On the Perturbation Function of Ranking and Balance for Weighted Online Bipartite Matching

Jingxun Liang
IIIS, Tsinghua University, Beijing, China

Zhihao Gavin Tang
ITCS, Shanghai University of Finance and Economics, China

Yixuan Even Xu
IIIS, Tsinghua University, Beijing, China

Yuhao Zhang
Shanghai Jiao Tong University, China

Renfei Zhou
IIIS, Tsinghua University, Beijing, China

Abstract

Ranking and Balance are arguably the two most important algorithms in the online matching literature. They achieve the same optimal competitive ratio of $1 - 1/e$ for the integral version and fractional version of online bipartite matching by Karp, Vazirani, and Vazirani (STOC 1990) respectively. The two algorithms have been generalized to weighted online bipartite matching problems, including vertex-weighted online bipartite matching and AdWords, by utilizing a perturbation function. The canonical choice of the perturbation function is $f(x) = 1 - e^{x-1}$ as it leads to the optimal competitive ratio of $1 - 1/e$ in both settings.

We advance the understanding of the weighted generalizations of Ranking and Balance in this paper, with a focus on studying the effect of different perturbation functions. First, we prove that the canonical perturbation function is the *unique* optimal perturbation function for vertex-weighted online bipartite matching. In stark contrast, all perturbation functions achieve the optimal competitive ratio of $1 - 1/e$ in the unweighted setting. Second, we prove that the generalization of Ranking to AdWords with unknown budgets using the canonical perturbation function is at most 0.624 competitive, refuting a conjecture of Vazirani (2021). More generally, as an application of the first result, we prove that no perturbation function leads to the prominent competitive ratio of $1 - 1/e$ by establishing an upper bound of $1 - 1/e - 0.0003$. Finally, we propose the online budget-additive welfare maximization problem that is intermediate between AdWords and AdWords with unknown budgets, and we design an optimal $1 - 1/e$ competitive algorithm by generalizing Balance.

2012 ACM Subject Classification Theory of computation → Online algorithms

Keywords and phrases Online Matching, AdWords, Ranking, Water-Filling

Digital Object Identifier 10.4230/LIPIcs.ESA.2023.80

Related Version *Full Version*: `https://arxiv.org/abs/2210.10370`

Supplementary Material
Software (Source Code): `https://github.com/orbitingflea/perturbation-function`
archived at `swh:1:dir:24101e3982129e07e5c1f644f0ff611800a98730`

Funding This work is supported by Science and Technology Innovation 2030 – "New Generation of Artificial Intelligence" Major Project No.(2018AAA0100903), Program for Innovative Research Team of Shanghai University of Finance and Economics (IRTSHUFE) and the Fundamental Research Funds for the Central Universities.
Zhihao Gavin Tang: Zhihao Gavin Tang is partially supported by Huawei Theory Lab.
Yuhao Zhang: Yuhao Zhang is supported by National Natural Science Foundation of China, Grant No. 62102251.

Acknowledgements We thank all anonymous reviewers for their insightful comments.

1 Introduction

Online bipartite matching has been extensively studied since the seminal work of Karp, Vazirani, and Vazirani [22]. Two remarkable algorithms, Ranking of Karp et al. [22] and Balance of Kalyanasundaram and Pruhs [19], achieve the same optimal competitive ratio of $1 - 1/e$ for the integral (randomized) and fractional (deterministic) version of the problem respectively.

AdWords and Vertex-weighted. Motivated by the display of advertising on the Internet, Mehta et al. [25] generalized the online bipartite matching problem so that it allows weighted graphs. Consider an underlying bipartite graph $G = (L \cup R, E)$ with L, R being offline and online vertices. Each vertex $u \in L$ is associated with a budget B_u, and each edge $(u, v) \in E$ is associated with a bid w_{uv}. The offline vertices and their corresponding budgets are known in advance. The online vertices arrive one at a time, with their incident edges and associated bids, and have to be matched immediately and irrevocably to some $u \in L$. An offline vertex $u \in L$ can be matched to multiple online vertices. Let S_u be the set of online vertices matched to u and then, the revenue of u equals $\min\{B_u, \sum_{v \in S_u} w_{uv}\}$, that is, the revenue cannot exceed the budget. The goal is to maximize the total revenue and to compete against the optimal revenue of an offline algorithm that knows the whole graph.

Mehta et al. [25] established an optimal $(1 - 1/e)$-competitive algorithm for the fractional version[1] of the problem by generalizing Balance to Perturbed-Balance. Later, Aggarwal et al. [1] considered a restricted setting of AdWords, called vertex-weighted online bipartite matching, in which all edges incident to u have the same weight of $w_u = B_u$. They generalized Ranking to Perturbed-Ranking and obtained the same $1 - 1/e$ competitive ratio for the integral version of the problem.

The two generalizations are both greedy-based algorithms with a careful perturbation of the weights. Specifically, upon the arrival of each vertex v, the algorithm matches it with the offline vertex u with the maximum perturbed weight $f(x_u) \cdot w_{uv}$, among those neighbors whose budgets have not yet been exhausted. Here, x_u corresponds to the random rank of u in Perturbed-Ranking and the fraction of budget spent so far in Perturbed-Balance. The canonical choice of the perturbation function is $f(x) = 1 - e^{x-1}$, applied by Mehta et al. [25] and Aggarwal et al. [1]. Notably, Devanur, Jain, and Kleinberg [7] provided a unified primal-dual analysis for Perturbed-Ranking and Perturbed-Balance, in which the perturbation function plays a critical role even for the unweighted online bipartite matching problem.

Despite the above successful generalizations of Ranking and Balance from unweighted to weighted graphs, we lack an understanding of the extra difficulty introduced by weighted graphs. In this paper, we revisit the two classic algorithms and focus on the perturbation function. Notice that for unweighted graphs, Perturbed-Ranking (resp. Perturbed-Balance) with an arbitrary perturbation function degenerates to the same Ranking (resp. Balance) algorithm and achieves the optimal competitive ratio of $1 - 1/e$. We examine the importance of perturbation functions by studying the performance of Perturbed-Ranking and Perturbed-Balance on weighted graphs with an arbitrary perturbation function.

Our first result confirms the importance of the perturbation function, proving that the canonical perturbation function $f(x) = 1 - e^{x-1}$ is the unique optimal perturbation function for vertex-weighted online bipartite matching.

[1] The paper states their result with the small-bid assumption, i.e., $\gamma = \max_{u,v} w_{uv}/B_u$ is small. We consider the fractional AdWords problem, corresponding to the case when $\gamma \to 0$. See our discussion in Section 2.

▶ **Theorem 1.** *The perturbation function* $f(x) = 1 - e^{x-1}$ *is unique (up to a scale factor) for* Perturbed-Ranking *and* Perturbed-Balance *to achieve the optimal competitive ratio of* $1 - 1/e$ *for vertex-weighted online bipartite matching.*

It is surprising that this question has been overlooked by the online matching community. We introduce a new family of hard instances that heavily exploits the power of weighted graphs. Noticeably, prior to our work, the only $1 - 1/e$ impossibility result by Karp et al. [22] is established for unweighted graphs. This advanced understanding is also the starting point for us to explore the limitation of Perturbed-Ranking in a more general model, i.e., AdWords with unknown budgets.

