Fully Online Matching II: Beating Ranking and Water-filling

Zhiyi Huang*, Zhihao Gavin Tang†, Xiaowei Wu‡ and Yuhao Zhang*
*The University of Hong Kong
{zhiyi, yhzhang2}@cs.hku.hk
†Shanghai University of Finance and Economics
tang.zhihao@mail.shufe.edu.cn
‡University of Macau
xiaoweiwu@um.edu.mo

***Abstract*— Karp, Vazirani, and Vazirani (STOC 1990) initiated the study of online bipartite matching, which has held a central role in online algorithms ever since. Of particular importance are the Ranking algorithm for integral matching and the Water-filling algorithm for fractional matching. Most algorithms in the literature can be viewed as adaptations of these two in the corresponding models. Recently, Huang et al. (SODA 2019, JACM 2020) introduced a more general model called *fully online matching*, which considers general graphs and allows all vertices to arrive online. They also generalized Ranking and Water-filling to fully online matching and gave some tight analysis: Ranking is $\Omega \approx 0.567$-competitive on bipartite graphs where the Ω-constant satisfies $\Omega e^{\Omega} = 1$, and Water-filling is $2-\sqrt{2} \approx 0.585$-competitive on general graphs.**

We propose fully online matching algorithms strictly better than Ranking and Water-filling. For integral matching on bipartite graphs, we build on the online primal dual analysis of Ranking and Water-filling to design a 0.569-competitive hybrid algorithm called Balanced Ranking. To our knowledge, it is the first integral algorithm in the online matching literature that successfully integrates ideas from Water-filling. For fractional matching on general graphs, we give a 0.592-competitive algorithm called Eager Water-filling, which may match a vertex on its arrival. By contrast, the original Water-filling algorithm always matches vertices at their deadlines. Our result for fractional matching further shows a separation between fully online matching and the general vertex arrival model by Wang and Wong (ICALP 2015), due to an upper bound of 0.5914 in the latter model by Buchbinder, Segev, and Tkach (ESA 2017).

***Keywords*-Online Matching; Ranking; Water-filling**

I. Introduction

Online matching is one of the oldest and most fruitful topic in the online algorithms literature. It dates back to thirty years ago when Karp et al. [2] proposed the online bipartite matching problem and the Ranking algorithm. Consider a bipartite graph, where the left-hand-side vertices are offline, i.e., known upfront to the algorithm, and the right-hand-side vertices are online arriving one at a time. On the arrival of an online vertex, the algorithm observes its incident edges and must immediately and irrevocably decide how to match it. The goal is to maximize the cardinality of the matching. For a real-world example, think of online advertising where the offline and online vertices correspond to advertisers and impressions respectively. Ranking picks a random permutation of the offline vertices, and matches each online vertex to the first unmatched neighbor by the permutation. Karp et al. [2] showed that it is $1-\frac{1}{e}$-competitive, and this is the best possible for the problem.

In online advertising, an advertiser can usually be matched to many impressions. This is the b-matching model of Kalyanasundaram and Pruhs [3] where b is the number of times an offline vertex can be matched, a.k.a., its capacity. This is also closely related to the fractional relaxation of online bipartite matching where each online vertex may be matched fractionally to multiple offline neighbors so long as the total matched amount does not exceed one unit.[1] In this case, the optimal $1-\frac{1}{e}$ competitive ratio can be achieved with a deterministic algorithm called Water-filling (a.k.a. Water-level or Balance). It matches each online vertex continuously to the least matched offline neighbors.

Ranking and Water-filling are the most fundamental algorithms in online bipartite matching and its variants. Most algorithms in the online matching literature under worst-case analysis can be viewed as adaptations of them in the corresponding models.

Full version of this paper is available at arxiv.org [1]. Zhiyi Huang is supported by the Hong Kong Research Grants Council (RGC) under Grant No. 17202019E. Zhihao Gavin Tang is supported by National Natural Science Foundation of China (NSFC) under Grant No. 61902233. Xiaowei Wu is supported by The Science and Technology Development Fund, Macau SAR (File No. SKL-IOTSC-2018-2020) and the Start-up Research Grant of University of Macau (File No. SRG2020-00020-IOTSC).

[1] The fractional problem is equivalent to a b-matching problem in which online vertices arrive in batches of b copies and b tends to infinity. Further, the assumption of having b copies per online vertex is irrelevant in existing analysis.

2575-8454/20/$31.00 ©2020 IEEE
DOI 10.1109/FOCS46700.2020.00130

Fully Online Matching.: Let us turn to a different real-world scenario which involves a bipartite matching problem with an online flavor. Consider a ride-hailing platform that matches drivers on one side and passengers on the other side. This is *not* captured by the model of Karp et al. [2] because vertices on both sides of the bipartite graph arrive and depart online. To this end, Huang et al. [4, 5, 6] recently introduced a generalized model called *fully online matching*. Each vertex in the fully online model is associated with not only an arrival time but also a deadline. Each edge is revealed to the algorithm when both endpoints have arrived, and can be selected into the matching anytime before the endpoints' deadlines provided that they are still unmatched. Furthermore, the model extends naturally to general graphs, capturing an even broader class of problems including matching passengers in ride-sharing.

Huang et al. [4, 5, 6] generalized both Ranking and Water-filling to fully online matching. Both algorithms only match vertices at their deadlines. Ranking selects a random permutation of *all vertices*;[2] then, for any vertex that stays unmatched till its deadline, Ranking matches it to the first available neighbor by the permutation. Similarly, for any vertex that is not fully matched by its deadline, Water-filling matches the remaining portion fractionally to the least-matched available neighbors. They showed that Ranking is 0.521-competitive on general graphs. For bipartite graphs, they gave a tight analysis showing that Ranking is $\Omega \approx 0.567$-competitive, where the Ω-constant is the solution of $\Omega \cdot e^{\Omega} = 1$. Further, they proved a tight $2 - \sqrt{2} \approx 0.585$ competitive ratio of Water-filling on general graphs. Finally, they separated fully online matching with online bipartite matching of Karp et al. [2] by showing that there is no $1 - \frac{1}{e}$-competitive algorithm in the fully online model.

A. Our Contributions

This work is driven by a natural question: *Are Ranking and Water-filling optimal in fully online matching, like in many other online matching models?* In particular, is the $\Omega \approx 0.567$ competitive ratio the best possible for integral algorithms on bipartite graphs? How about the $2 - \sqrt{2} \approx 0.585$ competitive ratio for fractional algorithms on general graphs? Surprisingly, the answers are no! There are algorithms strictly better than Ranking and Water-filling in fully online matching!

Beating Ranking on Bipartite Graphs.: We follow a simple intuition: since Water-filling has a superior competitive ratio, we may "correct" the decisions by Ranking with those by Water-filling. While easy to state, this intuition is difficult to substantiate. In fact, to our knowledge, there is no integral algorithm in the online matching literature prior to our work which successfully integrates ideas from Water-filling. To explain our algorithm, we need the following equivalent interpretations of Ranking and Water-filling from the randomized primal dual technique (see, e.g., Devanur et al. [7]).

- **Ranking:** Draw a random number $y_u \in [0, 1]$ for each vertex u. Then, if v is matched to vertex u at u's deadline, they split one unit of gain. Vertex v keeps $g(y_v) = e^{y_v - 1}$ to itself, and offers $1 - g(y_v)$ to u. For each vertex u which stays unmatched till its deadline, it matches to the offline neighbor who offers the most, i.e., the one with the smallest y_v.
- **Water-filling:** For each vertex u, let $x_u \in [0, 1]$ denote its matched portion, a.k.a. its water level. Then, if v is matched to vertex u at u's deadline by some infinitesimal amount ϵ, they split the gain of ϵ. Vertex v keeps $f(x_v) = e^{x_v - 1}$ times ϵ to itself, and offers $\big(1 - f(x_v)\big)\epsilon$ to u. For each vertex u which is not fully matched by its deadline, it fractionally matches to the least matched offline neighbors to maximize the total offer.

