Bicriteria Online Matching: Maximizing Weight and Cardinality

Nitish Korula[1], Vahab S. Mirrokni[1], and Morteza Zadimoghaddam[2]

[1] Google Research, New York NY 10011, USA, {nitish, mirrokni}@google.com
[2] Massachusetts Institute of Technology, Cambridge MA 02139, USA, morteza@mit.edu

Abstract. Inspired by online ad allocation problems, many results have been developed for online matching problems. Most of the previous work deals with a single objective, but, in practice, there is a need to optimize multiple objectives. Here, as an illustrative example motivated by display ads allocation, we study a bi-objective online matching problem.
In particular, we consider a set of fixed nodes (ads) with capacity constraints, and a set of online items (pageviews) arriving one by one. Upon arrival of an online item i, a set of eligible fixed neighbors (ads) for the item is revealed, together with a weight w_{ia} for eligible neighbor a. The problem is to assign each item to an eligible neighbor online, while respecting the capacity constraints; the goal is to maximize both the total weight of the matching and the cardinality. In this paper, we present both approximation algorithms and hardness results for this problem.
An (α, β)-approximation for this problem is a matching with weight at least α fraction of the maximum weighted matching, and cardinality at least β fraction of maximum cardinality matching. We present a parametrized approximation algorithm that allows a smooth tradeoff curve between the two objectives: when the capacities of fixed nodes are large, we give a $p(1 - 1/e^{1/p}), (1 - p)(1 - 1/e^{1/1-p})$-approximation for any $0 \leq p \leq 1$, and prove a 'hardness curve' combining several inapproximability results. These upper and lower bounds are always close (with a maximum gap of 9%), and exactly coincide at the point $(0.43, 0.43)$. For small capacities, we present a smooth parametrized approximation curve for the problem between $(0, 1 - 1/e)$ and $(1/2, 0)$ passing through a $(1/3, 0.3698)$-approximation.

1 Introduction

In the past decade, there has been much progress in designing better algorithms for online matching problems. This line of research has been inspired by interesting combinatorial techniques that are applicable in this setting, and by online ad allocation problems. For example, the display advertising problem has been modeled as maximizing the weight of an online matching instance [11,10,8,2,19]. While weight is indeed important, this model ignores the fact that cardinality of the matching is also crucial in the display ad application. This example illustrates the fact that in many real applications of online allocation, one needs to optimize multiple objective functions, though most of the previous work in this area deals with only a single objective function. On the other hand, there is a large body of work exploring *offline* multi-objective optimization in the approximation algorithms literature. In this paper, we focus on simultaneously maximizing *online* two objectives which have been studied extensively in matching problems: cardinality and weight. Besides being a natural mathematical problem, this is motivated by online display advertising applications.

Applications in Display Advertising. In online display advertising, advertisers typically purchase bundles of millions of display ad impressions from web publishers. Display ad serving systems that assign ads to pages on behalf of publishers must satisfy the contracts with advertisers, respecting targeting criteria and delivery goals. Modulo this, publishers try to allocate ads intelligently to maximize overall quality (measured, for example, by clicks), and therefore a desirable property of an ad serving system is to maximize this quality while satisfying the contracts to deliver the purchased number $n(a)$ impressions to advertiser a. This has been modeled in the literature (e.g., [11,1,24,8,2,19]) as an online allocation problem, where quality is represented by edge weights, and contracts are enforced by overall delivery goals: While trying to maximize the *weight* of the allocation, the ad serving systems should deliver $n(a)$ impressions to advertiser a. However, online algorithms with adversarial input cannot guarantee the delivery of $n(a)$ impressions, and hence the goals $n(a)$ were previously modeled as upper bounds. But maximizing the *cardinality* subject to these upper bounds is identical to delivering as close to the targets as possible. This motivates our model of the display ad problem as simultaneously maximizing weight and cardinality.

Problem Formulation. More specifically, we study the following bicriteria online matching problem: consider a set of bins (also referred to as fixed nodes, or ads) A with capacity constraints $n(a) > 0$, and a set of online items (referred to as online nodes, or impressions or pageviews) I arriving one by one. Upon arrival of an online item i, a set S_i of eligible bins (fixed node neighbors) for the item is revealed, together with a weight w_{ia} for eligible bin $a \in S_i$. The problem is to assign each item i to an eligible bin in S_i or discard it online, while respecting the capacity constraints, so bin a gets at most $n(a)$ online items. The goal is to maximize both the cardinality of the allocation (i.e. the total number of assigned items) and the sum of the weights of the allocated online items.

It was shown in [11] that achieving any positive approximation guarantee for the total weight of the allocation requires the *free disposal* assumption, i.e. that there is no penalty for assigning more online nodes to a bin than its capacity, though these extra nodes do not count towards the objective. In the advertising application, this means that in the presence of a contract for $n(a)$ impressions, advertisers are only pleased by – or at least indifferent to – getting *more* than $n(a)$ impressions. More specifically, if a set I^a of online items are assigned to each bin a, and $I^a(k)$ denotes the set of k online nodes with maximum weight in I^a, the goal is to simultaneously maximize cardinality which is $\sum_{a \in A} \min(|I^a|, n(a))$, and total weight which is $\sum_{a \in A} \sum_{i \in I^a(n(a))} w_{ia}$.