AdWords with Unknown Budgets. A major open question left by Mehta et al. [25] is the competitive ratio of Perturbed-Ranking for the fractional AdWords problem. In addition to obvious theoretical interests, the Perturbed-Ranking algorithm has a merit of budget-obliviousness, as pointed out by Vazirani [28] and Udwani [27]. I.e., the algorithm does not need to know the budget of each vertex, in contrast to the Perturbed-Balance algorithm. Formally, consider the setting of AdWords with unknown budgets: the algorithm has no prior knowledge of the budgets and only learns the budget of each vertex u when the total revenue of u first exceeds its budget. Observe that Perturbed-Balance is not applicable in this setting, since its decision at each step depends on the fraction of budget spent on each offline vertex.

Perturbed-Ranking is the only known algorithm for AdWords with unknown budgets so far. Using the canonical perturbation function $f(x) = 1 - e^{x-1}$, Vazirani [28] proved Perturbed-Ranking is $(1 - 1/e)$-competitive assuming a no-surpassing property. Udwani [27] proved that the algorithm is 0.508-competitive in the general case and is 0.522-competitive with a different perturbation function $f(x) = 1 - e^{1.15(x-1)}$.

It is natural to ask if other perturbation functions can lead to a better competitive ratio, or even $1 - 1/e$. In this paper, we give a limitation of *all* perturbation functions, showing a separation between vertex-weighted online bipartite matching and AdWords on the performance of Perturbed-Ranking.

We first show that Perturbed-Ranking with the canonical perturbation function can only achieve a competitive ratio of at most $0.624 < 1 - 1/e$. Then, together with the family of instances we constructed in the proof of Theorem 1, we manage to prove that any perturbation function cannot lead to the prominent competitive ratio of $1 - 1/e$:

▶ **Theorem 2.** *The competitive ratio of* Perturbed-Ranking *algorithm with any perturbation function* $f(x)$ *on fractional AdWords is at most* $1 - 1/e - 0.0003$*. In particular, using the canonical function* $f(x) = 1 - e^{x-1}$*, the competitive ratio is at most* 0.624*.*

Our result refutes the conjecture of Vazirani [28] that Perturbed-Ranking is $1 - 1/e$ competitive. Moreover, our construction is clean and simple, suggesting that the no-surpassing assumption might be too strong to hold in reality. Such result leads to the conjecture that there is no $1 - 1/e$ competitive algorithm for AdWords with unknown budgets.

Remark. Very recently, independently by our work, Udwani [27] updated his paper and proved that a specific perturbation function family $f(x) = 1 - e^{\beta(x-1)}$ is at most 0.624-competitive for any $\beta > 0$. Our result provides a stronger observation by another approach that shows *all* perturbation functions cannot achieve the competitive ratio of $1 - 1/e$.

Online Budget-additive Welfare Maximization. The above upper bound of the competitive ratio for Perturbed-Ranking suggests that AdWords with unknown budgets should be strictly harder than AdWords, in terms of the worse-case competitive ratio. Unfortunately, our current construction is specific to the Perturbed-Ranking algorithm and does not serve as a problem hardness.

Inspired by the online submodular welfare maximization problem (that we discuss below in the related work section), we consider a variant which we call online budget-additive welfare maximization problem, that lies in between the original AdWords and AdWords with unknown budgets. Specifically, we assume that 1) the algorithm has no information of the budgets at the beginning, 2) at each step, the algorithm can query for each vertex u, the value of $w_u(S) = \min\{B_u, \sum_{v \in S} w_{uv}\}$ for any subset S of arrived vertices. Notice that AdWords with unknown budgets can be interpreted in a similar way, except that the algorithm can only query those sets S that are subsets of $S(u) \cup \{v\}$ where $S(u)$ is the set of current matched vertices to u.

Our final result is an optimal algorithm for the fractional version of the problem. We hope it would shed some light on designing online algorithms beyond Perturbed-Ranking in the AdWords with unknown budgets setting and designing algorithms beyond greedy (with unrestricted computational power) in the online submodular welfare maximization setting.

▶ **Theorem 3.** *There exists a fractional algorithm that achieves the competitive ratio of* $(1 - 1/e)$ *for the Online Budget-Additive Welfare Maximization problem.*

Roadmap. Section 2 presents the formal definitions of the Perturbed-Ranking and Perturbed-Balance algorithms. We prove Theorem 1 in Section 3 and Theorem 2 in Section 4. Due to space limit, we provide the proof of Theorem 3 in the full version.

1.1 Related Work

The seminal work of Karp et al. [22] studied the unweighted and one-sided online bipartite matching model and proposed the optimal $(1 - 1/e)$-competitive algorithm: Ranking. The analysis of Ranking has been refined and simplified by a series of works [2, 7, 11, 8]. Kalyanasundaram and Pruhs studied the b-matching model and designed Balance (fractional) that also achieves the competitive ratio of $1 - 1/e$. The model has been generalized to many weighted variants, e.g., vertex-weighted [1, 14, 15, 18], edge-weighted [3, 9, 10], and AdWords [6, 25, 17]. Besides the aforementioned generalization of Ranking and Balance to the vertex-weighted and AdWords settings, Huang et al. [15] generalized Ranking to the vertex-weighted setting with random arrival order, by utilizing a two-dimensional perturbation function. They achieved a competitive ratio of 0.653 that is subsequently improved to 0.662 by Jin and Williamson [18]. Another line of work adapts Ranking and Balance to other matching problems, including online bipartite matching with random arrivals [21, 24], oblivious matching [5, 26] and fully online matching [12, 13, 16].

The most general extension of online bipartite matching is the online submodular welfare maximization problem. It captures most of the weighted variants of online bipartite matching discussed above. In this problem, a set of n offline vertices are given, each associated with a monotone submodular function w_u. Upon the arrival of an online vertex, it must be assigned to one of the offline vertices and the goal is to maximized the welfare $\sum_u w_u(S_u)$, where S_u is the set of vertices received by u. The algorithm is assumed to have value oracles for the functions. I.e., an algorithm can query the value of $w_u(S)$ for an arbitrary subset S of arrived online vertices. Kapralov et al. [20] proved that the 0.5-competitive greedy algorithm is optimal with restricted computational powers. For the unknown i.i.d. setting, they provided an optimal $(1-1/e)$-competitive algorithm. In the random arrival model, Korula et al. [23] proved that greedy is at least 0.5052-competitive, and the ratio is improved to 0.5096 by Buchbinder et al. [4]. Our budget-additive welfare maximization problem is a special case of the submodular welfare maximization problem where every w_u is a budget-additive function and admits an $(1-1/e)$-competitive algorithm.