For any nondecreasing f and g, we define a hybrid algorithm called Balance Ranking as follows.

- **Balanced Ranking:** For each vertex u, draw a random number $y_u \in [0, 1]$. Let x_u be the probability that u is matched, a.k.a. its water level. Then, if v is matched to another vertex u at u's deadline, they split one unit of gain. Vertex v keeps $f(x_v) + g(y_v)$ to itself, and offers $1 - f(x_v) - g(y_v)$ to u, *where x_v is v's water level after u's deadline*. For each vertex u which stays unmatched till its deadline, it matches to the offline neighbor who offers the most.

While the algorithm is a simple combination of the alternative interpretations of Ranking and Water-filling, it is crucial to match each vertex based on the water levels of the neighbors *after* the matching decision of the current vertex. We show in Section III-A that it is a well-defined algorithm.

Then, we analyze Balanced Ranking under the online primal dual framework and design the functions f and g by solving a differential equation arising from the analysis. See Section III-C.

Theorem I.1. *Balanced Ranking is 0.569-competitive for fully online matching on bipartite graphs.*

Beating Water-filling.: We start with an observation that Water-filling works in an even harder model, where an edge is revealed to the algorithm only when an endpoint reaches the deadline. In fact, the hardness result by Huang et al. [5] implies that Water-filling is

[2] For example, draw a random number in $[0, 1)$ on the arrival of each vertex and sort them by the numbers.

optimal in the harder model. The observation suggests, however, Water-filling gives up the information about the edges among the vertices which have arrived but have not yet reached the deadlines. Intuitively, we shall be able to improve the competitive ratio by taking such information into account. Our algorithm utilizes the information implicitly by eagerly matching vertices partially on their arrivals. Indeed, the eager matches are precisely among vertices that have arrived but have not reached the deadlines. For any nondecreasing function f, define the Eager Water-filling algorithm as follows.

- **Eager Water-filling:** On the arrival of each vertex u, match it fractionally to the least matched offline neighbors v as long as v's offer is larger than what u wants for itself at the current water level, i.e., $1-f(x_v) \geq f(x_u)$. If a vertex u is not fully matched by its deadline, match it fractionally to the least matched offline neighbors to maximize the total offer.

We can naturally interpret the algorithm as having vertex u make decisions *assuming the graph stays as it is*. Suppose some neighbor v satisfies $1-f(x_v) \geq f(x_u)$ on u's arrival. On the one hand, an eager match with v offers $1-f(x_v)$ to u per unit of match. On the other hand, if u opts to wait, it risks getting matched at some other vertex's deadline in which case u keeps only $f(x_u)$ per unit of match, inferior to an eager match with v. Further, v may be matched by some other vertex while u waits. Finally, even if none of these happens, u at best has the same options at its deadline compared to the eager matches on its arrival, assuming the graph stays the same. In sum, u shall fractionally match to v on its arrival as in Eager Water-filling. We remark that in principle, we only have to match an edge when at least one endpoint reaches its deadline and our description of the algorithm is for the purpose of an intuitive interpretation. Indeed, we can reserve any edge and delay its matching decision until one endpoint reaches its deadline.

Again, we analyze Eager Water-filling under the primal dual framework and design the function f by solving a differential equation arising from the analysis. See Section IV.

Theorem I.2. *Eager Water-filling is 0.592-competitive for fractional fully online matching.*

This result separates fully online matching from another model called general vertex arrival by Wang and Wong [8] because of a 0.5914 upper bound on the best possible competitive ratio in the latter model by Buchbinder et al. [9]. Intriguingly, general vertex arrival is essentially fully online matching *restricted to eager matches only*. In other words, we obtain the separation by forfeiting part of the flexibility to defer decisions till the deadlines, and by incorporating eager matches which are allowed in the general vertex arrival model in the first place.

B. Other Related Works

The analysis of Ranking in online bipartite matching has been refined and simplified in a series of papers by Goel and Mehta [10], Birnbaum and Mathieu [11], Devanur et al. [7], Eden et al. [12], and Feige [13] . In particular, the online primal dual framework by Devanur et al. [7] has been the backbone of the competitive analysis in fully online matching including those in this paper.

Many variants of online bipartite matching have been introduced. Mehta et al. [14] proposed the first generalization called AdWords motivated by online advertising, which has been simplified and generalized under online primal dual [15, 16]. Aggarwal et al. [17] considered the vertex-weighted problem and extended Ranking to this model. Feldman et al. [18] investigated the fractional edge-weighted case. Their algorithm can be seen as an adaptation of Water-filling, and the analysis was simplified by Devanur et al. [19]. Mehta and Panigrahi [20] introduced a model with stochastic rewards and the results were later improved by Mehta et al. [21] and Huang and Zhang [22]. Some of the models have also been studied under the assumption of a random arrival order [23, 24, 25].

Besides fully online matching and general vertex arrival, there is an even harder edge arrival model. Recently, Gamlath et al. [26] proved that the trivial 0.5-competitive greedy algorithm is the best possible. They also obtained the first integral algorithm that breaks the 0.5 barrier in the general vertex arrival model. Prior to that, there were some positive results for special cases of edge arrival, e.g., when the graph is a forest [9]. The fully online matching problem is closely related to the online windowed matching problem by Ashlagi et al. [27], which can be viewed as an edge-weighted version of fully online matching under the first-in-first-out assumption.

II. Preliminaries

Model.: Consider an undirected graph $G=(V,E)$. Initially, the algorithm has no information about G. Then, we proceed in $2|V|$ steps, each of which is one of the following two kinds:

- *Arrival of a vertex u:* The algorithm observes the edges between u and the previously arrived vertices. This is the earliest step when u can be matched.
- *Deadline of a vertex u:* This is the last step when u can be matched. We guarantee that all neighbors

of u arrive before u's deadline.[3]

The goal is to maximize the size of the matching. Following the standard competitive analysis, an algorithm is Γ-competitive for some $0 \le \Gamma \le 1$, if for any fully online matching instance, the expected size of its matching is at least Γ times the optimal matching in hindsight.

Observe that fully online matching generalizes the model of Karp et al. [2], because the latter can be seen as having the offline vertices arrive at the beginning and leave at the end, and letting the deadline of each online vertex be right after its arrival.

Integral vs. Fractional Algorithms.: An integral algorithm must match each vertex u in whole to another vertex, although the matching decisions could be randomized. A fractional algorithm, however, may match a vertex u fractionally to multiple vertices, e.g., $\frac{1}{2}$ to v_1, $\frac{1}{4}$ to v_2, and another $\frac{1}{4}$ to v_3, as long as the total amount is at most 1.

Matching LP.: For any edge $(u, v) \in E$, let x_{uv} be the probability/fraction that edge (u, v) is matched by the algorithm. Consider the following standard matching LP and its dual:

$$\begin{aligned} \max: \quad & \textstyle\sum_{(u,v)\in E} x_{uv} & \\ \text{s.t.} \quad & \textstyle\sum_{v:(u,v)\in E} x_{uv} \le 1 & \forall u \in V \\ & x_{uv} \ge 0 & \forall (u,v) \in E \\ \min: \quad & \textstyle\sum_{u\in V} \alpha_u & \\ \text{s.t.} \quad & \alpha_u + \alpha_v \ge 1 & \forall (u,v) \in E \\ & \alpha_u \ge 0 & \forall u \in V \end{aligned}$$

Let P and D denote the primal and dual objectives respectively. Observe that by the above choice of x_{uv}'s, P also equals the expected size of the algorithm's matching.

Randomized Online Primal Dual Framework.: An online primal dual algorithm maintains not only a matching but also a dual assignment online.

Lemma II.1 ([7]). *An online primal dual algorithm is Γ-competitive if we have:*

- Approximate dual feasibility in expectation: $\forall (u, v) \in E,\ \mathbb{E}[\alpha_u] + \mathbb{E}[\alpha_v] \ge \Gamma$;
- Reverse weak duality in expectation: $P \ge \mathbb{E}[D]$.