Throughout this paper, we use W_{opt} to denote the maximum weight matching, and overload this notation to also refer to the weight of this matching. Similarly, we use C_{opt} to denote both the maximum cardinality matching and its cardinality. Note that C_{opt} and W_{opt} may be distinct matchings. We aim to find (α, β)-approximations for the bicriteria online matching problem: These are matchings with weight at least αW_{opt} and cardinality at least βC_{opt}. Our approach is to study parametrized approximation algorithms that allow a smooth tradeoff curve between the two objectives, and prove both approximation and hardness results in this framework. As an offline problem, the above bicriteria problem can be solved optimally in polynomial time, i.e., one can check if there exists an assignment of cardinality c and weight w respecting capacity constraints. (One can verify this by observing that the integer linear programming formulation for the offline problem is totally unimodular, and therefore the problem can be solved by solving the corresponding LP relaxation.) However in the online competitive setting, even maximizing one of these two objectives does not admit better than a $1 - 1/e$ approximation [18]. A naive greedy algorithm gives a $\frac{1}{2}$-approximation for maximizing a single objective, either for cardinality or for total weight under the free disposal assumption.

Results and Techniques. The seminal result of Karp, Vazirani and Vazirani [18] gives a simple randomized $(1-1/e)$-competitive algorithm for maximizing cardinality. For the weight objective, no algorithm better than the greedy $1/2$-approximation is known, but for the case of large capacities, a $1 - 1/e$-approximation has been developed [11] following the primal-dual analysis framework of Buchbinder *et al.* [5,21]. Using these results, one can easily get a $(\frac{p}{2}, (1-p)(1-\frac{1}{e}))$-approximation for the bicriteria online matching problem with small capacities, and a $\left(p(1-\frac{1}{e}), (1-p)(1-\frac{1}{e})\right)$-approximation for large capacities. These factors are achieved by applying the online algorithm for weight, WeightAlg, and the online algorithm for cardinality, CardinalityAlg, as subroutines as follows: When an online item arrives, pass it to WeightAlg with probability p, and CardinalityAlg with probability $1-p$. As for a hardness result, it is easy to show that an approximation factor better than $(\alpha, 1-\alpha)$ is not achievable for any $\alpha > 0$. There is a large gap between the above approximation factors and hardness results. For example, the naive algorithm gives a $(0.4, 0.23)$-approximation, but the hardness result does not preclude a $(0.4, 0.6)$-approximation. In this paper, we tighten the gap between these lower and upper bounds, and present new tradeoff curves for both algorithms and hardness results. Our lower and upper bound results are summarized in Figure 1. For the case of large capacities, these upper and lower bound curves are always close (with a maximum vertical gap of 9%), and exactly coincide at the point $(0.43, 0.43)$.

We first describe our hardness results. In fact, we prove three separate inapproximability results which can be combined to yield a 'hardness curve' for the problem. The first result gives better upper bounds for large values of β; this is based on structural properties of matchings, proving some invariants for any online algorithm on a family of instances, and writing a factor-revealing mathematical program (see Section 2.1). The second main result is an improved upper bound for large values of α, and is based on a new family of instances for which achieving a large value for α implies very small values of β (see Section 2.2). Finally, we show that for any achievable (α, β), we have $\alpha + \beta \leq 1 - \frac{1}{e^2}$ (see Theorem 3).

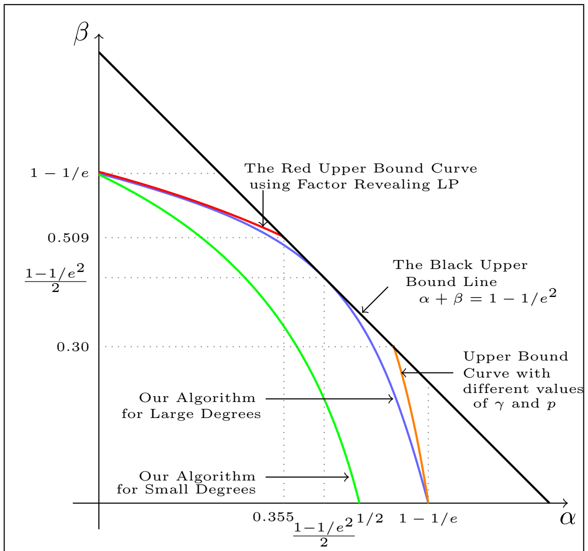

Figure 1: New curves for upper and lower bounds.

These hardness results show the limit of what can be achieved in this model. We next turn to algorithms, to see how close we can come to these limits. The key to our new algorithmic results lies in the fact that though each subroutine WeightAlg and CardinalityAlg only receives a fraction of the online items, it can use the entire set of bins. This may result in both subroutines filling up a bin, but if WeightAlg places t items in a bin, we can discard t of the items placed there by CardinalityAlg and still get at least the cardinality obtained by CardinalityAlg and the weight obtained by WeightAlg. Each subroutine therefore has access to the entire bin capacity, which is more than it 'needs' for those items passed to it. Thus, its competitive ratio can be made better than $1 - 1/e$. For large capacities, we prove the following theorem by extending the primal-dual analysis of Buchbinder *et al.* and Feldman *et al.* [11,21,5].

Theorem 1. *For all* $0 < p < 1$*, there is an algorithm for the bicriteria online matching problem with competitive ratios tending to* $\left(p(1 - \frac{1}{e^{1/p}}), (1-p)(1 - \frac{1}{e^{1/(1-p)}})\right)$ *as* $\min_a\{n(a)\}$ *tends to infinity.*

For small capacities, our result is more technical and is based on studying structural properties of matchings, proving invariants for our online algorithm over any instance, and solving a factor-revealing LP that combines these new invariants and previously known combinatorial techniques by Karp, Vazirani, Vazirani, and Birnbaum and Mathieu [18,4]. Factor revealing LPs have been used in the context of online allocation problems [21,20]. In our setting, we need to prove new variants and introduce new inequalities to take into account and analyze the tradeoff between the two objective functions. This result can also be parametrized by p, the fraction of items sent to WeightAlg, but we do not have a closed form expression. Hence, we state the result for $p = 1/2$.