Moreover, the AdWords with unknown budgets problem suggests us to study a more restricted oracle access for online submodular welfare maximization. We call it *marginal oracle*, that on the arrival of an online vertex v, the algorithm can only query the value of $w_u(S)$ for $S \subseteq S_u(v) \cup \{v\}$, where $S_u(v)$ is the current matched vertex set to u. Based on our discoveries in this paper, we make the following three conjectures for future work:

- Online submodular welfare maximization with marginal oracles does *not* admit a $1-1/e$ competitive algorithm.
- AdWords with unknown budgets does *not* admit a $1-1/e$ competitive algorithm.
- Online submodular welfare maximization admits a $0.5+\Omega(1)$ competitive algorithm.

All the three conjectures assume unlimited computational powers so that the third conjecture does not violate the impossibility result of [20]. Observe that if the second conjecture holds, it automatically confirms the first conjecture, and implies a price of budget-obliviousness for AdWords.

2 Preliminaries

We first give the formal definitions of Perturbed-Ranking and Perturbed-Balance for the vertex-weighted online bipartite matching problem and then discuss their extensions to the fractional AdWords problem. Both algorithms depends on a perturbation function.

▶ **Definition 4** (Perturbation Function). *A perturbation function is a non-increasing and right continuous function $f(x):[0,1]\to[0,1]$.*

2.1 Vertex-weighted

Given a perturbation function f, the two algorithms are defined as below. Observe that Perturbed-Ranking is a randomized algorithm and Perturbed-Balance is a deterministic algorithm.

▶ **Definition 5** (Perturbed-Ranking [1]). *Sample a rank y_u for each offline vertex $u \in L$ independently from a uniform distribution on $[0,1]$. On the arrival of an online vertex v, we match v to the unmatched neighbor u with maximum perturbed weight $f(y_u)\cdot w_u$.*

▶ **Definition 6** (Perturbed-Balance). *On the arrival time of an online vertex v, we continuously match v to the offline neighbor u with maximum perturbed weight $f(x_u)\cdot w_u$, where x_u is the current matched portion of u.*

We remark that in the context of Perturbed-Ranking, a perturbation function can be interpreted as an alternative representation of a $[0,1]$-bounded random variable, in which $f(x)$ corresponds to the value of a quantile x. Moreover, the right continuity is necessary for the Perturbed-Balance algorithm to be well-defined.

2.2 AdWords

In Section 4, we shall work on the fractional version of AdWords (and its variant) that is (slightly) more relaxed than the AdWords problem with small bid assumption. See e.g. Udwani [27] for a more detailed discussion.

Fractional AdWords. The fractional AdWords problem allows each edge (u, v) to be fractionally matched by an amount of $x_{uv} \in [0, 1]$, as long as the total matched portion of each online vertex v is no more than a unit, i.e., $\sum_u x_{uv} \le 1$.

Consider the following generalizations of Perturbed-Ranking and Perturbed-Balance for the fractional AdWords problem:

▶ **Definition 7** (Perturbed-Ranking [28, 27])**.** *Sample a rank y_u for each offline vertex $u \in L$ independently from a uniform distribution on* $[0, 1]$*. On the arrival of an online vertex v, we* continuously *match v to the neighbor u with maximum perturbed weight $f(y_u) \cdot w_{uv}$, among those neighbors whose budgets have not been exhausted yet.*

▶ **Definition 8** (Perturbed-Balance (a.k.a. MSVV [25]))**.** *On the arrival time of an online vertex v, we continuously match v to the offline neighbor u with maximum perturbed weight $f(x_u/B_u) \cdot w_{uv}$, where x_u is the current used budget of u.*

3 Vertex-Weighted

In this section, we consider vertex-weighted online bipartite matching. We prove that to achieve the optimal competitive ratio of $\Gamma = 1 - 1/e$, the canonical choice of the perturbation function $f(x) = 1 - e^{x-1}$ is unique (up to a scale factor).

Our result holds for both Perturbed-Ranking and Perturbed-Balance. Indeed, we establish a dominance of Perturbed-Balance over Perturbed-Ranking in terms of worst-case competitive ratio.

▶ **Lemma 9.** *For any perturbation function f, the competitive ratio of* Perturbed-Ranking *is at most the competitive ratio of* Perturbed-Balance *for vertex-weighted online bipartite matching.*

We sketch our proof below and provide the detailed proof in the full version.

Proof Sketch. Given an arbitrary instance $G = (L \cup R, E, w)$, we construct an instance G' so that the competitive ratio of Perturbed-Balance for G and the competitive ratio of Perturbed-Ranking for G' are approximately the same.

- For each offline vertex $u \in L$, create N duplicates $\{u^{(i)}\}_{i=1}^N$ in G' and with weights $w_{u^{(i)}} = w_u$.
- For each online vertex $v \in R$, create N duplicates $\{v^{(i)}\}_{i=1}^N$ in G' that arrive in a sequence.
- For each $(u, v) \in E$, let there be a complete bipartite graph between $\{u^{(i)}\}$ and $\{v^{(i)}\}$ in G'.

Now, we consider the behavior of Perturbed-Ranking on G'. Intuitively, although the ranks are drawn independently for each offline vertex, the set of N ranks of $\{u^{(i)}\}_{i=1}^N$ should be "close" to $\left\{\frac{1}{N}, \frac{2}{N}, \ldots, 1\right\}$ with high probability when N is sufficiently large. Formally, we prove the following mathematical fact in the full version.

▶ **Lemma 10.** *Let $x_1, \ldots, x_n$ be i.i.d. random variables sampled from* $[0, 1]$ *uniformly. Let y_i be the i-th order statistics of $\{x_1, \ldots, x_n\}$, for $i = 1, \ldots, n$. Then for any parameter ε with $4n^{-1/4} < \varepsilon < 1$, we have*

$$\Pr_{x_1,\ldots,x_n}\left[\left|y_i - \frac{i}{n}\right| \le \varepsilon, \forall i \in [n]\right] \ge 1 - 2ne^{-\sqrt{n}/6}. \tag{3.1}$$

For now, we assume the set of ranks of $\{u^{(i)}\}$ is $\{\frac{1}{N}, \frac{2}{N}, \dots, 1\}$ for each vertex u. Then, upon the arrival of $\{v^{(i)}\}$, we are basically running a discretized version of the Perturbed-Balance algorithm, with a step size of $\frac{1}{N}$. To formalize this intuition, we can introduce a family of ε-approximate Perturbed-Balance algorithms that approaches the behavior of Perturbed-Balance when $\varepsilon \to 0$. Moreover, we prove that the Perturbed-Ranking algorithm can be interpreted as an ε-approximate Perturbed-Balance algorithm when the ranks of $\{u^{(i)}\}$ behave nicely (which is of high probability). Finally, we conclude the proof of the lemma by letting N go to infinity. The detail is provided in the full version. ◀

Equipped with the above lemma, we hereafter focus on the easy-to-analyze Perturbed-Balance algorithm, since it is deterministic while Perturbed-Ranking is randomized.