The algorithms in this paper will satisfy reverse weak duality in expectation with equality. This is because whenever an edge (u, v) is matched by our algorithms, the increment in matching size is split between the dual variables α_u and α_v of the two endpoints.

[3]Consider the ride-hailing example. The guarantee effectively means that, for instance, a driver on a day shift cannot be matched with a passenger in the evening.

III. Balanced Ranking

This section presents the Balanced Ranking algorithm, which is a hybrid algorithm building on both Ranking and Water-filling for fully online matching on bipartite graphs, and prove Theorem I.1.

A. Matching with Ranks and Lookahead Water Levels

Recall the primal dual interpretations of Ranking and Water-filling as follows. Ranking fixes a nondecreasing function g, and draws a random rank $y_u \in [0, 1]$ for each vertex u. At the deadline of each vertex u, if u is not matched yet the algorithm matches it to its neighbor v with the largest offer $1 - g(y_v)$. Water-filling maintains the matched fraction $x_u \in [0, 1]$ of each vertex u, a.k.a. its water level, and matches u fractionally to the neighbors with the largest offers $1 - f(x_v)$ per unit of match for some nondecreasing function f. Hence, a natural hybrid algorithm is to define the offer of each vertex v to be $1 - g(y_v) - f(x_v)$, and to match the neighbor with the largest offer. Here, the water level x_v in a randomized integral algorithm is the probability that v is matched; this is equivalent to the probability that it is *passive*, i.e. it is matched before its deadline.

Lookahead Water Levels.: Observe, however, that the water levels change over time. Therefore, we need to further elaborate at what time we evaluate the water levels x_v's in the hybrid algorithm. Suppose we are to match a vertex u which stays unmatched by its deadline. The first instinct may be to use the current water levels right *before* the deadline of u. Surprisingly, the attempt fails according to our analysis. We instead consider the water levels right *after* the deadline of u, which we call the *lookahead water levels*. Intuitively, balancing the lookahead water levels keeps as many options available as possible to hedge against all future possibilities.

To avoid confusion, we use $x_v^{(u)}$ to denote v's water level right after u's deadline. It is exactly equal to the probability that v is passive after u's deadline.

Computing Lookahead Water Levels.: The algorithm is still incomplete as it involves circular definitions. The lookahead water levels $x_v^{(u)}$'s depend on the matching decision at u's deadline, which is made based on the lookahead water levels $x_v^{(u)}$'s. Next we argue this is not only well defined but further efficiently computable up to high accuracy. Assuming:

- f is 1-Lipschitz i.e., $f(x) - f(y) \le x - y$ for any $x \ge y \in [0, 1]$;
- g is $\frac{1}{100}$-reverse Lipschitz, i.e., $g(x) - g(y) \ge \frac{x-y}{100}$ for any $x \ge y \in [0, 1]$.

The constants 1 and $\frac{1}{100}$ are arbitrary so long as the former is not too small and the latter is not too large. They are only for the convenience in the definition of the algorithm and are not binding constraints in our analysis. Even if not stated explicitly, there is some optimal choice of f and g in our competitive analysis with the above Lipschitz and reverse Lipschitz properties.

Lemma III.1. *Suppose the algorithm is well defined before u's deadline. In* $\mathsf{poly}(|V|, \frac{1}{\epsilon})$ *time, we can compute* $\widehat{f}_v \in [0,1]$ *for all* $v \in V$*, such that whenever* u *stays unmatched by its deadline, matching* u *to vertex* v *with the largest* $1 - g(y_v) - \widehat{f}_v$ *leads to water levels* $x_v^{(u)}$*'s with* $\widehat{f}_v - \epsilon \le f(x_v^{(u)}) \le \widehat{f}_v$*.*

Proof: We start with the trivial overestimates $\widehat{f}_v = 1$ for all $v \in V$. Then, we iteratively refine them while keeping the invariant that they are overestimates. That is, whenever u stays unmatched by its deadline, matching u to the vertex v with the largest $1-g(y_v)-\widehat{f}_v$ leads to water levels $x_v^{(u)}$'s such that $f(x_v^{(u)}) \le \widehat{f}_v$. Finally, we bound the time complexity by proving that the sum of the estimates, i.e., $\sum_{v\in V} \widehat{f}_v$, decreases at least linearly.

Concretely, whenever there is a vertex v with $\widehat{f}_v - f(x_v^{(u)}) > \epsilon$, decrease the estimate $\widehat{f}_v$ by $\frac{\epsilon}{101}$. Here, we can compute $x_v^{(u)}$ up to high enough accuracy from sample runs of the algorithm by standard concentration bounds. In doing so, the water level $x_v^{(u)}$ increases and the water levels $x_w^{(u)}$ for all $w \notin \{u, v\}$ weakly decreases.

We first argue that the invariant still holds. It suffices to consider v because for any other vertex the water level weakly decreases and the estimate stays the same. Next we show that the water level of v increases by at most $\frac{100\epsilon}{101}$ and thus, maintains the invariant. Equivalently, we claim that the probability v is matched at u's deadline increases by at most $\frac{100\epsilon}{101}$, which is true even conditioned on the ranks $\vec{y}_{\text{-}v}$ of the other vertices. By decreasing $\widehat{f}_v$ by $\frac{\epsilon}{101}$, the threshold $g(y_v)$ above which v is picked by u, increases by the same amount. This in turn increases the threshold rank y_v by at most $\frac{100\epsilon}{101}$ because g is $\frac{1}{100}$-reverse Lipschitz. Since y_v is uniform from $[0,1]$, we conclude that the probability that v is matched at u's deadline increases by at most $\frac{100\epsilon}{101}$, conditioned on *any* $\vec{y}_{\text{-}v}$.

Finally, the algorithm terminates in $O(\frac{|V|}{\epsilon})$ iterations, because the sum of the estimates, i.e., $\sum_{v\in V} \widehat{f}_v$, decreases by at at least $\Omega(\epsilon)$ per iteration and it is between 0 and $|V|$. ■

Our analysis degrades gracefully in the error term ϵ in the above lemma. For simplicity, the rest of the section assumes the limit case when $f(x_v^{(u)}) = \widehat{f}_v$. See Algorithm 1.

Algorithm 1 Balanced Ranking, with Dual Assignments

at u**'s arrival:**

draw rank $y_u \in [0,1]$ uniformly at random

at u**'s deadline, if it is unmatched:**

compute the lookahead water levels $x_v^{(u)}$ for all $v \in V$ (Lemma III.1)

match u to the unmatched neighbor v with the largest $1 - g(y_v) - f(x_v^{(u)})$

let $\alpha_u = 1 - g(y_v) - f(x_v^{(u)})$ and $\alpha_v = g(y_v) + f(x_v^{(u)})$

B. Notations and Basic Properties

Our analysis of Balanced Ranking builds on the approach of Huang et al. [4, 5, 6]. This section adopts some notations from their analysis, and establishes several basic properties of Ranking that continue to hold for Balanced Ranking.

In the following, for any instance G and any realization of ranks $\vec{y}$, let $M_G(\vec{y})$ be the matching produced by Balanced Ranking. Let $G_{\text{-}u}$ be the instance with u removed from G. If G is clear in the context, we omit the subscript to write $M_G(\vec{y})$ as $M(\vec{y})$, and $M_{G_{\text{-}u}}(\vec{y}_{\text{-}u})$ as $M(\vec{y}_{\text{-}u})$. We remark that when running Balanced Ranking on instance $G_{\text{-}u}$, the lookahead water levels remain defined by instance G. In other words, in the thought experiment that removes u, we assume that the ranks and lookahead water levels of vertices other than u remain unchanged.

Definition III.1 (Active and Passive)**.** *If an edge* (u, v) *is matched in* $M(\vec{y})$ *at* u*'s deadline, we say that* u *is* active *and* v *is* passive.

The roles of active and passive vertices in the analysis are similar to the online and offline vertices respectively in the model of Karp et al. [2].