Theorem 2. *For all $0 \leq p \leq 1$, the approximation guarantee of our algorithm for the bicriteria online matching problem is lower bounded by the green curve of Figure 1. In particular, for $p = 1/2$, we have the point* $(1/3, 0.3698)$.

Related Work. Our work is related to online ad allocation problems, including the *Display Ads Allocation (DA)* problem [11,10,1,24], and the *AdWords (AW)* problem [21,7]. In both of these problems, the publisher must assign online impressions to an inventory of ads, optimizing efficiency or revenue of the allocation while respecting pre-specified contracts. The Display Ad (DA) problem is the online matching problem described above only considering the weight objective [11,2,19]. In the AdWords (AW) problem, the publisher allocates impressions resulting from search queries. Advertiser a has a budget $B(a)$ on the total spend instead of a bound $n(a)$ on the number of impressions. Assigning impression i to advertiser a consumes w_{ia} units of a's budget instead of 1 of the $n(a)$ slots, as in the DA problem. For both of these problems, $1 - \frac{1}{e}$-approximation algorithms have been designed under the assumption of large capacities [21,5,11]. None of the above papers for the adversarial model study multiple objectives at the same time.

Besides the adversarial model studied in this paper, online ad allocations have been studied extensively in various *stochastic models*. In particular, the problem has been studied in the *random order model*, where impressions arrive in a random order [7,10,1,24,17,20,23]; and the *iid* model in which impressions arrive iid according to a known (or unknown) distribution [12,22,16,8,9]. In such stochastic settings, primal and dual techniques have been applied to getting improved approximation algorithms. These techniques are based on computing offline optimal primal or dual solutions of an expected instance, and using this solution online [12,7]. It is not hard to generalize these techniques to the bicritera online matching problem. In this extended abstract, we focus on the adversarial model, and leave discussions of extensions of such techniques for the stochastic bicriteria problem to the full version of the paper. Note that in order to deal with traffic spikes, adversarial competitive analysis is important from a practical perspective, as discussed in [23].

Most previous work on online problems with multiple objectives has been in the domain of routing and scheduling, and with different models. Typically, goals are to maximize throughput and fairness; see the work of Goel *et al.* [15,14], Buchbinder and Naor [6], and Wang *et al.* [25]. In this literature, different objectives often come from applying different functions on the same set of inputs, such as processing times or bandwidth allocations. In a model more similar to ours, Bilò *et al.* [3] consider scheduling where each job has two different and unrelated requirements, processing time and memory; the goal is to minimize makespan while also minimizing maximum memory requirements on each machine. In another problem with distinct metrics, Flammini and Nicosia [13] consider the k-server problem with a distance metric and time metric defined on the set of service locations. However, unlike our algorithms, theirs do not compete simultaneously against the best solution for each objective; instead, they compete against offline solutions that must simultaneously do well on both objectives.

Further, the competitive ratio depends on the relative values of the two objectives. Such results are of limited use in advertising applications, for instance, where click-through rates per impression may vary by several orders of magnitude.

2 Hardness Instances

In this section for any $0 \leq \alpha \leq 1-1/e$, we prove upper bounds on β such that the bicriteria online matching problem admits an (α, β)-approximation. Note that it is not possible to achieve α-approximation guarantee for the total weight of the allocation for any $\alpha > 1-1/e$. We have two types of techniques to achieve upper bounds: a) Factor-Revealing Linear Programs, b) Super Exponential Weights Instances, which are discussed in Subsections 2.1, and 2.2 respectively. Factor revealing LP hardness instances give us the red upper bound curve in Figure 1. The orange upper bound curve in Figure 1 is proved by Super Exponential Weights Instances presented in Subsection 2.2, and the black upper bound line in Figure 1 is proved in Theorem 3.

2.1 Better Upper Bounds via Factor-Revealing Linear Programs

We construct an instance, and a linear program $LP_{\alpha,\beta}$ based on the instance where α and β are two parameters in the linear program. We prove that if there exists an (α, β)-approximation for the bicriteria online matching problem, we can find a feasible solution for $LP_{\alpha,\beta}$ based on the algorithm's allocation to the generated instance. Finally we find out for which pairs (α, β) the linear program $LP_{\alpha,\beta}$ is infeasible. These pairs (α, β) are upper bounds for the bicriteria online matching problem.

For any two integers C, l, and some large weight $W \gg 4l^2$, we construct the instance as follows. We have l phases, and each phase consists of l sets of C identical items, i.e. l^2C items in total. For any $1 \leq t, i \leq l$, we define $O_{t,i}$ to be the set i in phase t that has C identical items. In each phase, we observe the sets of items in increasing order of i. There are two types of bins: a) l weight bins $b_1, b_2, \cdots, b_l$ which are shared between different phases, b) l^2 cardinality bins $\{b'_{t,i}\}_{1\leq t,i\leq l}$. For each phase $1 \leq t \leq l$, we have l separate bins $\{b'_{t,i}\}_{1\leq i\leq l}$. The capacity of all bins is C. We pick two permutations $\pi_t, \sigma_t \in \mathbb{S}_n$ uniformly at random at the beginning of each phase t to construct edges. We note that these permutations are private knowledge, and they are not revealed to the algorithm. For any $1 \leq i \leq j \leq l$, we put an edge between every item in set $O_{t,i}$ and bin $b'_{t,\sigma_t(j)}$ with weight 1 where $\sigma_t(j)$ is the jth number in permutation σ_t. We also put an edge between every item in set $O_{t,i}$ and bin $b_{\pi_t(j)}$ (for each $j \geq i$) with weight W^t.