Our main construction is a family of instances that strongly restricts the shape of the perturbation function. Naturally, our construction is built upon the classical upper triangle graph that gives the tight $1 - 1/e$ competitive ratio for unweighted online bipartite matching. On the other hand, our construction consists of a few novel gadgets that might be useful for other weighted online matching problems. The following lemma also serves as a stepping stone of our result for the AdWords with unknown budget problem in Section 4.

▶ **Lemma 11.** *If the* Perturbed-Balance *algorithm achieves a competitive ratio of* Γ *for vertex-weighted online bipartite matching, then the perturbation function* f *satisfies the following:*

$$\left(\beta + 1 - e^{\beta-1} - \Gamma\right) \cdot f(\alpha) \geq (1 - (1-\Gamma) \cdot e^{\alpha}) \cdot \int_0^{\beta} f(x)\mathrm{d}x, \quad \forall \alpha, \beta \in [0,1]. \tag{3.2}$$

We defer its proof till the end of the section and proceed by first proving our main theorem.

▶ **Theorem 1.** *The perturbation function* $f(x) = 1 - e^{x-1}$ *is unique (up to a scale factor) for* Perturbed-Ranking *and* Perturbed-Balance *to achieve the optimal competitive ratio of* $1 - 1/e$ *for vertex-weighted online bipartite matching.*

Proof. By Lemma 9, it suffices to prove the theorem for Perturbed-Balance. By Lemma 11 with $\Gamma = 1 - 1/e$, the perturbation function $f(x)$ satisfies the following:

$$\frac{f(\alpha)}{1 - e^{\alpha-1}} \geq \frac{\int_0^{\beta} f(x)\mathrm{d}x}{\beta - e^{\beta-1} + e^{-1}}, \quad \forall \alpha, \beta \in (0,1).$$

Let $M \stackrel{\text{def}}{=} \inf_{\alpha \in (0,1)} \frac{f(\alpha)}{1-e^{\alpha-1}}$. We must then have

$$f(\alpha) \geq M(1 - e^{\alpha-1}), \quad \forall \alpha \in [0,1], \tag{3.3}$$

$$\int_0^{\beta} f(x)\mathrm{d}x \leq M(\beta - e^{\beta-1} + e^{-1}), \quad \forall \beta \in [0,1]. \tag{3.4}$$

Taking the integral of $f(x)$ and applying (3.3), we have

$$\int_0^{\beta} f(x)\mathrm{d}x \geq M \int_0^{\beta} (1 - e^{x-1})\mathrm{d}x = M(\beta - e^{\beta-1} + e^{-1}).$$

Together with (3.4), we conclude that $f(x) = M(1 - e^{x-1})$ for all $x \in [0,1]$ according to the right-continuity of function f. Specifically, if there exists an $x^* \in [0,1)$, that $f(x^*) = M(1 - e^{x^*-1}) + \varepsilon$ for some $\varepsilon > 0$, then there exists a sufficiently small $\delta > 0$ such that for any $x \in [x^*, x^* + \delta]$, $f(x) \geq M(1 - e^{x-1}) + \frac{\varepsilon}{2}$. Then we can see by (3.3) that

$$\int_0^1 f(x)\mathrm{d}x \geq \int_0^1 \Big(M(1-e^{x-1}) + \mathbb{1}[x^* \leq x \leq x^*+\delta]\cdot\frac{\varepsilon}{2}\Big)\mathrm{d}x = \frac{M}{e} + \frac{\varepsilon\delta}{2},$$

which violates the statement of (3.4), $\int_0^1 f(x)\mathrm{d}x \leq \frac{M}{e}$. Therefore, $\forall x \in [0,1)$, $f(x) = M(1-e^{x-1})$. Also, for $f(1)$, note that $f(x)$ is decreasing, so $f(1) \leq \lim_{x\to 1^-} f(x) = 0$. Then $f(1) = 0$.

This shows that $\forall x \in [0,1]$, $f(x) = M(1-e^{x-1})$, concluding the proof of the theorem. ◀

3.1 Proof of Lemma 11

Fix $\alpha, \beta \in [0,1]$. Let n, m be sufficiently large numbers. Refer to Figure 3.1 for our instance.

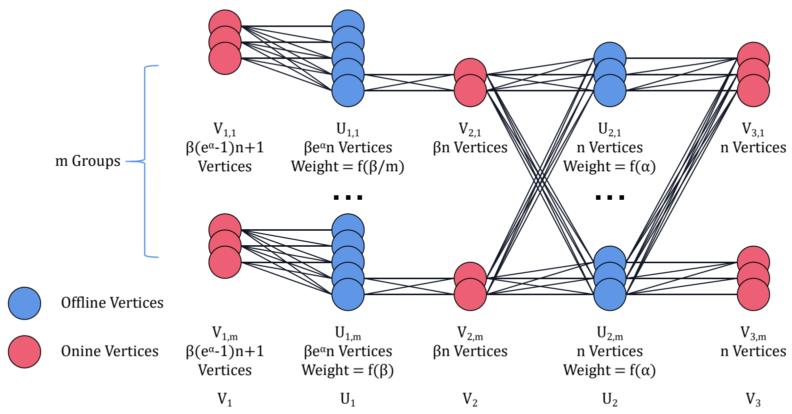

Figure 3.1 Instance 1.

Our construction consists of m groups of vertices, and each group consists of 5 parts. We use $V_{1,i}, V_{2,i}, V_{3,i}$ to denote the three online parts of group i and $U_{1,i}, U_{2,i}$ to denote the two offline parts of group i. Let $V_j = \cup_{i\in m} V_{j,i}$ for $j \in \{1,2,3\}$ and $U_j = \cup_{i\in m} U_{j,i}$ for $j \in \{1,2\}$. We first define the vertices of the graph: for each $i \leq m$,

- $U_{1,i}$ consists of $(\beta e^\alpha n + 1)$ offline vertices[2] with the same weight of $f(\frac{i}{m}\cdot\beta)$.
- $U_{2,i}$ consists of n offline vertices with the same weight of $f(\alpha)$.
- $V_{1,i}, V_{2,i}, V_{3,i}$ consist of $\beta(e^\alpha - 1)n, \beta n, n$ online vertices, respectively.

The arrival order of the vertices is the following:

$$V_{1,1} \to V_{1,2} \to \cdots \to V_{1,m} \to V_{2,1} \to V_{2,2} \to \cdots \to V_{2,m} \to V_{3,1} \to V_{3,2} \to \cdots \to V_{3,m}.$$

Next, we define the edges of the graph:

- $V_{1,i}$ and $U_{1,i}$ are connected as an *upper triangle*. I.e., the j-th vertex of $V_{1,i}$ is connected to the k-th vertex of $U_{1,i}$ if and only in $k \geq j$.
- $V_{2,i}$ and the last βn vertices in $U_{1,i}$ are *fully connected*.
- V_2 and U_2 are *fully connected*.

[2] When $\beta e^\alpha n$ is a fraction, let there be $\lceil \beta e^\alpha n\rceil$ vertices. Since we are interested in the case when n, m are sufficiently large, we safely omit the ceiling function for the simplicity of notation. We apply a similar treatment for fractions throughout the paper.