A main structural property of Ranking is the alternating path property that characterizes how the matching changes when the rank of a vertex changes. It also holds to Balanced Ranking. The proof can be found in the full version of this paper.

Lemma III.2 (Alternating Path)**.** *In a bipartite instance* G*, if* u *is matched in* $M(\vec{y})$*, no neighbor of* u *gets better from* $M(\vec{y})$ *to* $M(\vec{y}_{\text{-}u})$*. Here, passive is better than active, and active is better than unmatched. Conditioned on being passive, it is better to match a vertex with an earlier deadline. Conditioned on being active, it is better to match a vertex* v *with larger* $1-g(y_v)-f(x_v)$*.*

Next, we define the an important set of concepts called marginal ranks.

Definition III.2 (Marginal Rank)**.** *For any instance*

G, any vertex u, and any ranks $\vec{y}_{\text{-}u}$ of other vertices, the marginal rank of u w.r.t. G and $\vec{y}_{\text{-}u}$, denoted by $\lambda_u(G, \vec{y}_{\text{-}u})$, is the largest rank of u such that it is passive, i.e., $\lambda_u(G, \vec{y}_{\text{-}u}) = \sup\{y_u :$ u is passive in $M(y_u, \vec{y}_{\text{-}u})\}$.

For any pair of neighbors (u, v) where u's deadline is earlier than v's, we focus on the instance up to the deadline of u. For simplicity, we assume u's deadline to be the end of the instance and define the following marginal ranks with respect to the instance right after u's deadline.

Definition III.3 (Marginal Ranks τ and γ)**.** *Fix any instance G, any edge (u, v), and any ranks $\vec{y}_{\text{-}uv}$ of the vertices other than u and v. Let $\tau = \lambda_u(G_{\text{-}v}, \vec{y}_{\text{-}uv})$ be the marginal rank of u w.r.t. instance $G_{\text{-}v}$ with v removed, and ranks $\vec{y}_{\text{-}uv}$. Similarly, let $\gamma = \lambda_v(G_{\text{-}u}, \vec{y}_{\text{-}uv})$.*

Definition III.4 (Marginal Rank θ)**.** *Fix any instance G, any edge (u, v) in which u has an earlier deadline, any rank y_u of u, and any ranks $\vec{y}_{\text{-}uv}$ of the vertices other than u and v. Let $\theta(y_u) = \lambda_v(G, (y_u, \vec{y}_{\text{-}uv}))$ be the marginal rank of v w.r.t. G and ranks $(y_u, \vec{y}_{\text{-}uv})$.*

In fact we are only interested in $\theta(y_u)$ for $y_u > \tau$. The next lemma states that it suffices to consider a single value θ.

Lemma III.3. *There exists $\theta \geq \gamma$ such that $\theta(y_u) = \theta$ for any $y_u > \tau$.*

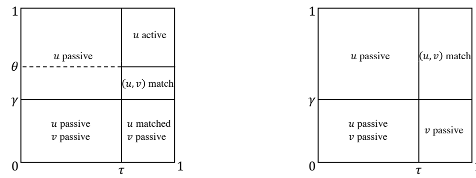
Proof: Consider the graph with v removed and $y_u = \tau^+$. By the definition of τ, u remains unmatched before its deadline. Consider inserting v with $y_v \in (\gamma, 1)$. According to the definition of γ, v must also be unmatched before u's deadline. That is, for any $y_u \in (\tau, 1)$ and $y_v \in (\gamma, 1)$, both u, v are unmatched before u's deadline. Note that at this moment, the rank of u does not play any role for its decision. Hence, there exists a common θ such that v would matched by u iff $y_v < \theta$. ■

We remark that θ may be 1, in which case v is passive regardless of its rank y_v. We will treat as a degenerate case and will handle it separately in the analysis (see Lemma III.4 and Figure 1b).

The marginal ranks τ, γ, and θ provide a characterization of the matching results of u and v as their ranks change. This is summarized in the following lemma, whose counterpart for Ranking were shown as Lemma 4.1, 4.2, and 4.3 in Huang et al. [5]. See also Figure 1 for a more visualized illustration.

Lemma III.4. *For any instance G, any edge (u, v) where u has an earlier deadline than v, any ranks $\vec{y}_{\text{-}uv}$ of other vertices, and the corresponding marginal ranks τ, γ, and θ, we have:*

- *u is passive when $y_u \in (0, \tau)$ and $y_v \in (0, 1)$;*



(a) General case: $\theta < 1$. (b) Degenerate case: $\theta = 1$.

Figure 1: The horizontal and vertical axes correspond to y_u, y_v respectively.

- *v is passive when $y_v \in (0, \gamma)$ and $y_u \in (0, 1)$;*
- *for any $y_u \in (\tau, 1)$, v is matched if and only if $y_v \in (0, \theta)$;*
- *for any $y_u \in (\tau, 1)$ and $y_v \in (\gamma, \theta)$, u actively matches v;*
- *for any $y_u \in (\tau, 1)$ and $y_v \in (\theta, 1)$, $\alpha_u \geq 1 - g(\theta) - f(x_v^{(u)})$, i.e., u's gain is at least what v offers at its marginal rank $y_v = \theta$;*
- *when $\theta < 1$, u is matched when $y_u \in (\tau, 1)$ and $y_v \in (0, \gamma)$; if we further have u is active, then $\alpha_u \geq 1 - g(\theta) - f(x_v^{(u)})$.*

Proof: We prove the statements sequentially. By the definition of τ, u is passively matched when $y_u \leq \tau$ and v is removed from the graph. By Lemma III.2, inserting v (with any rank) to the graph cannot make u worse. Hence, u must be passive. Similarly, v is passive when $y_v \leq \gamma$. This finishes the proof of the first and the second statements.

The third and the fourth statements hold by the definition of θ. Furthermore, consider when $y_u \in (\tau, 1)$ and $y_v = \theta^+$, u has v as a candidate but decides to choose another vertex z. Note that v offers $g(\theta) + f(x_v^{(u)})$. We have $\alpha_u = 1 - g(y_z) - f(x_z^{(u)}) \geq 1 - g(\theta) - f(x_v^{(u)})$. When we further increase the rank y_v, u's matching status shall not change. This concludes the fifth statement.

Finally, when $\theta < 1$, consider the graph with v removed and when $y_u \in (\tau, 1)$. This is equivalent to the case when $y_v > \theta$ and according to the previous discussion, u matches a vertex z and $\alpha_u \geq 1 - g(\theta) - f(x_v^{(u)})$. By Lemma III.2, after inserting vertex v with rank $y_v \in (0, \gamma)$, u's matching status becomes no worse than actively choosing z. In other words, u must be matched when $y_u \in (\tau, 1)$ and $y_v \in (0, \gamma)$. Furthermore, if u is active, α_u does not decrease, i.e. $\alpha_u \geq 1 - g(y_z) - f(x_z^{(u)}) \geq 1 - g(\theta) - f(x_v^{(u)})$. ■

C. Analysis of Balanced Ranking

Recall the randomized online primal dual framework as in Lemma II.1. Further recall that reverse weak duality in expectation holds trivially with equality by our definition of the dual variables. It remains to show approximate dual feasibility in expectation, i.e., to lower bound $\mathbb{E}\big[\alpha_u + \alpha_v\big]$. Since the dual variables depend on functions f and g, the lower bound will also be expressed in terms of these functions. It shall not be surprising that the contribution from g is identical to the bound by Huang et al. [5]. After all, the algorithm degenerates to Ranking if we let $f(x) \equiv 0$. For brevity, we denote the lower bound derived by Huang et al. [5] as a function $G : [0,1]^3 \to [0,1]$ as

$$G(\tau,\gamma,\theta) \stackrel{\text{def}}{=} \begin{cases} \int_0^\tau g(y_u)\mathrm{d}y_u + \int_0^\gamma g(y_v)\mathrm{d}y_v \\ +\big(1-\tau\big)\cdot\big(1-\gamma-(1-\theta)g(\theta)\big) \\ +\ \gamma\cdot\int_\tau^1 \min\big\{(1-g(\theta)), g(y_u)\big\}\mathrm{d}y_u, \theta<1\ ; \\ \int_0^\tau g(y_u)\mathrm{d}y_u \\ +\int_0^\gamma g(y_v)\mathrm{d}y_v + (1-\tau)\cdot(1-\gamma), \quad \theta = 1\ . \end{cases} \tag{1}$$

We lower bound the approximate dual feasibility in the following main technical lemma.