Suppose there exists an (α, β)-approximation algorithm $A_{\alpha,\beta}$ for the bicriteria online matching problem. For any $1 \leq t, i \leq l$, let $x_{t,i}$ be the expected number of items in set $O_{t,i}$ that algorithm $A_{\alpha,\beta}$ assigns to weight bins $\{b_{\pi_t(j)}\}_{j=i}^{l}$. Similarly we define $y_{t,i}$ to be the expected number of items in set $O_{t,i}$ that algorithm

$A_{\alpha,\beta}$ assigns to cardinality bins $\{b'_{t,\sigma_t(j)}\}_{j=i}^{l}$. We know that when set $O_{t,i}$ arrives, although the algorithm can distinguish between weight and cardinality bins, it sees no difference between the weight bins $\{b_{\pi_t(j)}\}_{j=i}^{l}$, and no difference between the cardinality bins $\{b'_{t,\sigma_t(j)}\}_{j=i}^{l}$. By uniform selection of π and σ, we ensure that in expectation the $x_{t,i}$ items are allocated equally to weight bins $\{b_{\pi_t(j)}\}_{j=i}^{l}$, and the $y_{t,i}$ items are allocated equally to cardinality bins $\{b'_{t,\sigma_t(j)}\}_{j=i}^{l}$. In other words, for $1 \leq i \leq j \leq l$, in expectation $x_{t,i}/(l-i+1)$ and $y_{t,i}/(l-i+1)$ items of set $O_{t,i}$ is allocated to bins $b_{\pi_t(j)}$ and $b'_{t,\sigma_t(j)}$, respectively. It is worth noting that similar ideas have been used in previous papers on online matching [18,4].

Since weights of all edges to cardinality bins are 1, we can assume that the items assigned to cardinality bins are kept until the end of the algorithm, and they will not be thrown away. We can similarly say that the weights of all items for weight bins is the same in a single phase, so we can assume that an item that has been assigned to some weight bin in a phase will not be thrown away at least until the end of the phase. However, the algorithm might use the free disposal assumption for weight bins in different phases. We have the following capacity constraints on bins $b_{\pi_t(j)}$ and $b'_{t,\sigma_t(j)}$:

$$\forall 1 \leq t, j \leq l\colon \sum_{i=1}^{j} x_{t,i}/(l-i+1) \leq C \ \&\ \sum_{i=1}^{j} y_{t,i}/(l-i+1) \leq C. \qquad (1)$$

At any stage of phase t, the total weight assigned by the algorithm cannot be less than α times the optimal weight allocation up to that stage, or we would not have weight αW_{opt} if the input stopped at this point. After set $O_{t,i}$ arrives, the maximum weight allocation achieves at least total weight CiW^t which is achieved by assigning items in set $O_{t,i'}$ to weight bin $b_{\pi_t(i')}$ for each $1 \leq i' \leq i$. On the other hand, the expected weight in allocation of algorithm $A_{\alpha,\beta}$ is at most $C(tl + W^{t-1}l) + W^t \sum_{i'=1}^{i} x_{t,i'} \leq W^t(C/\sqrt{W} + \sum_{i'=1}^{i} x_{t,i'})$. Therefore we have the following inequality for any $1 \leq t, i \leq l$:

$$\sum_{i'=1}^{i} x_{t,i'}/C \geq \alpha i - 1/\sqrt{W}. \qquad (2)$$

We show in Lemma 1 that the linear program $LP_{\alpha,\beta}$ is feasible if there exists an algorithm $A_{\alpha,\beta}$ by defining $p_i = \sum_{t=1}^{l} x_{t,i}/lC$, and $q_i = \sum_{t=1}^{l} y_{t,i}/lC$. Now for any α, we can find the maximum β for which the $LP_{\alpha,\beta}$ has some feasible solution for large values of l and W. These factor-revealing linear programs yield the red upper bound curve in Figure 1.

$LP_{\alpha,\beta}$

C1:	$\sum_{i'=1}^{i} p_{i'}$	$\geq \alpha i - 1/\sqrt{W}$	$\forall 1 \leq i \leq l$
C2:	$\sum_{i=1}^{l} q_i$	$\geq l\beta - 1$	
C3:	$p_i + q_i$	≤ 1	$\forall 1 \leq i \leq l$
C4:	$\sum_{i=1}^{j} p_i/(l-i+1)$	≤ 1	$\forall 1 \leq j \leq l$
C5:	$\sum_{i=1}^{j} q_i/(l-i+1)$	≤ 1	$\forall 1 \leq j \leq l$

Lemma 1. *If there exists an (α, β)-approximation algorithm for the bicriteria online matching problem, there exists a feasible solution for $LP_{\alpha,\beta}$ as well.*

In addition to computational bounds for infeasibility of certain (α, β) pairs, we can theoretically prove in Theorem 3 that for any (α, β) with $\alpha+\beta > 1-1/e^2$, the $LP_{\alpha,\beta}$ is infeasible so there exists no (α, β) approximation for the problem. We note that Theorem 3 is a simple generalization of the $1-1/e$ hardness result for the classic online matching problem [18,4].