- V_3 and U_2 are connected as a *upper triangle*. I.e., the j-th vertex of V_3 is connected to the k-th vertex of U_2 if and only in $k \geq j$.

We first calculate the optimum matching of the graph. That is, matching together (V_3, U_2) and $(V_1 \cup V_2, U_1)$. Therefore, we have

$$\mathrm{OPT} = \underbrace{nm \cdot f(\alpha)}_{(V_3,U_2)} + \underbrace{\sum_{i=1}^{m} \beta e^{\alpha} n \cdot f\left(\frac{i}{m} \cdot \beta\right)}_{(V_1 \cup V_2, U_1)} = nm \cdot \left(f(\alpha) + e^{\alpha} \int_0^{\beta} f(x) \mathrm{d}x + o(1)\right), \quad (3.5)$$

where the second equality holds when m goes to infinity.

Next, we analyze the the performance of Perturbed-Balance. We split the whole instance into three stages, corresponding to the arrivals of V_1, V_2, V_3 respectively.

First stage (V_1). Upon the arrival of each vertex in V_1, it matches uniformly to its neighbours in U_1. The behavior of Perturbed-Balance is the same for different groups. We analyze the matched portion of the last βn vertices of each group after the first stage:

$$x_u = \frac{1}{\beta e^{\alpha} n} + \frac{1}{\beta e^{\alpha} n - 1} + \cdots + \frac{1}{\beta n} = \ln\left(\frac{\beta e^{\alpha} n}{\beta n}\right) + o(1) = \alpha + o(1),$$

where the equality holds when n goes to infinity.

Second Stage (V_2). Upon the arrival of each vertex v of $V_{2,i}$, it will be weighing the perturbed weights from $U_{1,i}$ and U_2:

$$\begin{aligned} & f(x_{u_1}) \cdot w_{u_1} = f(\alpha + o(1)) \cdot f\left(\frac{i}{m}\right), && \text{for } u_1 \in U_{1,i} \cap N(v), \\ \text{and} \quad & f(x_{u_2}) \cdot w_{u_2} \geq f\left(\frac{i}{m}\right) \cdot f(\alpha), && \text{for } u_2 \in U_2. \end{aligned}$$

Notice that the perturbed weights from U_2 is always larger than the perturbed weights from $U_{1,i}$. We claim that in the second stage, all vertices of V_2 would be fully matched to U_2 by Perturbed-Balance. Thus, at the end of the second stage, all vertices in U_2 have matched portion β.

Third stage (V_3). The behavior of the last stage is similar to the behavior of the first stage, except that all vertices in U_2 start with a matched portion of β. After the arrival of the k-th vertex in V_3, the matched portion of its neighbor equals

$$\beta + \frac{1}{nm} + \frac{1}{nm-1} + \cdots + \frac{1}{nm-k+1} \geq \beta + \ln\left(\frac{nm}{nm-k+1}\right).$$

Consequently, only the first $(1 - e^{\beta-1})nm + 1$ vertices from V_3 can be matched. For the rest of the online vertices, all their neighbors would be fully matched before their arrivals.

To sum up, we calculate the performance of Perturbed-Balance.

$$\begin{aligned} \mathrm{ALG} &\leq \underbrace{\sum_{i=1}^{m} (\beta(e^{\alpha}-1)n + 1) \cdot f\left(\frac{i}{m} \cdot \beta\right)}_{(V_1,U_1)} + \underbrace{\beta nm \cdot f(\alpha)}_{(V_2,U_2)} + \underbrace{\left((1-e^{\beta-1})nm + 1\right) \cdot f(\alpha)}_{(V_3,U_2)} \\ &= nm \cdot \left((e^{\alpha}-1) \int_0^{\beta} f(x) \mathrm{d}x + (\beta + 1 - e^{\beta-1}) \cdot f(\alpha) + o(1)\right). \end{aligned} \quad (3.6)$$

Together with (3.5) and the assumption that Perturbed-Balance is Γ-competitive, we conclude the proof by letting $n, m \to \infty$:

$$(e^\alpha - 1)\int_0^\beta f(x)\mathrm{d}x + (\beta + 1 - e^{\beta-1}) \cdot f(\alpha) \geq \Gamma \cdot \left(f(\alpha) + e^\alpha \int_0^\beta f(x)\mathrm{d}x\right)$$
$$\iff \left(\beta + 1 - e^{\beta-1} - \Gamma\right) \cdot f(\alpha) \geq (1 - (1-\Gamma)\cdot e^\alpha) \cdot \int_0^\beta f(x)\mathrm{d}x, \quad \forall \alpha, \beta \in [0,1].$$

4 AdWords with Unknown Budget

In this section, we prove Theorem 2.

▶ **Theorem 2.** *The competitive ratio of* Perturbed-Ranking *algorithm with any perturbation function $f(x)$ on fractional AdWords is at most $1 - 1/e - 0.0003$. In particular, using the canonical function $f(x) = 1 - e^{x-1}$, the competitive ratio is at most 0.624.*

We first construct a hard instance for which Perturbed-Ranking with $f(x) = 1 - e^{x-1}$ only achieves a competitive ratio of 0.624. Recall that the vertex-weighted online bipartite matching problem is a special case of the AdWords problem. Together with Theorem 1, it should be convincing that Perturbed-Ranking (with any perturbation function) cannot achieve the prominent competitive ratio of $1 - 1/e$.

Our construction for general perturbation functions has a similar structure as the construction for the canonical perturbation function. On the other hand, general perturbation functions introduce extra technical difficulties to our argument that we shall discuss soon.

4.1 Canonical Perturbation Function $f(x) = 1 - e^{x-1}$

We prove the result by the following lemma.

▶ **Lemma 12.** *If* Perturbed-Ranking *with perturbation function $f(x) = 1 - e^{x-1}$ achieves a competitive ratio of Γ on AdWords, then*

$$(1-\Gamma)\cdot f(\alpha) \geq (\Gamma - \alpha) \cdot \int_0^\alpha f(x)\mathrm{d}x + \Gamma \cdot \int_\alpha^1 f(x)\mathrm{d}x, \quad \forall \alpha \in [0,1]. \tag{4.1}$$

Proof. Fix $\alpha \in [0,1]$. Let n be a sufficiently large number. Refer to Figure 4.1 for our instance.

Our construction consists of $n+1$ offline vertices $u_0, u_1, \ldots, u_n$ and $2n$ online vertices $v_1, v_2, \ldots, v_{2n}$. The budgets of $u_1, u_2, \ldots, u_n$ are all 1 and the budget of u_0 is unlimited. The online vertices arrive in ascending order of their indices, i.e. v_i is the i-th arriving online vertex. Next, we define the edges of the graph:

- $v_1, v_2, \ldots, v_n$ are connected to u_0, with edge weights $b_1, b_2, \ldots, b_n$ respectively.
- $v_1, v_2, \ldots, v_{2n}$ are fully connected to $u_1, u_2, \ldots, u_n$ with edge weights 1.

