ is $OPT2'$ and it can be proved that $OPT2 \leq OPT2' \cdot \lceil s_{max}/s_{min} \rceil$. Then we will compare $OPT2'$ with $ALG2$ and a competitive ratio of $5 \cdot \lceil s_{max}/s_{min} \rceil$ is proved.

Theorem 6. *The HALF-algorithm is* $5 \cdot \lceil s_{max}/s_{min} \rceil$*-competitive when maximum demands are non-uniform.*

Proof. Let $ALG2$ be the social welfare achieved by the HALF-algorithm. Let $O2$ be the optimal solution and $OPT2$ be the optimal social welfare. We use $O2$ to construct a new solution $O2'$. For any advertiser i who wins x_i time-slots in $O2$, we choose the first $x_i' = \lceil x_i / \lceil s_{max}/s_{min} \rceil \rceil$ time-slots from all these x_i time-slots and allocate them to i in $O2'$. The social welfare of $O2'$ is $OPT2'$. As $x_i \le x_i' \cdot \lceil s_{max}/s_{min} \rceil$ for any i, we get:

$$OPT2 \le OPT2' \cdot \lceil s_{max}/s_{min} \rceil.$$

In the auction, each advertiser i demands at most s_i time-slots. Now consider another scenario where each advertiser i's maximum demand is s_{min} instead of s_i and her other information remains the same as before. In this scenario, the social welfare achieved by HALF-algorithm is $ALG3$. The optimal solution is $O3$ and the optimal social welfare is $OPT3$. As all maximal demands are uniform, by Theorem 5, we can get:

$$OPT3 \le 5 \cdot ALG3.$$

Recall that advertiser i wins x_i' slots in $O2'$. As $x_i' \le \lceil s_{max} / \lceil s_{max}/s_{min} \rceil \rceil \le s_{min}$, advertiser i wins no more than s_{min} slots in $O2'$. In solution $O3$, any advertiser i can also win no more than s_{min} slots. As $O3$ is the optimal solution, $OPT3$ should be the maximum social welfare and $OPT2' \le OPT3$. On the other hand, note that the only difference between the two scenarios we have considered is advertisers' maximum demands s_i. However, no matter what the value of s_i is, the HALF-algorithm will only allocate each advertiser 0 or s time-slots. In these two scenarios, the HALF-algorithm has the same output and then $ALG2 = ALG3$. Thus,

$$\begin{aligned} OPT2 \le \lceil s_{max}/s_{min} \rceil \cdot OPT2' &\le \lceil s_{max}/s_{min} \rceil \cdot OPT3 \\ \le 5 \cdot \lceil s_{max}/s_{min} \rceil \cdot ALG3 &= 5 \cdot \lceil s_{max}/s_{min} \rceil \cdot ALG2. \end{aligned}$$

References

1. Aggarwal, G., Goel, G., Karande, C., Mehta, A.: Online vertex-weighted bipartite matching and single-bid budgeted allocations. In: Proceedings of the Twenty-Second Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 1253–1264. SIAM (2011)
2. Aggarwal, G., Hartline, J.D.: Knapsack auctions. In: Proceedings of the Seventeenth Annual ACM-SIAM Symposium on Discrete Algorithm, pp. 1083–1092. ACM (2006)
3. Archer, A., Tardos, É.: Truthful mechanisms for one-parameter agents. In: Proceedings of the 42nd IEEE Symposium on Foundations of Computer Science, pp. 482–491. IEEE (2001)
4. Azar, Y., Khaitsin, E.: Prompt mechanism for ad placement over time. In: Persiano, G. (ed.) SAGT 2011. LNCS, vol. 6982, pp. 19–30. Springer, Heidelberg (2011)
5. Bartal, Y., Chin, F.Y.L., Chrobak, M., Fung, S.P.Y., Jawor, W., Lavi, R., Sgall, J., Tichý, T.: Online competitive algorithms for maximizing weighted throughput of unit jobs. In: Diekert, V., Habib, M. (eds.) STACS 2004. LNCS, vol. 2996, pp. 187–198. Springer, Heidelberg (2004)

6. Borgs, C., Chayes, J., Etesami, O., Immorlica, N., Jain, K., Mahdian, M.: Dynamics of bid optimization in online advertisement auctions. In: Proceedings of the 16th International Conference on World Wide Web, pp. 531–540. ACM (2007)
7. Chan, W.-T., Lam, T.-W., Ting, H.-F., Wong, P.W.H.: New results on on-demand broadcasting with deadline via job scheduling with cancellation. In: Chwa, K.-Y., Munro, J.I. (eds.) COCOON 2004. LNCS, vol. 3106, pp. 210–218. Springer, Heidelberg (2004)
8. Chin, F.Y.L., Fung, S.P.Y.: Online scheduling with partial job values: Does timesharing or randomization help? Algorithmica 37(3), 149–164 (2003)
9. Chrobak, M., Jawor, W., Sgall, J., Tichý, T.: Improved online algorithms for buffer management in qoS switches. In: Albers, S., Radzik, T. (eds.) ESA 2004. LNCS, vol. 3221, pp. 204–215. Springer, Heidelberg (2004)
10. Cole, R., Dobzinski, S., Fleischer, L.K.: Prompt mechanisms for online auctions. In: Monien, B., Schroeder, U.-P. (eds.) SAGT 2008. LNCS, vol. 4997, pp. 170–181. Springer, Heidelberg (2008)
11. Englert, M., Westermann, M.: Considering suppressed packets improves buffer management in qos switches. In: Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 209–218. Society for Industrial and Applied Mathematics (2007)
12. Nisan, N., Bayer, J., Chandra, D., Franji, T., Gardner, R., Matias, Y., Rhodes, N., Seltzer, M., Tom, D., Varian, H., Zigmond, D.: Google's auction for tv ads. In: Albers, S., Marchetti-Spaccamela, A., Matias, Y., Nikoletseas, S., Thomas, W. (eds.) ICALP 2009, Part II. LNCS, vol. 5556, pp. 309–327. Springer, Heidelberg (2009)
13. Ting, H.-F.: A near optimal scheduler for on-demand data broadcasts. In: Calamoneri, T., Finocchi, I., Italiano, G.F. (eds.) CIAC 2006. LNCS, vol. 3998, pp. 163–174. Springer, Heidelberg (2006)
14. Vickrey, W.: Counterspeculation, auctions, and competitive sealed tenders. The Journal of Finance 16(1), 8–37 (1961)
15. Zhou, Y., Chakrabarty, D., Lukose, R.: Budget constrained bidding in keyword auctions and online knapsack problems. In: Papadimitriou, C., Zhang, S. (eds.) WINE 2008. LNCS, vol. 5385, pp. 566–576. Springer, Heidelberg (2008)