Theorem 3. *For any small $\epsilon > 0$, and $\alpha + \beta \geq 1 - 1/e^2 + \epsilon$, there exists no (α, β)-approximation algorithm for the bicriteria matching problem.*

Proof. We just need to show that $LP_{\alpha,\beta}$ is infeasible. Given a solution of $LP_{\alpha,\beta}$, we find a feasible solution for LP'_ϵ defined below by setting $r_i = p_i + q_i$ for any $1 \leq i \leq l$.

$$
\begin{array}{llll}
LP'_\epsilon & \sum_{i=1}^{l} r_i & \geq (1 - 1/e^2 + \epsilon/2)l & \\
& r_i & \leq 1 & \forall 1 \leq i \leq l \\
& \sum_{i=1}^{j} r_i/(l - i + 1) & \leq 2 & \forall 1 \leq j \leq l
\end{array}
$$

The first inequality in LP'_ϵ is implied by summing up the constraint C1 for $i = l$, and constraint C2 in $LP_{\alpha,\beta}$, and also using the fact that $\alpha+\beta \geq (1-1/e^2+\epsilon/2)+\epsilon/2$. We note that the $\epsilon/2$ difference between the $\alpha+\beta$ and $1-1/e^2+\epsilon/2$ takes care of $-1/\sqrt{W}$ and -1 in the right hand sides of constraints C1 and C2 for large enough values of l and W. Now we prove that LP'_ϵ is infeasible for any $\epsilon > 0$ and large enough l. Suppose there exists a feasible solution $r_1, r_2, \cdots, r_n$. For any pair $1 \leq i < j \leq n$, if we have $r_i < 1$ and $r_j > 0$, we update the values of r_i and r_j to $r_i^{new} = r_i+\min\{1-r_i, r_j\}$, and $r_j^{new} = r_j-\min\{1-r_i, r_j\}$. Since we are moving the same amount from r_j to r_i (for some $i < j$), all constraints still hold. If we do this operation iteratively until there is no pair r_i and r_j with the above properties, we reach a solution $\{r'_i\}_{i=1}^{l}$ of this form: $1, 1, \cdots, 1, x, 0, 0, \cdots, 0$ for some $0 \leq x \leq 1$. Let t be the maximum index for which r'_t is 1. Using the third inequality for $j = l$, we have that $\sum_{i=1}^{t} 1/(l - i + 1) \leq 2$ which means that $\ln(l/(l-t+1)) \leq 2$. So t is not greater than $l(1 - 1/e^2)$, and consequently $\sum_{i=1}^{l} r'_i \leq t+1 \leq (1-1/e^2)l+1 < (1-1/e^2+\epsilon/2)l$. This contradiction proves that LP'_ϵ is infeasible which completes the proof of theorem.

2.2 Hardness Results for Large Values of Weight Approximation Factor

The factor-revealing linear program $LP_{\alpha,\beta}$ gives almost tight bounds for small values of α. In particular, the gap between the the upper and lower bounds for the cardinality approximation ratio β is less than 0.025 for $\alpha \leq (1-1/e^2)/2$. But for large values of α ($\alpha > (1-1/e^2)/2$), this approach does not give anything better than the $\alpha+\beta \leq 1-1/e^2$ bound proved in Theorem 3 . This leaves a maximum gap of $1/e - 1/e^2 \approx 0.23$ between the upper and lower bounds at $\alpha = 1 - 1/e$. In order to close the gap at $\alpha = 1 - 1/e$, we present a different analysis based on a new set of instances, and reduce the maximum gap between lower and upper bounds from 0.23 to less than 0.09 for all values of $\alpha \geq (1 - 1/e^2)/2$.

The main idea is to construct a hardness instance I_γ for any $1/e \leq \gamma < 1$, and prove that for any $0 \leq p \leq 1 - \gamma$, the pair $(1 - 1/e - f(p), p/(1 - \gamma))$ is an upper bound on (α, β) where $f(p)$ is $\frac{p}{e(\gamma+p)}$. In other words, there exists no (α, β)-approximation algorithm for this problem with both $\alpha > 1 - 1/e - f(p)$ and $\beta > p/(1-\gamma)$. By enumerating different pairs of γ and p, we find the orange upper bound curve in Figure 1.

For any $\gamma \geq 1/e$, we construct instance I_γ as follows: The instance is identical to the hardness instance in Subsection 2.1, but we change some of the edge weights. To keep the description short, we only describe the edges with modified weights here. Let r be $\lfloor 0.5 \log_{1/\gamma} l \rfloor$. In each phase $1 \leq t \leq l$, we partition the l sets of items $\{O_{t,i}\}_{i=1}^{l}$ into r groups. The first $l(1 - \gamma)$ sets are in the first group. From the remaining γl sets, we put the first $(1 - \gamma)$ fraction in the second group and so on. Formally, we put set $O_{t,i}$ in group $1 \leq z < r$ for any $i \in [l - l\gamma^{z-1} + 1, l - l\gamma^{z}]$. Group r of phase t contains the last $l\gamma^{r-1}$ sets of items in phase t. The weight of all edges from sets of items in group z in phase t is $W^{(t-1)r+z}$ for any $1 \leq z \leq r$ and $1 \leq t \leq l$.

Given an (α, β)-approximation algorithm $A_{\alpha,\beta}$, we similarly define $x_{t,i}$ and $y_{t,i}$ to be the expected number of items from set $O_{t,i}$ assigned to weight and cardinality bins by algorithm $A_{\alpha,\beta}$ respectively. We show in the following lemma that in order to have a high α, the algorithm should allocate a large fraction of sets of items in each group to the weight bins.