Observe that the bound $G(\tau,\gamma,\theta)$ in Eqn. (1) is *local*, in the sense that it is achieved by taking expectation over y_u and y_v only, for an arbitrarily fixed $\vec{y}_{\text{-}u,v}$. In contrast, our lower bound in Eqn. (2) is *global*, in the sense that we need to take expectation of $G(\tau,\gamma,\theta)$ over $\vec{y}_{\text{-}u,v}$. Additionally the bound due to function f is also global, as it takes as input the lookahead water levels.

Lemma III.5. *For any edge (u,v) in which u has an earlier deadline, we have:*

$$\mathbb{E}\big[\alpha_u+\alpha_v\big] \ge \mathbb{E}\big[G(\tau,\gamma,\theta)\big] + F(x_u, x_v^{(u)})\ , \tag{2}$$

where F is defined as: $F(x_u, x_v^{(u)}) \stackrel{\text{def}}{=} \int_0^{x_u} f(x)\mathrm{d}x + \int_0^{x_v^{(u)}} f(x)\mathrm{d}x - (1-x_u)\cdot f(x_v^{(u)})$.

Recall that $x_u = \Pr\big[u \textit{ passive}\big]$ *and* $x_v^{(u)} = \Pr\big[v \textit{ passive after } u\textit{'s deadline}\big]$.

Proof Sketch: We focus on the bound due to f. For a complete proof that includes the proof on the bound due to g, please refer to the full version of this paper. We lower bound the value of α_u and α_v right after the deadline of u, because we have little control over what happens to u and v after that.

If u has been passively matched by some vertex w, what u keeps to itself has term $f(x_u^{(w)})$. We use $x'_u = x_u^{(w)}$ to denote u's water level right after it is passively matched. Similarly, if v has been passively matched, what v keeps to itself has term $f(x'_v)$. If u actively matches at its deadline, we shall bound its best offer by the offer of v, which has a negative term $-f\big(x_v^{(u)}\big)$. Observe that v cannot be active by u's deadline by the assumption that u has an earlier deadline. In sum, when lower bounding $\mathbb{E}\left[\alpha_u+\alpha_v\right]$, the bound due to f is at least:

$$\mathbb{E}\Big[\mathbb{1}[u \text{ passive}]\cdot f(x'_u) + \mathbb{1}[v \text{ passive}]\cdot f(x'_v) - \mathbb{1}[u \text{ active}]\cdot f(x_v^{(u)})\Big]\ . \tag{3}$$

Here we use $\mathbb{1}[\text{event}]$ to denote the indicator, which is 1 if the event holds; 0 otherwise.

Next, we crucially rely on the fact that Balanced Ranking uses lookahead water levels to lower bound the first term in Eqn. (3) by the first integral in the lemma. Let $w_1, w_2, \ldots, w_k$, where $w_k = u$, be the sequence of vertices with deadlines earlier than u, sorted by the deadlines. By the definition, we have $x'_u = x_u^{(w_i)}$ with probability $x_u^{(w_i)} - x_u^{(w_{i-1})}$. For convenience, we define $x_u^{(w_0)} = 0$.

$$\begin{aligned}\mathbb{E}\Big[\mathbb{1}[u \text{ passive}]\cdot f(x'_u)\Big] &= \sum_{i=1}^{k}\Big(x_u^{(w_i)} - x_u^{(w_{i-1})}\Big) f\big(x_u^{(w_i)}\big) \\ &\ge \int_{x_u^{(w_0)}}^{x_u^{(w_k)}} f\big(x\big)\mathrm{d}x = \int_0^{x_u} f\big(x\big)\mathrm{d}x\ . \end{aligned} \tag{4}$$

By the same argument, we have $\mathbb{E}\Big[\mathbb{1}[v \text{ passive}]\cdot f(x'_v)\Big] \ge \int_0^{x_v^{(u)}} f\big(x\big)\mathrm{d}x$.

Finally, the third term in Eqn. (3) is lower bounded by the third term in the lemma because:

$$\begin{aligned}&\mathbb{E}\Big[\mathbb{1}[u \text{ active}]\cdot f(x_v^{(u)})\Big] = \Pr\big[u \text{ active}\big] f(x_v^{(u)}) \\ &\le \big(1-\Pr\big[u \text{ passive}\big]\big) f(x_v^{(u)}) = \big(1-x_u\big) f(x_v^{(u)})\ . \end{aligned} \tag{5}$$

Putting the bounds together concludes the proof. ∎

Failed Attempt: Handling f and g Separately.: It remains to design functions f and g so that the RHS of Eqn. (2) is at least the competitive ratio Γ. Suppose we do not have any control over the marginal ranks τ, γ, θ and water levels x_u, $x_v^{(u)}$, i.e., they can take any arbitrary combination of values in $[0,1]$. Then, the designs of f and g become two separate problems. Huang et al. [5] found the optimal g such that $\min_{\tau,\gamma,\theta}\{G(\tau,\gamma,\theta)\} = \Omega \approx 0.567$ to show that Ranking is Ω-competitive. Unfortunately, the bound $F(x_u, x_v^{(u)})$ for any nondecreasing function f is at most 0 at $x_u = 0$.

In order to beat the Ω competitive ratio, which is proved tight for Ranking [5], it is crucial to establish a connection between the marginal ranks and the water levels.

Binding Marginal Ranks and Waterlevels.: Fortunately, the marginal ranks threshold ranks τ, γ, θ and water levels x_u, $x_v^{(u)}$ are *not* arbitrary. Recall from the first conclusion of Lemma III.4 that u is passive for all $y_u \in [\tau, 1]$ and $y_v \in [0, 1]$. Hence conditioned on any $\vec{y}_{\text{-}u}$, the probability that u is passive is at least τ. Taking the expectation over $\vec{y}_{\text{-}u}$ yields the following lemma.

Lemma III.6. *For any edge* (u, v) *in which* u *has an earlier deadline, we have* $\mathbb{E}\big[\tau\big] \le x_u$.

Our Final Plan.: To utilize the above relation between τ and x_u, we introduce an auxiliary *convex* function $\ell : [0, 1] \to [0, 1]$ such that $\ell(\tau)$ lower bounds $\min_{\gamma \le \theta}\{G(\tau, \gamma, \theta)\}$. Then, we can lower bound the first term on the RHS of Eqn. (2) as:

$$\mathbb{E}\big[G(\tau, \gamma, \theta)\big] \ge \mathbb{E}\big[\ell(\tau)\big] \ge \ell\big(\mathbb{E}\big[\tau\big]\big) \ . \qquad \text{(convexity of } \ell\text{)}$$

Further observe that $F(x_u, x_v^{(u)})$ is nondecreasing in x_u. We have $F\big(x_u, x_v^{(u)}\big) \ge F\big(\mathbb{E}\big[\tau\big], x_v^{(u)}\big)$ by Lemma III.6. It remains to lower bound $\ell\big(\mathbb{E}\big[\tau\big]\big) + F\big(\mathbb{E}\big[\tau\big], x_v^{(u)}\big)$, for any $\mathbb{E}\big[\tau\big]$ and $x_v^{(u)}$.

A set of sufficient conditions for Γ-competitiveness w.r.t. functions f, g, and the competitive ratio Γ is summarized as the next lemma. The proof of Lemma III.7 is given in the full version of this paper.