Before we define the weights, we make an extra assumption to simplify our analysis:

$$\forall 1 \leq i \leq n, \text{the rank of } u_i \text{ is } y_i = i/n.$$

Indeed, by Lemma 10, we would have that the set of ranks $\{y_1, \ldots, y_n\}$ are "close" to $\{\frac{1}{n}, \frac{2}{n}, \cdots, 1\}$. Moreover, all vertices $u_1, \ldots, u_n$ are symmetric in our graph. This assumption would significantly simplify our analysis and can be removed by a more conservative choice of the edge weights. We omit the tedious details for simplicity.

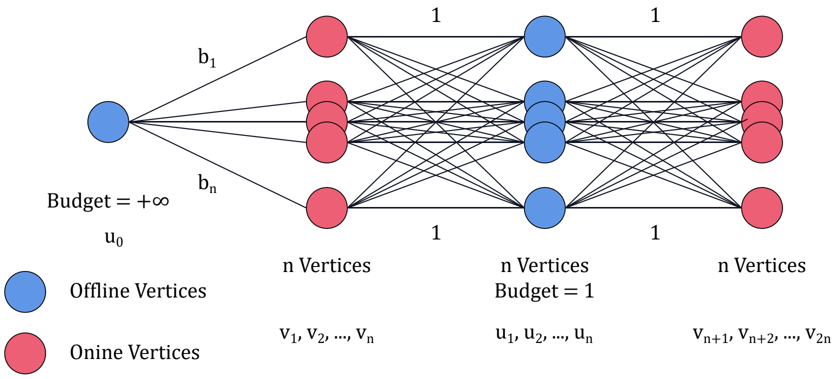

Figure 4.1 Instance 2.

Let $b_i = \frac{f(i/n)}{f(\alpha)}$. The offline optimum is to match $v_1, v_2, \ldots, v_n$ to u_0 and to match $v_{n+1}, v_{n+2}, \ldots, v_{2n}$ to $u_1, u_2, \ldots, u_n$, respectively. Consequently,

$$\mathrm{OPT} = n + \sum_{i=1}^{n} b_i = n \cdot \left(1 + \frac{1}{f(\alpha)} \int_0^1 f(x)\mathrm{d}x + o(1)\right).$$

With the assumption, the only randomness of Perturbed-Ranking is the rank y_0 of u_0.

Case 1. $(y_0 \geq \alpha)$ For each online vertex v_i, the perturbed weight of (u_0, v_i) is

$$b_i \cdot f(y_0) = \frac{f(y_0)}{f(\alpha)} \cdot f\left(\frac{i}{n}\right) \leq f\left(\frac{i}{n}\right),$$

while the perturbed weight of (u_i, v_i) is $f(y_i) = f\left(\frac{i}{n}\right)$. Therefore, Perturbed-Ranking matches (u_i, v_i) for all $1 \leq i \leq n$ and we have $\mathrm{ALG}(y_0) = n$.

Case 2. $(y_0 < \alpha)$ In this case, some of the $v_1, v_2, \ldots, v_n$ would be matched to u_0. However, the number of vertices matched to u_0 should be no more than αn. The reason is as follows. For each online vertex v_i, suppose the number of previous vertices matched to u_0 is larger than αn, then the perturbed weight of (u_0, v_i) is $\frac{f(y_0)}{f(\alpha)} \cdot f\left(\frac{i}{n}\right)$, while the perturbed weight of $(u_{i-\alpha n}, v_i)$ is $f\left(\frac{i}{n} - \alpha\right)$. Notice that $f(x) = 1 - e^{x-1}$ is a log-concave function. Hence,

$$\frac{f(y_0)}{f(\alpha)} \cdot f\left(\frac{i}{n}\right) \leq \frac{f(0)}{f(\alpha)} \cdot f\left(\frac{i}{n}\right) \leq f\left(\frac{i}{n} - \alpha\right).$$

In other words, v_i will not match u_0. This concludes the proof that the number of vertices matched to u_0 is no more than αn. Notice that b_i's are non-increasing, we have

$$\mathrm{ALG}(y_0) \leq \sum_{i=1}^{\alpha n} b_i + n = \sum_{i=1}^{\alpha n} \frac{f(i/n)}{f(\alpha)} + n = n \cdot \left(\frac{1}{f(\alpha)} \int_0^{\alpha} f(x)\mathrm{d}x + 1 + o(1)\right).$$

Putting the two cases together and assuming that Perturbed-Ranking algorithm is Γ-competitive, we conclude the proof of the lemma.

$$\Gamma \leq \frac{\mathbb{E}[\mathrm{ALG}]}{\mathrm{OPT}} = \frac{\alpha \cdot n \cdot \left(\frac{1}{f(\alpha)} \int_0^{\alpha} f(x)\mathrm{d}x + 1 + o(1)\right) + (1-\alpha) \cdot n}{n \cdot \left(1 + \frac{1}{f(\alpha)} \int_0^1 f(x)\mathrm{d}x + o(1)\right)} = \frac{\alpha \cdot \int_0^{\alpha} f(x)\mathrm{d}x + f(\alpha)}{\int_0^1 f(x)\mathrm{d}x + f(\alpha)}$$

$$\iff (1-\Gamma) \cdot f(\alpha) \geq (\Gamma - \alpha) \cdot \int_0^{\alpha} f(x)\mathrm{d}x + \Gamma \cdot \int_{\alpha}^{1} f(x)\mathrm{d}x. \qquad ◀$$

▶ **Corollary 13.** Perturbed-Ranking *with* $f(x) = 1 - e^{x-1}$ *is at most* 0.624*-competitive for AdWords.*

Proof. Plugging in $\alpha = 0.1$ and $f(x) = 1 - e^{x-1}$ in equation (4.1), we have

$$\Gamma \le \frac{\alpha \cdot \int_0^\alpha f(x)\mathrm{d}x + f(\alpha)}{\int_0^1 f(x)\mathrm{d}x + f(\alpha)} = \frac{0.1 \cdot \left(0.1 - e^{-0.9} + e^{-1}\right) + 1 - e^{-0.9}}{e^{-1} + 1 - e^{-0.9}} < 0.624 \ . \qquad \blacktriangleleft$$

4.2 General Perturbation Functions

Before we delve into the detailed proof, we explain the technical difficulty introduced by general perturbation functions. Our plan is to generalize Lemma 12: if we are able to prove equation (4.1) for an arbitrary function f, we would then conclude our theorem by combining it with equation (3.2).

However, our argument of the second case ($y_0 < \alpha$) of Lemma 12 crucially relies on the specific formula of $f(x)$. I.e., to upper bound the performance of Perturbed-Ranking, we use the fact that $f(0) \cdot f(x) \le f(\alpha) \cdot f(x - \alpha)$.