Lemma 2. *For any phase $1 \leq t \leq l$, and group $1 \leq z < r$, if the expected number of items assigned to cardinality bins in group z of phase t is at least $plC\gamma^{z-1}$ (which is p times the number of all items in groups $z, z+1, \cdots, r$ of phase t), the weight approximation ratio cannot be greater than $1 - 1/e - f(p)$ where $f(p)$ is $\frac{p}{e(\gamma+p)}$.*

We conclude this part with the main result of this subsection:

Theorem 4. *For any small $\epsilon > 0$, $1/e \leq \gamma < 1$, and $0 \leq p \leq 1 - \gamma$, any algorithm for bicriteria online matching problem with weight approximation guarantee, α, at least $1 - 1/e - f(p)$ cannot have cardinality approximation guarantee, β, greater than $p/(1 - \gamma) + \epsilon$.*

Proof. Using Lemma 2, for any group $1 \leq z < r$ in any phase $1 \leq t \leq l$, we know that at most p fraction of items are assigned to cardinality bins, because $1-1/e-f(p)$ is a strictly increasing function in p. Since in each phase the number of items is decreasing with a factor of γ in consecutive groups, the total fraction of items assigned to cardinality bins is at most $p + p\gamma + p\gamma^2 + \cdots + p\gamma^{r-2}$ plus the fraction of items assigned to cardinality in the last group r of phase t. Even if the algorithm assigns all of group r to cardinality, it does not achieve more than fraction γ^{r-1} from these items in each phase. Since the optimal cardinality algorithm can match all items, the cardinality approximation guarantee is at most $p(1+\gamma+\gamma^2+\cdots+\gamma^{r-2})+\gamma^{r-1}$. For large enough l (and consequently large enough r), this sum is not more than $p/(1-\gamma)+\epsilon$.

One way to compute the best values for p and γ corresponding to the best upper bound curve is to solve complex equations explicitly. Instead, we compute these values numerically by trying different values of p and γ which, in turn, yield the orange upper bound curve in Figure 1.

3 Algorithm for Large Capacities

We now turn to algorithms, to see how close one can come to matching the upper bounds of the previous section. In this section, we assume that the capacity $n(a)$ of each bin $a \in A$ is "large", and give an algorithm with the guarantees in Theorem 1 as $\min_{a\in A} n(a) \to \infty$.

Recall that our algorithm `Alg` uses two subroutines WeightAlg and CardinalityAlg, each of which, if given an online item, suggests a bin to place it in. Each item i is independently passed to WeightAlg with probability p and CardinalityAlg with the remaining probability $1-p$. First note that CardinalityAlg and WeightAlg are independent and unaware of each other; each of them thinks that the only items which exist are those passed to it. This allows us to analyze the two subroutines separately.

We now describe how `Alg` uses the subroutines. If WeightAlg suggests matching item i to a bin a, we match i to a. If a already has $n(a)$ items assigned to it in total, we remove any item assigned by CardinalityAlg arbitrarily; if all $n(a)$ were assigned by WeightAlg, we remove the item of lowest value for a. If CardinalityAlg suggests matching item i to a', we make this match unless a' has already had at least $n(a')$ total items assigned to it by both subroutines. In other words, the assignments of CardinalityAlg might be thrown away by some assignments of WeightAlg; however, the total number of items in a bin is always at least the the number assigned by CardinalityAlg. Items assigned by WeightAlg are never thrown away due to CardinalityAlg; they may only be replaced by later assignments of WeightAlg. Thus, we have proved the following proposition.

Proposition 1. *The weight and cardinality of the allocation of* `Alg` *are respectively at least as large as the weight of the allocation of* WeightAlg *and the cardinality of the allocation of* CardinalityAlg.

Note that the above proposition does not hold for any two arbitrary weight functions, and this is where we need one of the objectives to be cardinality. We now describe WeightAlg and CardinalityAlg, and prove Theorem 1. WeightAlg is essentially the exponentially-weighted primal-dual algorithm from [11], which was shown to achieve a $1-\frac{1}{e}$ approximation for the weighted online matching problem with large degrees. For completeness, we present the primal and dual LP relaxations for weighted matching below, and then describe the algorithm. In the primal LP, for each item i and bin a, variable x_{ia} denotes whether impression i is one of the $n(a)$ most valuable items for bin a.

Primal			**Dual**	
max	$\sum_{i,a} w_{ia} x_{ia}$			
$\sum_a x_{ia} \leq$	1	$(\forall\ i)$	$\min \sum_a n(a)\beta_a + \sum_i z_i$	
$\sum_i x_{ia} \leq$	$n(a)$	$(\forall\ a)$	$\beta_a + z_i \geq$	w_{ia} $(\forall i, a)$
$x_{ia} \geq$	0	$(\forall\ i, a)$	$\beta_a, z_i \geq$	0 $(\forall i, a)$

Following the techniques of Buchbinder *et al.* [5], the algorithm of [11] simultaneously maintains a feasible solution to the primal LP, and provides a feasible solution to the dual LP after all online nodes arrive. Each dual variable β_a is initialized to 0. When item i arrives online:

– Assign i to the bin $a' = \arg\max_a\{w_{ia} - \beta_a\}$. (If this quantity is negative for all a, discard i.)
– Set $x_{ia'} = 1$. If a' previously had $n(a')$ items assigned to it, set $x_{i'a'} = 0$ for the least valuable item i' previously assigned to a'.
– In the dual solution, set $z_i = w_{ia'} - \beta_{a'}$ and *update* dual variable $\beta_{a'}$ as described below.

Definition 1 (Exponential Weighting). *Let $w_1, w_2, \ldots w_{n(a)}$ be the weights of the $n(a)$ items currently assigned to bin a, sorted in non-increasing order, and padded with 0s if necessary.*
Set $\beta_a = \frac{1}{p \cdot n(a) \cdot \left((1+1/p \cdot n(a))^{n(a)} - 1\right)} \sum_{j=1}^{n(a)} w_j \left(1 + \frac{1}{p \cdot n(a)}\right)^{j-1}$.