Lemma III.7. *There are increasing function* $g : [0, 1] \to [0, 1]$*, non-decreasing function* $f : [0, 1] \to [0, 1]$ *and a convex function* $\ell : [0, 1] \to [0, 1]$ *such that for* $\Gamma = 0.5690$*:*

$$\begin{aligned}
&\forall \tau, \gamma \in [0, 1], \forall \theta \in [\gamma, 1) : \\
&\quad \textstyle\int_0^\tau g(y_u)\mathrm{d}y_u + \int_0^\gamma g(y_v)\mathrm{d}y_v \\
&\qquad + \big(1 - \tau\big)\big(1 - \gamma - (1 - \theta)g(\theta)\big) \\
&\qquad + \textstyle\int_\tau^1 \gamma \cdot \min\big\{g(y_u), 1 - g(\theta)\big\}\mathrm{d}y_u \ge \ell(\tau) \ ; \\
&\forall \tau, \gamma \in [0, 1] : \\
&\quad \textstyle\int_0^\tau g(y_u)\mathrm{d}y_u + \int_0^\gamma g(y_v)\mathrm{d}y_v \\
&\qquad + \big(1 - \tau\big)\big(1 - \gamma\big) \ge \ell(\tau) \ ; \\
&\forall \mathbb{E}[\tau], x_v^{(u)} \in [0, 1] : \\
&\quad \ell(\mathbb{E}[\tau]) + \textstyle\int_0^{\mathbb{E}[\tau]} f(x)\mathrm{d}x + \int_0^{x_v^{(u)}} f(x)\mathrm{d}x \\
&\qquad - \big(1 - \mathbb{E}[\tau]\big)f(x_v^{(u)}) \ge \Gamma \ ; \\
&\forall x \in [0, 1], \forall y \in [0, x] : \\
&\quad f(x) - f(y) \le x - y \ ; \\
&\forall x \in [0, 1], \forall y \in [0, x] : \\
&\quad g(x) - g(y) \ge \frac{x - y}{100} \ ; \\
&\quad g(1) + f(1) \le 1 \ .
\end{aligned}$$

Theorem III.1 (Theorem I.1 Restated)**.** *Balanced Ranking with the functions* f *and* g *chosen in Lemma III.7 is* 0.569*-competitive for fully online matching on bipartite graphs.*

Proof: We have discussed all the ingredients in this section. It remains to put them together. Let f, g, and ℓ be the functions constructed in Lemma III.7. Observe that f, g satisfy the Lipschitzness and reverse Lipschitzness assumed in Section III-A. Our algorithm is well-defined. Since the function f and g are nonnegative and $g(1)+f(1) \le 1$, the dual variables α_u's are nonnegative.

Recall that $x_u = \Pr\big[u \text{ passive}\big]$ and $x_v^{(u)} = \Pr\big[v \text{ passive after } u\text{'s deadline}\big]$. Approximate dual feasibility in expectation follows by Lemma III.5 and Lemma III.7 as

$$\begin{aligned}
\mathbb{E}\big[\alpha_u + \alpha_v\big] \ge& \mathbb{E}\big[G(\tau, \gamma, \theta)\big] + F\big(x_u, x_v^{(u)}\big) \\
\ge& \mathbb{E}\big[\ell(\tau)\big] + F\big(x_u, x_v^{(u)}\big) \\
\ge& \ell\big(\mathbb{E}[\tau]\big) + F\big(x_u, x_v^{(u)}\big) \\
\ge& \ell\big(\mathbb{E}[\tau]\big) + F\big(\mathbb{E}[\tau], x_v^{(u)}\big) \ge \Gamma = 0.5690 \ .
\end{aligned}$$

Finally, recall that reverse weak duality in expectation follows trivially with equality by our definition of the dual variables. ■

IV. Eager Water-Filling Algorithm

In this section we present the Eager Water-filling algorithm for fractional fully online matching and prove Theorem I.2. We first briefly summarize the competitive analysis of Water-filling algorithm [5] to build intuition. Recall that Water-filling is a lazy algorithm where each vertex sits back and waits until its deadline. At the deadline of a vertex u, Water-filling continuously matches u to the unmatched neighbors with the smallest matched portion (a.k.a. water level). The algorithm simultaneously updates the dual variables. Whenever $\mathrm{d}x$ fraction of edge (u, v) is matched at u's deadline, we increase α_u, α_v by $(1 - f(x_v))\mathrm{d}x$ and $f(x_v)\mathrm{d}x$ respectively, where x_v is the current water level of v. Huang et al. [5] conclude the competitive ratio of Water-filling by showing approximate dual feasibility with an appropriate choice of f.

The primal-dual analysis gives an intuitive economic interpretation of the Water-filling algorithm. At any moment, each vertex v prices itself at $f(x_v)$ according to the current water level and offers a share of $1 - f(x_v)$ to its neighbor. At the deadline of a vertex u, it chooses the unmatched neighbor that is willing to give u the largest share of gain. From this viewpoint, however, Water-filling is unnatural in the following scenario. Suppose at u's arrival, it has an existing neighbor v who is willing to offer a share of the gain that is larger than what u can get from being passive matched later, i.e. $1 - f(x_v) \ge f(x_u)$. Why would u prefer to wait as in Water-filling, instead of grabbing v immediately? By waiting there is risk that 1) v is taken by some other vertex before u's deadline and that 2) u is passively matched before its own deadline which gives a lower portion of the gain to u. To this end, we propose the following variant of Water-filling.

Eager Water-filling.: Fix an increasing function $f : [0,1] \to [0,1]$. Initialize all x_{uv}'s and α_u's to be zero. For convenience of analysis we also fix the requirement that $f(0) = 0$ and $f(1) = 1$.

1) Upon the arrival of a vertex u, u continuously matches the neighbor v with lowest water level if $f(x_u) + f(x_v) \leq 1$. The process increases x_u and the lowest water level of neighbors of u until $f(x_u) + f(x_v) > 1$ for all neighbor v of u.
2) At the deadline of u, u continuously matches the neighbor v with lowest water level until $x_u = 1$, or $x_v = 1$ for all neighbor v of u.

Note that the second step of Eager Water-filling is the same as Water-filling.

In both steps, when we match u with its neighbor v, we consider u as the active vertex and v as the passive vertex. When x_{uv} increases by $\mathrm{d}x$, we update the dual variables α_u and α_v as follows:

$$\mathrm{d}\alpha_u = (1 - f(x_v))\mathrm{d}x \quad \text{and} \quad \mathrm{d}\alpha_v = f(x_v)\mathrm{d}x.$$

A. Analysis of Eager Water-filling

By Lemma II.1, it suffices to show that for any pair of neighbors u and v we have $\alpha_u + \alpha_v \geq \Gamma$ in order to prove Eager Water-filling is Γ-competitive. Unlike Balanced Ranking/Ranking, Eager Water-filling is a deterministic algorithm and thus, no randomness is involved for the dual variables α_u's. Fix any pair of neighbors u and v, and assume u has an earlier deadline than v.

Let p_u be the water level of u right *before* u's deadline. Let p_v be the water level of v right *after* u's deadline. Let t_u, t_v be the water levels of u and v right *after* their arrivals, respectively. We prove the following lower bound on the gain of u and v.

Lemma IV.1. *Right after u's deadline, we have*

$$\alpha_v + \alpha_u \geq t_v \cdot f(t_v) + \int_{t_v}^{p_v} f(x)\mathrm{d}x + t_u \cdot f(t_u) + \int_{t_u}^{p_u} f(x)\mathrm{d}x + (1 - p_u) \cdot (1 - f(p_v)). \quad (6)$$

Proof: Recall that when v arrives, v matches some neighbor actively until $x_v = t_v$. Moreover, when x_v increases (actively) from 0 to t_v, the neighbor z it matches always satisfies $f(x_z) + f(t_v) \leq 1$. Thus when x_v increases by $\mathrm{d}x$ the gain of v is $(1 - f(x_z))\mathrm{d}x \geq f(t_v)\mathrm{d}x$. Hence right after v's arrival we have $\alpha_v \geq t_v \cdot f(t_v)$. When x_v further increases from t_v to p_v between v's arrival and u's deadline, α_v increases at the rate of $f(x_v)$. Thus after u's deadline we have $\alpha_v \geq t_v \cdot f(t_v) + \int_{t_v}^{p_v} f(x)\mathrm{d}x$.