For a general perturbation function $f(x)$, if we stick to the same property that no more than αn vertices can be matched to u_0 when $y_0 < \alpha$, we could achieve it by setting the weights b_i to be smaller. Indeed, if $f\left(\frac{i}{n} - \alpha\right) \ge f(0) \cdot b_i$ holds, the previous analysis can be easily generalized. Hence, a natural attempt is to modify the instance as the following.

$$b_i = \begin{cases} \frac{1}{f(\alpha)} \cdot f\left(\frac{i}{n}\right) & i < \alpha n, \\ \min\left\{\frac{1}{f(\alpha)} \cdot f\left(\frac{i}{n}\right), \frac{1}{f(0)} \cdot f\left(\frac{i}{n} - \alpha\right)\right\} & i \ge \alpha n. \end{cases}$$

Unfortunately, this modification is not strong enough to give a constant strictly smaller than $1 - 1/e$. The reason is that the function f may have a steep drop in the neighborhood of 0, which leads to negligible b_i's in the above construction.

On the other hand, the failure of the analysis comes from our coarse and brutal relaxation by establishing a single upper bound for all $y_0 < \alpha$. For instance, if the function steeply drops at some $\beta \in [0, \alpha]$, then we could resolve the issue by considering two cases of $y_0 < \beta$ or $y_0 \in [\beta, \alpha)$. Formally, we prove the following lemma that is slightly weaker than (4.1).

▶ **Lemma 14.** *If* Perturbed-Ranking *with perturbation function* $f(x)$ *achieves a competitive ratio of* Γ *on AdWords, then*

$$(1 - \Gamma) \cdot f(\alpha) \ge (\Gamma - \alpha) \cdot \int_0^\alpha f(x)\mathrm{d}x + (\Gamma - \beta) \cdot \int_\alpha^1 \min\left\{f(x), \frac{f(\alpha)}{f(\beta)} f(x - \alpha)\right\}\mathrm{d}x,$$
$$\forall \alpha, \beta \in [0, 1], \alpha \ge \beta. \quad (4.2)$$

Proof. We apply the same construction as in Lemma 12 and make the same assumption that $y_i = \frac{i}{n}$ for the simplicity of presentation. We modify the instance by setting the weights b_i differently:

$$\forall 1 \le i \le n, \quad b_i = \begin{cases} \frac{1}{f(\alpha)} \cdot f\left(\frac{i}{n}\right) & i < \alpha n, \\ \min\left\{\frac{1}{f(\alpha)} \cdot f\left(\frac{i}{n}\right), \frac{1}{f(\beta)} \cdot f\left(\frac{i}{n} - \alpha\right)\right\} & i \ge \alpha n. \end{cases}$$

The optimal solution equals

$$\mathrm{OPT} = n + \sum_{i=1}^n b_i = n \cdot \left(1 + \frac{1}{f(\alpha)} \int_0^\alpha f(x)\mathrm{d}x + \int_\alpha^1 \min\left\{\frac{f(x)}{f(\alpha)}, \frac{f(x - \alpha)}{f(\beta)}\right\}\mathrm{d}x + o(1)\right). \quad (4.3)$$

Next, we consider the performance of Perturbed-Ranking depending on the value of y_0.

Case 1. ($y_0 < \beta$) We use OPT as a trivial upper bound of ALG, i.e., $\mathrm{ALG}(y_0) \le \mathrm{OPT}$.

Case 2. ($y_0 \ge \alpha$) For each online vertex v_i, the perturbed weight of (u_0, v_i) is

$$b_i \cdot f(y_0) \le \frac{f(y_0)}{f(\alpha)} \cdot f\left(\frac{i}{n}\right) \le f\left(\frac{i}{n}\right),$$

while the perturbed weight of (u_i, v_i) is $f(y_i) = f\left(\frac{i}{n}\right)$. Therefore, Perturbed-Ranking matches (u_i, v_i) for all $1 \le i \le n$ and we have $\mathrm{ALG}(y_0) = n$.

Case 3. ($y_0 \in [\beta, \alpha)$) We prove that the number of vertices matched to u_0 is no more than αn. This can be similarly argued as follows. For each online vertex i ($i > \alpha n$), suppose the number of vertices matched to u_0 is already αn, the perturbed weight for u_0 is $f(y_0) \cdot b_i \le \frac{f(y_0)}{f(\beta)} \cdot f\left(\frac{i}{n} - \alpha\right)$, while the perturbed weight for $u_{i-\alpha n}$ is $f\left(\frac{i}{n} - \alpha\right)$. So that $f\left(\frac{i}{n} - \alpha\right) \ge f(y_0) \cdot b_i$ and thus v_i will not choose u_0 again. Therefore, as the number of vertices matched to u_0 is no more than αn, we have

$$\mathrm{ALG} \le n + \sum_{i=1}^{\alpha n} b_i = n \cdot \left(1 + \frac{1}{f(\alpha)} \int_0^\alpha f(x)\mathrm{d}x + o(1)\right).$$

Taking expectation over the randomness of y_0, we conclude that

$$\mathbb{E}[\mathrm{ALG}] \le n \cdot \left(\frac{\alpha - \beta}{f(\alpha)} \int_0^\alpha f(x)\mathrm{d}x + 1 - \beta + o(1)\right) + \beta \cdot \mathrm{OPT}. \tag{4.4}$$

Finally, we conclude the proof by plugging in (4.3) and (4.4) to $\mathbb{E}[\mathrm{ALG}] \ge \Gamma \cdot \mathrm{OPT}$ and letting n goes to infinity. ◀

Recall that the vertex-weighted online bipartite matching problem is a special case of AdWords. We conclude the proof of Theorem 2 by Lemma 11, 14 and the following mathematical fact. We provide the proof of the following lemma in the full version.

▶ **Lemma 15.** *If a perturbation function f and $\Gamma > 0$ satisfy the following conditions:*

$$\left(\alpha + 1 - e^{\alpha-1} - \Gamma\right) \cdot f(\beta) \ge \left(1 - (1-\Gamma) \cdot e^{\beta}\right) \cdot \int_0^\alpha f(x)\mathrm{d}x, \quad \forall \alpha, \beta \in [0,1], \tag{4.5}$$

$$(1-\Gamma) f(\alpha) \ge (\Gamma - \alpha) \int_0^\alpha f(x)\mathrm{d}x + (\Gamma - \beta) \int_\alpha^1 \min\left\{f(x), \frac{f(\alpha)}{f(\beta)} f(x-\alpha)\right\}\mathrm{d}x, \quad \forall \alpha, \beta \in [0,1], \alpha \ge \beta, \tag{4.6}$$

then $\Gamma < 1 - 1/e - 0.0003$.

References

1 Gagan Aggarwal, Gagan Goel, Chinmay Karande, and Aranyak Mehta. Online vertex-weighted bipartite matching and single-bid budgeted allocations. In *SODA*, pages 1253–1264, 2011. URL: `http://www.siam.org/proceedings/soda/2011/SODA11_094_aggarwalg.pdf`.