Lemma 3. *If* WeightAlg *is the primal-dual algorithm, with dual variables β_a updated according to the exponential weighting rule defined above, the competitive ratio of* WeightAlg *regarding the weight objective is at least $p \cdot \left(1 - \frac{1}{k}\right)$ where $k = \left(1 + \frac{1}{p \cdot d}\right)^d$, and $d = \min_a\{n(a)\}$. Note that $\lim_{d \to \infty} k = e^{1/p}$.*

We provide some brief intuition here. If *all* items are passed to WeightAlg, it was proved in [11] that the algorithm has competitive ratio tending to $1 - 1/e$ as $d = \min_a\{n(a)\}$ tends to ∞; this is the statement of Lemma 3 when $p = 1$. Now, suppose each item is passed to WeightAlg with probability p. The expected value of the optimum matching induced by those items passed to WeightAlg is at least $p \cdot W_{\text{opt}}$, and this is *nearly* true (up to $o(1)$ terms) *even if we reduce the capacity of each bin a to $p \cdot n(a)$*. This follows since W_{opt} assigns at most $n(a)$ items to bin a, and as we are unlikely to sample more than $p \cdot n(a)$ of these items for the reduced instance, we do not lose much by reducing capacities. But note that WeightAlg can use the entire capacity $n(a)$, while there is a solution of value close to pW_{opt} even with capacities $p \cdot n(a)$. This extra capacity allows an improved competitive ratio of $1 - \frac{1}{e^{1/p}}$, proving the lemma.

Algorithm CardinalityAlg is identical to WeightAlg, except that it assumes all items have weight 1 for each bin. Since items are assigned to CardinalityAlg with probability $1 - p$, Lemma 3 implies the following corollary. This concludes the proof of Theorem 1.

Corollary 1. *The total cardinality of the allocation of* CardinalityAlg *is at least* $(1-p)\cdot\left(1-\frac{1}{k}\right)$*, where* $k = \left(1+\frac{1}{(1-p)\cdot d}\right)^d$*, and* $d = \min_a\{n(a)\}$*. Note that* $\lim_{d\to\infty} k = e^{1/(1-p)}$.

4 Algorithm for Small Capacities

We now consider algorithms for the case when the capacities of bins are not large. Without loss of generality, we assume that the capacity of each bin is one, because we can think about a bin with capacity c as c identical bins with capacity one. So we have a set A of bins each with capacity one, and a set of items I arriving online. As before, we use two subroutines WeightAlg and CardinalityAlg, but the algorithms are slightly different from those in the previous section. Each item $i \in I$ is independently passed to WeightAlg with probability p and CardinalityAlg with the remaining probability $1-p$.

In WeightAlg, we match item i (that has been passed to WeightAlg) to the bin that maximizes its marginal value. Formally we match i to bin $a = \arg\max_{a\in A}(w_{i,a} - w_{i',a})$ where i' is the last item assigned to a before item i.

In CardinalityAlg, we run the RANKING algorithm presented in [18]. So CardinalityAlg chooses a permutation π uniformly at random on the set of bins A, assigns an item i (that has been passed to it) to the bin a that is available, has the minimum rank in π, and there is also an edge between i and a.

4.1 $p/(p+1)$ lower bound on the weight approximation ratio

Let $n = |I|$ be the number of items. We denote the ith arrived item by i. Let a_i be the bin that i is matched to in W_{opt} for any $1 \le i \le n$. One can assume that all unmatched items in the optimum weight allocation are matched with zero-weight edges to an imaginary bin. So W_{opt} is equal to $\sum_{i=1}^{n} w_{i,a_i}$. Let S be the set of items that have been passed to WeightAlg. If WeightAlg matches item i to bin a_j for some $j > i$, we call this a forwarding allocation (edge) because item j (the match of a_j in W_{opt}) has not arrived yet. We call it a selected forwarding edge if $j \in S$. We define the marginal value of assigning item i to bin a to be w_{ia} minus the value of any item previously assigned to a.

Lemma 4. *The weight of the allocation of* WeightAlg *is at least* $(p/(p+1))W_{opt}$*.*

Proof. Each forwarding edge will be a selected forwarding edge with probability p because $Pr[j \in S]$ is p for any $j \in I$. Let F be the total weight of forwarding edges of WeightAlg, where by weight of a forwarding edge, we mean its marginal value (not the actual weight of the edge). Similarly, we define F_s to be the sum of marginal values of selected forwarding edges. We have the simple equality that the expected value of F, $E(F)$, is $E(F_s)/p$. We define W' and W_s to be the total marginal values of allocation of WeightAlg, and the sum $\sum_{i\in S} w_{i,a_i}$. We know that $E(W_s)$ is pW_{opt} because $Pr[i \in S]$ is p. We prove that W' is at least $W_s - F_s$.

For every item i that has been selected to be matched by WeightAlg, we get at least marginal value w_{i,a_i} minus the sum of all marginal values of items that have been assigned to bin a by WeightAlg up to now. If we sum up all these lower bounds on our gains for all selected items, we get $W_s(= \sum_{i \in S} w_{i,a_i})$ minus the sum of all marginal values of items that has been assigned to a_i before item i arrives for all $i \in S$. The latter part is exactly the definition of F_s. Therefore W' is at least $W_s - F_s$. We also know that $W' \geq F$. Using $E[F] \geq E[F_s]/p$, we have that $E(W')$ is at least $E(W_s) - pE(W')$, and this yields the $p/(p+1)$ approximation factor.