Similarly, right before u's deadline we have $\alpha_u \geq t_u \cdot f(t_u) + \int_{t_u}^{p_u} f(x)\mathrm{d}x$. If $p_v = 1$, then $(1 - p_u) \cdot (1 - f(p_v)) = 0$ and the statement is proved. Otherwise at u's deadline, x_u increases (actively) from p_u to 1, and u always matches a neighbor with water level at most p_v. Thus after the deadline of u we have $\alpha_u \geq t_u \cdot f(t_u) + \int_{t_u}^{p_u} f(x)\mathrm{d}x + (1 - p_u) \cdot (1 - f(p_v))$.

Putting the lower bounds of α_u and α_v together concludes the proof. ■

Comparison with Water-filling.: We make a comparison to the competitive analysis of Water-filling by Huang et al. [5]. Let p_u, p_v be defined in the same way as Eager Water-filling for Water-filling and dual variables be also updated in the same way. Observe that Water-filling is exactly the second step of our Eager Water-filling algorithm. Huang et al. proved that

$$\alpha_u + \alpha_v \geq \int_0^{p_u} f(x)\mathrm{d}x + (1 - p_u)(1 - f(p_v)) + \int_0^{p_v} f(x)\mathrm{d}x. \quad (7)$$

Observe that Eqn. (6) is at least as good as Eqn. (7), because $t \cdot f(t) \geq \int_0^t f(x)\mathrm{d}x$ for all t. On the other hand, we have not shown any constraint on the values of t_u, t_v. In the case when $t_u = t_v = 0$, Eqn. (6) degenerates to Eqn. (7).

We continue our analysis by observing that if v arrives earlier than u then right after u's arrival we have $f(t_u) + f(x_v) > 1$, and $x_v \leq p_v$. Thus we have the constraint that $f(t_u) + f(p_v) > 1$. Similarly, if u arrives earlier than v then we have $f(t_v) + f(p_u) > 1$.

Combining the constraints on t_u, p_u, t_v, p_v with Lemma IV.1, we show that there exists function f such that the total gain of u and v combined is strictly larger than the ratio $2 - \sqrt{2}$ that is proved tight for Water-filling.

B. Reformulating the Lower Bound

It remains to find an increasing function f such that the minimum of RHS of Eqn. (6), over possible values of t_u, t_v, p_u, p_v, is maximized. In this section we reformulate the lower bound and eliminate t_u and t_v from the lower bound.

Since f is strictly increasing, it is easy to see that the RHS of Eqn. (6) is increasing w.r.t. both t_u and t_v. Indeed, the function $t \cdot f(t) - \int_0^t f(x)\mathrm{d}x$ is monotonically increasing in t. Thus the minimum is achieved when t_u and t_v are minimized, subject to the constraint

$f(t_u) + f(p_v) > 1$ if v arrives earlier than u, or
$f(t_v) + f(p_u) > 1$ if u arrives earlier than v.

Let $h(\cdot) = f^{-1}(\cdot)$ be the inverse function of f. Note that h is also an increasing function defined on $[0, 1]$ such that $h(0) = 0$ and $h(1) = 1$. For any $p \in [0, 1]$, we have

$$\int_0^p f(x)\mathrm{d}x = p \cdot f(p) - \int_0^{f(p)} h(y)\mathrm{d}y.$$

Lemma IV.2. *If v arrives earlier than u, then we have*

$$\alpha_u + \alpha_v \geq \min_q \left\{ q \cdot h(q) - \int_0^q h(y)\mathrm{d}y + 1 - q \right\}.$$

Proof: Let $q_u = f(p_u), q_v = f(p_v)$. Recall that if v arrives earlier than u then the minimum of RHS of Eqn.(6) is achieved when $t_u = f^{-1}(1 - f(p_v)) = h(1 - q_v)$ and $t_v = 0$:

$$\alpha_v + \alpha_u \geq \int_0^{p_v} f(x)\mathrm{d}x + t_u \cdot f(t_u) + \int_{t_u}^{p_u} f(x)\mathrm{d}x + (1 - p_u) \cdot (1 - f(p_v)).$$

Observe that the derivative of RHS of the above equation over p_u is $f(p_u)+f(p_v)-1 \geq 0$, which implies that the minimum is achieved when p_u is minimized, i.e., $p_u = t_u$. Thus we have

$$\alpha_v+\alpha_u \geq \int_0^{p_v} f(x)\mathrm{d}x+t_u\cdot f(t_u)+(1-t_u)\cdot(1-f(p_v)).$$

Using $t_u = h(1 - q_v)$ we have $f(t_u) = 1 - q_v = 1 - f(p_v)$, which implies

$$\begin{aligned}\alpha_v + \alpha_u &\geq \int_0^{p_v} f(x)\mathrm{d}x + 1 - f(p_v) \\ &= h(q_v) \cdot q_v - \int_0^{q_v} h(y)\mathrm{d}y + 1 - q_v.\end{aligned}$$

Taking minimum of the RHS over q_v yields the lemma. ■

Lemma IV.3. *If u arrives earlier than v, then we have*

$$\alpha_u+\alpha_v \geq \min_{q_u,q_v} \Big\{ q_u \cdot h(q_u) - \int_0^{q_u} h(y)\mathrm{d}y + q_v \cdot h(q_v) - \int_0^{q_v} h(y)\mathrm{d}y + \int_0^{1-q_u} h(y)\mathrm{d}y + (1-h(q_u))\cdot(1-q_v) \Big\}.$$

Proof: Let $q_u = f(p_u), q_v = f(p_v)$. If u arrives earlier than v then the minimum of RHS of Eqn.(6) is achieved when $t_v = f^{-1}(1 - f(p_u)) = h(1 - q_u)$ and $t_u = 0$:

$$\begin{aligned}&\alpha_v + \alpha_u \\ \geq& t_v \cdot f(t_v) + \int_{t_v}^{p_v} f(x)\mathrm{d}x + \int_0^{p_u} f(x)\mathrm{d}x \\ &+ (1 - p_u) \cdot (1 - f(p_v)) \\ =& t_v \cdot f(t_v) + \int_0^{p_v} f(x)\mathrm{d}x - \int_0^{t_v} f(x)\mathrm{d}x \\ &+ \int_0^{p_u} f(x)\mathrm{d}x + (1 - p_u) \cdot (1 - f(p_v)) \\ =& h(1 - q_u) \cdot (1 - q_u) + \left(h(q_v) \cdot q_v - \int_0^{q_v} h(y)\mathrm{d}y \right) \\ &- \left(h(1 - q_u) \cdot (1 - q_u) - \int_0^{1-q_u} h(y)\mathrm{d}y \right) \\ &+ \left(h(q_u) \cdot q_u - \int_0^{q_u} h(y)\mathrm{d}y \right) \\ &+ (1 - h(q_u)) \cdot (1 - q_v) \\ =& q_u \cdot h(q_u) - \int_0^{q_u} h(y)\mathrm{d}y + q_v \cdot h(q_v) - \int_0^{q_v} h(y)\mathrm{d}y \\ &+ \int_0^{1-q_u} h(y)\mathrm{d}y + (1 - h(q_u)) \cdot (1 - q_v).\end{aligned}$$

Taking minimum of the RHS over q_u and q_v yields the lemma. ■

Finally, we use factor revealing lp techniques to find function h with the following property. The proof of Lemma IV.4 is provided in the full version of the paper.