2 Benjamin Birnbaum and Claire Mathieu. On-line bipartite matching made simple. *ACM SIGACT News*, 39(1):80–87, 2008.

3 Guy Blanc and Moses Charikar. Multiway online correlated selection. In *62nd IEEE Annual Symposium on Foundations of Computer Science, FOCS 2021, Denver, CO, USA, February 7-10, 2022*, pages 1277–1284. IEEE, 2021. `doi:10.1109/FOCS52979.2021.00124`.

4 Niv Buchbinder, Moran Feldman, Yuval Filmus, and Mohit Garg. Online submodular maximization: beating 1/2 made simple. *Math. Program.*, 183(1):149–169, 2020. doi:10.1007/s10107-019-01459-z.

5 T.-H. Hubert Chan, Fei Chen, Xiaowei Wu, and Zhichao Zhao. Ranking on arbitrary graphs: Rematch via continuous linear programming. *SIAM J. Comput.*, 47(4):1529–1546, 2018.

6 Nikhil R. Devanur and Thomas P. Hayes. The adwords problem: online keyword matching with budgeted bidders under random permutations. In John Chuang, Lance Fortnow, and Pearl Pu, editors, *Proceedings 10th ACM Conference on Electronic Commerce (EC-2009), Stanford, California, USA, July 6–10, 2009*, pages 71–78. ACM, 2009. doi:10.1145/1566374.1566384.

7 Nikhil R. Devanur, Kamal Jain, and Robert D. Kleinberg. Randomized primal-dual analysis of RANKING for online bipartite matching. In *SODA*, pages 101–107. SIAM, 2013.

8 Alon Eden, Michal Feldman, Amos Fiat, and Kineret Segal. An economics-based analysis of RANKING for online bipartite matching. In *SOSA*, pages 107–110. SIAM, 2021.

9 Matthew Fahrbach, Zhiyi Huang, Runzhou Tao, and Morteza Zadimoghaddam. Edge-weighted online bipartite matching. In *FOCS*, pages 412–423. IEEE, 2020.

10 Ruiquan Gao, Zhongtian He, Zhiyi Huang, Zipei Nie, Bijun Yuan, and Yan Zhong. Improved online correlated selection. In *62nd IEEE Annual Symposium on Foundations of Computer Science, FOCS 2021, Denver, CO, USA, February 7-10, 2022*, pages 1265–1276. IEEE, 2021. doi:10.1109/FOCS52979.2021.00123.

11 Gagan Goel and Aranyak Mehta. Online budgeted matching in random input models with applications to adwords. In *SODA*, pages 982–991, 2008. URL: http://dl.acm.org/citation.cfm?id=1347082.1347189.

12 Zhiyi Huang, Ning Kang, Zhihao Gavin Tang, Xiaowei Wu, Yuhao Zhang, and Xue Zhu. Fully online matching. *J. ACM*, 67(3):17:1–17:25, 2020.

13 Zhiyi Huang, Binghui Peng, Zhihao Gavin Tang, Runzhou Tao, Xiaowei Wu, and Yuhao Zhang. Tight competitive ratios of classic matching algorithms in the fully online model. In *SODA*, pages 2875–2886. SIAM, 2019.

14 Zhiyi Huang and Xinkai Shu. Online stochastic matching, poisson arrivals, and the natural linear program. In Samir Khuller and Virginia Vassilevska Williams, editors, *STOC '21: 53rd Annual ACM SIGACT Symposium on Theory of Computing, Virtual Event, Italy, June 21-25, 2021*, pages 682–693. ACM, 2021. doi:10.1145/3406325.3451079.

15 Zhiyi Huang, Zhihao Gavin Tang, Xiaowei Wu, and Yuhao Zhang. Online vertex-weighted bipartite matching: Beating 1-1/e with random arrivals. *ACM Transactions on Algorithms*, 15(3):1–15, 2019.

16 Zhiyi Huang, Zhihao Gavin Tang, Xiaowei Wu, and Yuhao Zhang. Fully online matching II: beating ranking and water-filling. In *FOCS*, pages 1380–1391. IEEE, 2020.

17 Zhiyi Huang, Qiankun Zhang, and Yuhao Zhang. Adwords in a panorama. In *FOCS*, pages 1416–1426. IEEE, 2020.

18 Billy Jin and David P. Williamson. Improved analysis of RANKING for online vertex-weighted bipartite matching in the random order model. In Michal Feldman, Hu Fu, and Inbal Talgam-Cohen, editors, *Web and Internet Economics - 17th International Conference, WINE 2021, Potsdam, Germany, December 14-17, 2021, Proceedings*, volume 13112 of *Lecture Notes in Computer Science*, pages 207–225. Springer, 2021. doi:10.1007/978-3-030-94676-0_12.

19 Bala Kalyanasundaram and Kirk Pruhs. An optimal deterministic algorithm for online b-matching. *Theoretical Computer Science*, 233(1-2):319–325, 2000.

20 Michael Kapralov, Ian Post, and Jan Vondrák. Online submodular welfare maximization: Greedy is optimal. In Sanjeev Khanna, editor, *Proceedings of the Twenty-Fourth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2013, New Orleans, Louisiana, USA, January 6-8, 2013*, pages 1216–1225. SIAM, 2013. doi:10.1137/1.9781611973105.88.

21 Chinmay Karande, Aranyak Mehta, and Pushkar Tripathi. Online bipartite matching with unknown distributions. In *STOC*, pages 587–596, 2011. doi:10.1145/1993636.1993715.

22 Richard M. Karp, Umesh V. Vazirani, and Vijay V. Vazirani. An optimal algorithm for on-line bipartite matching. In *STOC*, pages 352–358, 1990. doi:10.1145/100216.100262.

23 Nitish Korula, Vahab S. Mirrokni, and Morteza Zadimoghaddam. Online submodular welfare maximization: Greedy beats 1/2 in random order. *SIAM J. Comput.*, 47(3):1056–1086, 2018. doi:10.1137/15M1051142.

24 Mohammad Mahdian and Qiqi Yan. Online bipartite matching with random arrivals: an approach based on strongly factor-revealing LPs. In *STOC*, pages 597–606, 2011. doi:10.1145/1993636.1993716.

25 Aranyak Mehta, Amin Saberi, Umesh V. Vazirani, and Vijay V. Vazirani. Adwords and generalized online matching. *J. ACM*, 54(5):22, 2007.

26 Zhihao Gavin Tang, Xiaowei Wu, and Yuhao Zhang. Towards a better understanding of randomized greedy matching. In Konstantin Makarychev, Yury Makarychev, Madhur Tulsiani, Gautam Kamath, and Julia Chuzhoy, editors, *Proccedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing, STOC 2020, Chicago, IL, USA, June 22-26, 2020*, pages 1097–1110. ACM, 2020. doi:10.1145/3357713.3384265.

27 Rajan Udwani. Adwords with unknown budgets. *CoRR*, abs/2110.00504, 2021. arXiv:2110.00504.

28 Vijay V. Vazirani. Online bipartite matching and adwords. *CoRR*, abs/2107.10777, 2021. arXiv:2107.10777.