Corollary 2. *The weight and cardinality approximation guarantees of* `Alg` *are at least* $p/(p+1)$ *and* $(1-p)/(1-p+1)$ *respectively.*

4.2 Factor Revealing Linear Program for CardinalityAlg

Due to limited space, we just mention that we get the lower bounds on the cardinality competitive ratio in the green curve of Figure 1 using the following LP. This LP gives a valid lower bound for any integer k for $p = 1/2$. A few simple adjustments generalize this factor revealing LP to the general case of arbitrary $0 < p < 1$.

$$\begin{array}{lll}
\text{Minimize:} & \beta & \\
\forall 1 < i \leq k\text{:} & s_i \geq s_{i-1} \quad \& \ sf_i \geq sf_{i-1} & \& \ sb_i \geq sb_{i-1} \\
\forall 1 \leq i \leq k\text{:} & t_i \geq t_{i-1} \quad \& \ t_i \geq sf_i & \& \ s_i = sf_i + sb_i \\
 & \beta \geq s_k + t_k \ \& \ \beta \geq 1/2 - sf_k & \\
\forall 1 < i \leq k\text{:} & s_i - s_{i-1} \quad \geq 1/2k - (s_i + t_i)/k &
\end{array}$$

References

1. S. Agrawal, Z. Wang, and Y. Ye. A dynamic near-optimal algorithm for online linear programming. Working paper posted at http://www.stanford.edu/ yyye/.
2. A. Bhalgat, J. Feldman, and V. S. Mirrokni. Online ad allocation with smooth delivery. In *ACM Conference on Knowledge Discovery (KDD)*, 2012.
3. V. Bilò, M. Flammini, and L. Moscardelli. Pareto approximations for the bicriteria scheduling problem. *Journal of Parallel and Distributed Computing*, 66(3):393–402, 2006.
4. B. Birnbaum and C. Mathieu. On-line bipartite matching made simple. *SIGACT News*, 39(1):80–87, 2008.
5. N. Buchbinder, K. Jain, and J. Naor. Online Primal-Dual Algorithms for Maximizing Ad-Auctions Revenue. In *Proc. ESA*, page 253. Springer, 2007.
6. N. Buchbinder and J. Naor. Fair online load balancing. In *Proceedings of the eighteenth annual ACM symposium on Parallelism in algorithms and architectures*, pages 291–298. ACM, 2006.
7. N. Devanur and T. Hayes. The adwords problem: Online keyword matching with budgeted bidders under random permutations. In *ACM EC*, 2009.

8. N. R. Devanur, K. Jain, B. Sivan, and C. A. Wilkens. Near optimal online algorithms and fast approximation algorithms for resource allocation problems. In *ACM Conference on Electronic Commerce*, pages 29–38, 2011.
9. N. R. Devanur, B. Sivan, and Y. Azar. Asymptotically optimal algorithm for stochastic adwords. In *ACM ee on Electronic Commerce*, pages 388–404, 2012.
10. J. Feldman, M. Henzinger, N. Korula, V. S. Mirrokni, and C. Stein. Online stochastic packing applied to display ad allocation. In M. de Berg and U. Meyer, editors, *Algorithms - ESA 2010, 18th Annual European Symposium, Liverpool, UK, September 6-8, 2010. Proceedings, Part I*, volume 6346 of *Lecture Notes in Computer Science*, pages 182–194. Springer, 2010.
11. J. Feldman, N. Korula, V. S. Mirrokni, S. Muthukrishnan, and M. Pál. Online ad assignment with free disposal. In S. Leonardi, editor, *Internet and Network Economics, 5th International Workshop, WINE 2009, Rome, Italy, December 14-18, 2009. Proceedings*, volume 5929 of *Lecture Notes in Computer Science*, pages 374–385. Springer, 2009.
12. J. Feldman, A. Mehta, V. Mirrokni, and S. Muthukrishnan. Online stochastic matching: Beating 1 - 1/e. In *FOCS*, 2009.
13. M. Flammini and G. Nicosia. On multicriteria online problems. In *Proceedings of the 8th Annual European Symposium on Algorithms*, pages 191–201. Springer-Verlag, 2000.
14. A. Goel, A. Meyerson, and S. Plotkin. Approximate majorization and fair online load balancing. In *Proceedings of the twelfth annual ACM-SIAM symposium on Discrete algorithms*, pages 384–390. Society for Industrial and Applied Mathematics, 2001.
15. A. Goel, A. Meyerson, and S. Plotkin. Combining fairness with throughput: Online routing with multiple objectives. *Journal of Computer and System Sciences*, 63(1):62–79, 2001.
16. B. Haeupler, V. Mirrokni, and M. ZadiMoghaddam. Online stochastic weighted matching: Improved approximation algorithms. In *WINE*, 2011.
17. C. Karande, A. Mehta, and P. Tripathi. Online bipartite matching with unknown distributions. In *STOC*, 2011.
18. R. M. Karp, U. V. Vazirani, and V. V. Vazirani. An optimal algorithm for on-line bipartite matching. In *STOC*, pages 352–358, 1990.
19. N. Korula, V. S. Mirrokni, and Q. Yan. Whole-page ad allocation and. In *Ad Auctions Workshop*, 2012.
20. M. Mahdian and Q. Yan. Online bipartite matching with random arrivals: A strongly factor revealing lp approach. In *STOC*, 2011.
21. A. Mehta, A. Saberi, U. V. Vazirani, and V. V. Vazirani. Adwords and generalized online matching. *J. ACM*, 54(5), 2007.
22. H. Menshadi, S. OveisGharan, and A. Saberi. Offline optimization for online stochastic matching. In *SODA*, 2011.
23. V. Mirrokni, S. O. Gharan, and M. ZadiMoghaddam. Simultaneous approximations for adversarial and stochastic online budgeted allocation problems. In *SODA*, 2012.
24. E. Vee, S. Vassilvitskii, and J. Shanmugasundaram. Optimal online assignment with forecasts. In *ACM EC*, 2010.
25. C. Wang, X. Huang, and C. Hsu. Bi-objective optimization: An online algorithm for job assignment. *Advances in Grid and Pervasive Computing*, pages 223–234, 2009.