Lemma IV.4. *There exists an increasing function $h : [0, 1] \mapsto [0, 1]$ such that for $\Gamma = 0.5926$:*

$$\forall q \in [0, 1], \quad q \cdot h(q) - \int_0^q h(y)\mathrm{d}y + 1 - q \geq \Gamma, \tag{8}$$

$$\forall q_u, q_v \in [0, 1], \quad q_u \cdot h(q_u) - \int_0^{q_u} h(y)\mathrm{d}y + q_v \cdot h(q_v) - \int_0^{q_v} h(y)\mathrm{d}y + \int_0^{1-q_u} h(y)\mathrm{d}y + (1 - h(q_u)) \cdot (1 - q_v) \geq \Gamma, \tag{9}$$

$$h(0) = 0, h(1) = 1. \tag{10}$$

Theorem IV.1 (Theorem I.2 Restated)**.** *Eager Water-filling with the function $f = h^{-1}$ where h is chosen in Lemma IV.4 is 0.592-competitive for fractional fully online matching on general graphs.*

Proof: We conclude the competitive ratio of Eager Water-filling by putting the lemmas together. Approximate dual feasibility follows by the two cases. If v

arrives earlier than u, we have

$$\alpha_u + \alpha_v \geq \min_q \left\{ q \cdot h(q) - \int_0^q h(y)\mathrm{d}y + 1 - q \right\} \quad \text{(Lemma IV.2)}$$
$$\geq \Gamma = 0.592 \ . \quad \text{(Eqn. (8))}$$

If u arrives earlier than v, we have

$$\begin{aligned}
&\alpha_u + \alpha_v \\
&\geq \min_{q_u, q_v} \left\{ q_u \cdot h(q_u) - \int_0^{q_u} h(y)\mathrm{d}y \right. \\
&\qquad + q_v \cdot h(q_v) - \int_0^{q_v} h(y)\mathrm{d}y \\
&\qquad \left. + \int_0^{1-q_u} h(y)\mathrm{d}y + (1 - h(q_u)) \cdot (1 - q_v) \right\} \quad \text{(Lemma IV.3)} \\
&\geq \Gamma = 0.592 \ . \quad \text{(Eqn. (9))}
\end{aligned}$$

Finally, recall that reverse weak duality follows trivially with equality by our definition of the dual variables. ■

V. Future Directions

Balanced Ranking vs. Ranking on General Graphs.: An immediate next question about Balanced Ranking is whether it is still better than Ranking on general graphs. This is beyond the scope of the current paper since a tight analysis of Ranking remains elusive. An easier task is to show that Balanced Ranking is strictly better than 0.5211-competitive on general graphs. We leave these questions for future research.

Balanced Ranking with Eager Matches.: Another interesting direction is to explore the power of eager matches in integral fully online matching algorithms. There is a natural definition of Eager Ranking where a vertex v may be eagerly matched on its arrival to a neighbor u if $1 - g(y_u) \geq g(y_x)$. However, it is at best $\Omega \approx 0.567$-competitive due to the same hard instance for Ranking by Huang et al. [4, 6]. There is also a natural definition of Eager Balanced Ranking but its analysis seems to require ideas beyond those in this paper.

References

[1] Z. Huang, Z. G. Tang, X. Wu, and Y. Zhang, "Fully online matching II: beating ranking and water-filling," *CoRR*, vol. abs/2005.06311, 2020.

[2] R. M. Karp, U. V. Vazirani, and V. V. Vazirani, "An optimal algorithm for on-line bipartite matching," in *STOC*, 1990, pp. 352–358.

[3] B. Kalyanasundaram and K. Pruhs, "An optimal deterministic algorithm for online b-matching," *Theoretical Computer Science*, vol. 233, no. 1-2, pp. 319–325, 2000.

[4] Z. Huang, N. Kang, Z. G. Tang, X. Wu, Y. Zhang, and X. Zhu, "How to match when all vertices arrive online," in *STOC*, 2018, pp. 17–29.

[5] Z. Huang, B. Peng, Z. G. Tang, R. Tao, X. Wu, and Y. Zhang, "Tight competitive ratios of classic matching algorithms in the fully online model," in *SODA*. SIAM, 2019, pp. 2875–2886.

[6] Z. Huang, N. Kang, Z. G. Tang, X. Wu, Y. Zhang, and X. Zhu, "Fully online matching," *J. ACM*, vol. 67, no. 3, pp. 17:1–17:25, 2020.

[7] N. R. Devanur, K. Jain, and R. D. Kleinberg, "Randomized primal-dual analysis of RANKING for online bipartite matching," in *SODA*. SIAM, 2013, pp. 101–107.

[8] Y. Wang and S. C. Wong, "Two-sided online bipartite matching and vertex cover: Beating the greedy algorithm," in *ICALP*, 2015, pp. 1070–1081.

[9] N. Buchbinder, D. Segev, and Y. Tkach, "Online algorithms for maximum cardinality matching with edge arrivals," in *ESA*, ser. LIPIcs, vol. 87. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2017, pp. 22:1–22:14.

[10] G. Goel and A. Mehta, "Online budgeted matching in random input models with applications to adwords," in *SODA*, 2008, pp. 982–991.

[11] B. Birnbaum and C. Mathieu, "On-line bipartite matching made simple," *ACM SIGACT News*, vol. 39, no. 1, pp. 80–87, 2008.

[12] A. Eden, M. Feldman, A. Fiat, and K. Segal, "An economic-based analysis of RANKING for online bipartite matching," *CoRR*, vol. abs/1804.06637, 2018.

[13] U. Feige, "Tighter bounds for online bipartite matching," *CoRR*, vol. abs/1812.11774, 2018.

[14] A. Mehta, A. Saberi, U. Vazirani, and V. Vazirani, "Adwords and generalized on-line matching," in *FOCS*, 2005, pp. 264–273.

[15] N. Buchbinder, K. Jain, and J. Naor, "Online primal-dual algorithms for maximizing ad-auctions revenue," in *ESA*, ser. Lecture Notes in Computer Science, vol. 4698. Springer, 2007, pp. 253–264.

[16] N. R. Devanur and K. Jain, "Online matching with concave returns," in *STOC*. ACM, 2012, pp. 137–144.

[17] G. Aggarwal, G. Goel, C. Karande, and A. Mehta, "Online vertex-weighted bipartite matching and single-bid budgeted allocations," in *SODA*, 2011, pp. 1253–1264.

[18] J. Feldman, N. Korula, V. S. Mirrokni, S. Muthukrishnan, and M. Pál, "Online ad assignment with free disposal," in *WINE*, 2009, pp. 374–385.

[19] N. R. Devanur, Z. Huang, N. Korula, V. S. Mirrokni, and Q. Yan, "Whole-page optimization

and submodular welfare maximization with online bidders," *ACM Transactions on Economics and Computation*, vol. 4, no. 3, pp. 1–20, 2016.

[20] A. Mehta and D. Panigrahi, "Online matching with stochastic rewards," in *FOCS*. IEEE, 2012, pp. 728–737.

[21] A. Mehta, B. Waggoner, and M. Zadimoghaddam, "Online stochastic matching with unequal probabilities," in *SODA*. SIAM, 2014, pp. 1388–1404.

[22] Z. Huang and Q. Zhang, "Online primal dual meets online matching with stochastic rewards: configuration LP to the rescue," in *STOC (to appear)*, 2020.

[23] C. Karande, A. Mehta, and P. Tripathi, "Online bipartite matching with unknown distributions," in *STOC*, 2011, pp. 587–596.

[24] M. Mahdian and Q. Yan, "Online bipartite matching with random arrivals: an approach based on strongly factor-revealing LPs," in *STOC*, 2011, pp. 597–606.

[25] Z. Huang, Z. G. Tang, X. Wu, and Y. Zhang, "Online vertex-weighted bipartite matching: Beating 1-1/e with random arrivals," *ACM Transactions on Algorithms*, vol. 15, no. 3, pp. 1–15, 2019.

[26] B. Gamlath, M. Kapralov, A. Maggiori, O. Svensson, and D. Wajc, "Online matching with general arrivals," in *FOCS*. IEEE, 2019, pp. 26–37.

[27] I. Ashlagi, M. Burq, C. Dutta, P. Jaillet, A. Saberi, and C. Sholley, "Edge weighted online windowed matching," in *EC*, 2019, pp. 729–742.